

**UNIVERSIDADE FEDERAL DE SÃO CARLOS**

CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**PROCESSAMENTO DE CONHECIMENTO  
IMPRECISO COMBINANDO RACIOCÍNIO DE  
ONTOLOGIAS *FUZZY* E SISTEMAS DE  
INFERÊNCIA *FUZZY***

**CRISTIANE AKEMI YAGUINUMA**

**ORIENTADORA: PROFA. DRA. HELOISA DE ARRUDA CAMARGO  
COORIENTADORA: PROFA. DRA. MARILDE TEREZINHA PRADO SANTOS**

São Carlos – SP

Outubro/2013

**UNIVERSIDADE FEDERAL DE SÃO CARLOS**

CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**PROCESSAMENTO DE CONHECIMENTO  
IMPRECISO COMBINANDO RACIOCÍNIO DE  
ONTOLOGIAS *FUZZY* E SISTEMAS DE  
INFERÊNCIA *FUZZY***

**CRISTIANE AKEMI YAGUINUMA**

Tese apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Doutor em Ciência da Computação, área de concentração: Inteligência Artificial

Orientadora: Profa. Dra. Heloisa de Arruda Camargo

São Carlos – SP

Outubro/2013

**Ficha catalográfica elaborada pelo DePT da  
Biblioteca Comunitária/UFSCar**

Y12pc

Yaguinuma, Cristiane Akemi.

Processamento de conhecimento impreciso combinando raciocínio de ontologias *fuzzy* e sistemas de inferência *fuzzy* / Cristiane Akemi Yaguinuma. -- São Carlos : UFSCar, 2014. 174 f.

Tese (Doutorado) -- Universidade Federal de São Carlos, 2013.

1. Inteligência artificial. 2. Representação de conhecimento. 3. Ontologia. 4. *Fuzzy logic*. 5. Sistema *Fuzzy*. 6. Raciocínio aproximado. I. Título.

CDD: 006.3 (20<sup>a</sup>)

# Universidade Federal de São Carlos

Centro de Ciências Exatas e de Tecnologia

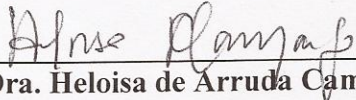
Programa de Pós-Graduação em Ciência da Computação

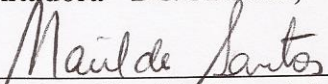
## “Processamento de Conhecimento Impreciso Combinando Raciocínio de Ontologias Fuzzy e Sistemas de Inferência Fuzzy”


Cristiane Akemi Yaguinuma

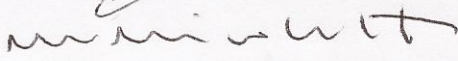
Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Doutor em Ciência da Computação.

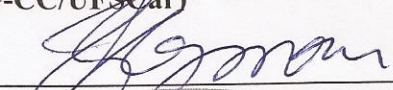
### Membros da Banca:

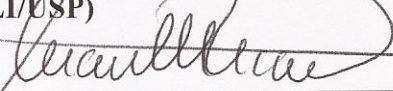
  
\_\_\_\_\_  
Profa. Dra. Heloisa de Arruda Camargo  
(Orientadora - DC/UFSCar)

  
\_\_\_\_\_  
Profa. Dra. Marildé Terezinha Prado Santos  
(Co-Orientadora - DC/UFSCar)

  
\_\_\_\_\_  
Prof. Dr. Estevam Rafael Hruschka Junior  
(DC/UFSCar)

  
\_\_\_\_\_  
Profa. Dra. Maria do Carmo Nicoletti  
(PPG-CC/UFSCar)

  
\_\_\_\_\_  
Prof. Dr. Fabio Gagliardi Cozman  
(POLI/USP)

  
\_\_\_\_\_  
Prof. Dr. Ivan Luiz Marques Ricarte  
(FEEC/UNICAMP)

São Carlos  
Dezembro/2013

Aos meus pais, Neusa e Kazuo.

Ao meu amor, Eduardo.

Ao saudoso professor, orientador e amigo, Mauro Biajiz.

## AGRADECIMENTOS

A Deus, pelo dom da vida, pela graça de completar mais esta etapa com saúde.

A meus queridos pais, Neusa e Kazuo, pelo amor, dedicação, compreensão e sabedoria ao valorizarem a herança mais rica que poderiam presentear-me: a educação.

A Eduardo, meu esposo, por estar sempre ao meu lado nos altos e baixos desta caminhada. Meu amor e companheiro para toda a vida.

A Edna e José Antônio, pelo carinho, compreensão e apoio, por acolherem-me como uma verdadeira filha.

A Heloisa, pela orientação, pelos sábios conselhos, por todo o aprendizado, além do grande exemplo de professora, pesquisadora e de pessoa íntegra e amiga, que sempre levarei comigo.

A Marilde, pela orientação, por acompanhar meu crescimento desde a graduação, sempre acreditando em meu trabalho.

Ao saudoso Mauro, pela motivação e confiança em meu potencial mesmo em um momento tão delicado de sua vida. Sua coragem diante dos desafios foi um grande exemplo para mim.

A Marek Reformat, pela oportunidade de ampliar meus horizontes e viver novas experiências durante o estágio de doutorado sanduíche na Universidade de Alberta.

A Carmo, pela cooperação e pelos conselhos que muito contribuíram para a minha formação como pesquisadora.

Aos coordenadores do PPG-CC durante o período do doutorado, por todo apoio e compreensão, sempre dispostos a ajudar da melhor forma possível.

Aos amigos do CIG, Pri, Dino, Fábio, Tati, Marcos e Mari, pelo carinho, amizade e cooperação, além dos deliciosos bolos de aniversário que tanto compartilhamos.

Aos amigos Kamila, Elis, Márcia, Fernanda e Daniel, pelo carinho, companheirismo e troca de experiências que tornaram esta jornada menos árdua.

Aos amigos do departamento, Cristina, Sandra Abib, Dona Vera e Augusto, pelo carinho, amizade e disposição para ajudar tanto em assuntos profissionais quanto pessoais.

A Walter, pela amizade e pelo apoio fundamental para a realização dos experimentos em cooperação com a Embrapa Gado de Leite.

Às amigas que conheci no exterior, Parisa, Sonbol, Xiao e Chris, que me acolheram em um lugar desconhecido e frio, porém repleto de boas recordações.

À CAPES e à Fapesp, pelo auxílio financeiro deste trabalho.

*“Quanto mais aumenta nosso conhecimento, mais evidente fica nossa ignorância”*

John F. Kennedy



## RESUMO

No contexto da Ciência da Computação, ontologias são utilizadas para representação de conhecimento em diversas aplicações, com o intuito de estruturar e tratar a semântica de domínios específicos. Embora representem e permitam inferir conhecimento implícito, as ontologias convencionais baseiam-se em um formalismo que não é capaz de expressar a imprecisão presente em fenômenos do mundo real e na linguagem humana. Para abordar esta limitação, há diversas pesquisas que investigam a incorporação de conceitos da teoria de conjuntos *fuzzy* e da lógica *fuzzy* em ontologias, resultando em ontologias *fuzzy*. Em especial, combinar o formalismo das ontologias *fuzzy* com o raciocínio baseado em regras *fuzzy*, utilizado com sucesso no contexto de sistemas de inferência *fuzzy*, pode proporcionar uma maior expressividade com relação às inferências envolvendo imprecisão. Neste sentido, o objetivo deste projeto de doutorado é explorar a integração do raciocínio de ontologias *fuzzy* e de sistemas de inferência *fuzzy*, resultando na definição e no desenvolvimento das abordagens HyFOM (*Hybrid integration of Fuzzy Ontology and Mamdani reasoning*) e FT-FIS (*Fuzzy Tableau and Fuzzy Inference System*). HyFOM baseia-se em uma arquitetura híbrida que combina motores de inferência existentes na literatura para ontologias, ontologias *fuzzy* e sistemas de inferência *fuzzy*, com foco na interação entre seus componentes independentes. FT-FIS define uma interface entre um algoritmo baseado em *tableau fuzzy* e um sistema de inferência *fuzzy*, incluindo o predicado *fuzzyRuleReasoning* que permite invocar o raciocínio baseado em regras *fuzzy* quando for necessário para as tarefas de raciocínio da ontologia *fuzzy*. A principal contribuição das arquiteturas de raciocínio de HyFOM e FT-FIS está na combinação de flexibilidade, em termos da semântica das regras *fuzzy*, com a colaboração entre as inferências de ambos tipos de raciocínio. Experimentos considerando a recomendação de atrações turísticas, baseados em dados sintéticos, revelaram que HyFOM e FT-FIS são capazes de proporcionar inferências integradas, além de uma aproximação mais expressiva da relação estabelecida pelas regras *fuzzy* que os resultados providos pelo raciocinador fuzzyDL. Em experimentos envolvendo o domínio de risco químico em alimentos, baseado em dados reais, os resultados de HyFOM e FT-FIS também são mais precisos que os resultados de fuzzyDL, em comparação com valores de referência disponíveis nesse domínio.

**Palavras-chave:** Representação de Conhecimento, Ontologia *Fuzzy*, Regras *Fuzzy*, Sistema de Inferência *Fuzzy*, Arquitetura de Raciocínio

## ABSTRACT

In Computer Science, ontologies are used for knowledge representation in a number of applications, aiming to structure and handle domain semantics through models shared by humans and computational systems. Although traditional ontologies model semantic information and support reasoning tasks, they are based on a formalism which is less suitable to express the vagueness inherent in real-world phenomena and human language. To address this issue, many proposals investigate how traditional ontologies can be extended by incorporating concepts from fuzzy sets and fuzzy logic, resulting in fuzzy ontologies. In special, combining the formalism from fuzzy ontologies with fuzzy rule-based reasoning, which has been successfully applied in the context of fuzzy inference systems, can lead to more expressive inferences involving imprecision. In this sense, this doctoral thesis aims at exploring the integration of fuzzy ontology reasoning with fuzzy inference systems, resulting in the definition and the development of two approaches: HyFOM (*Hybrid integration of Fuzzy Ontology and Mamdani reasoning*) and FT-FIS (*Fuzzy Tableau and Fuzzy Inference System*). HyFOM is based on a hybrid architecture combining reasoners for ontologies, fuzzy ontologies and fuzzy inference systems, focusing on the interaction among its independent components. FT-FIS defines an interface between a fuzzy *tableau*-based algorithm and a fuzzy inference system, including the *fuzzyRuleReasoning* predicate that allows fuzzy rule-based reasoning to be invoked whenever necessary for fuzzy ontology reasoning tasks. The main contribution of HyFOM and FT-FIS comes from their reasoning architectures, which combine flexibility in terms of fuzzy rule semantics with the collaboration between inferences from both types of reasoning. Experiments regarding the recommendation of touristic attractions, based on synthetic data, revealed that HyFOM and FT-FIS provide integrated inferences, in addition to a more expressive approximation of the relation defined by fuzzy rules than the results from the fuzzyDL reasoner. In experiments involving the evaluation of chemical risk in food samples, based on real data, results obtained by HyFOM and FT-FIS are also more precise than fuzzyDL results, in comparison with reference values available in this domain.

**Keywords:** Knowledge Representation, Fuzzy Ontology, Fuzzy Rules, Fuzzy Inference System, Reasoning Architecture

## LISTA DE FIGURAS

2.1	Sistema de inferência <i>fuzzy</i> (SIF). . . . .	23
2.2	Representação gráfica de funções de pertinência. . . . .	26
2.3	Variável linguística temperatura (Figura adaptada de (PEDRYCZ; GOMIDE, 1998)). . . . .	29
2.4	Sistema de inferência <i>fuzzy</i> com entradas e saídas numéricas. . . . .	36
3.1	Arquitetura para raciocínio proposta por Pan (2007). . . . .	50
3.2	Arquitetura para raciocínio proposta por Wang et al. (2008). . . . .	63
5.1	Arquitetura de HyFOM para integração de raciocínio de ontologias <i>fuzzy</i> e sistemas de inferência <i>fuzzy</i> . . . . .	83
5.2	Arquitetura de raciocínio de FT-FIS, no qual módulos em cinza representam o foco da extensão de FT-FIS para integrar o raciocínio de regras <i>fuzzy</i> . . . . .	97
6.1	Regras <i>fuzzy</i> utilizadas nos experimentos para recomendação de atrações turísticas. . . . .	119
6.2	Gráficos referentes aos resultados de HyFOM/FT-FIS. . . . .	124
6.3	Gráficos com os resultados de fuzzyDL. . . . .	127
6.4	Gráficos referentes aos resultados de FT-FIS. . . . .	132
6.5	Gráficos com os resultados de fuzzyDL para consulta 2 e entradas numéricas. . . . .	135
6.6	Risco agregado inferido por HyFOM e FT-FIS com método de Mamdani, combinado com diferentes métodos de defuzificação. . . . .	151
6.7	Risco agregado inferido por HyFOM e FT-FIS com método de Larsen, combinado com diferentes métodos de defuzificação. . . . .	151
6.8	Risco agregado inferido pelas abordagens de fuzzyDL. . . . .	153

6.9	Resultados de FT-FIS e fuzzyDL para consulta 2 e entradas numéricas. . . . .	156
-----	--	-----

## LISTA DE TABELAS

3.1	Sintaxe e semântica de $\mathcal{ALC}$ (BAADER, 2003). . . . .	43
3.2	Regras de expansão <i>tableau</i> para $\mathcal{ALC}$ (extraída de (HORROCKS, 2003)). . . . .	46
3.3	Sintaxe e semântica de expressões de tipos de dados. . . . .	48
3.4	Regras- $G$ para tratar expressões de tipos de dados (PAN, 2007). . . . .	50
3.5	Construtores de conceito estendidos de <i>fuzzy</i> $\mathcal{ALC}$ (STRACCIA, 2001). . . . .	53
3.6	Situações que definem pares conjugados (STRACCIA, 2001). . . . .	57
3.7	Regras de expansão para <i>fuzzy</i> $\mathcal{ALC}$ extraídas de (STRACCIA, 2001). . . . .	58
3.8	Sintaxe e semântica de expressões de tipos de dados <i>fuzzy</i> . . . . .	61
3.9	Regras- $G$ para tratar expressões de tipos de dados <i>fuzzy</i> (WANG; MA, 2008; WANG et al., 2008). . . . .	64
4.1	Características analisadas nos principais trabalhos correlatos. . . . .	75
5.1	Regras <i>fuzzy</i> para obter o preço do frete em função de outras propriedades. . . . .	81
5.2	Predicados embutidos de FT-FIS no domínio dos números reais. . . . .	92
5.3	Restrições de grau aplicadas sobre um conjunto <i>fuzzy</i> $FS$ em FT-FIS. . . . .	106
5.4	Sintaxe para definir e referenciar predicados <i>fuzzy</i> em FT-FIS. . . . .	109
6.1	Ontologia <i>fuzzy</i> sobre atrações turísticas utilizada em HyFOM e fuzzyDL. . . . .	117
6.2	Ontologia <i>fuzzy</i> sobre atrações turísticas utilizada em FT-FIS. . . . .	118
6.3	Regras implicativas <i>fuzzy</i> em fuzzyDL. . . . .	120
6.4	Regras <i>fuzzy</i> sob a semântica de Mamdani em fuzzyDL. . . . .	121
6.5	Resultados de HyFOM/FT-FIS para consulta 1 e entradas numéricas. . . . .	123
6.6	Resultados de fuzzyDL para consulta 1 e entradas numéricas. . . . .	126

6.7	Graus de pertinência obtidos a partir da satisfação inferida por HyFOM para consulta 2 e entradas numéricas. . . . .	130
6.8	Resultados de FT-FIS para consulta 2 e entradas numéricas. . . . .	131
6.9	Resultados de fuzzyDL para consulta 2 e entradas numéricas. . . . .	134
6.10	Resultados de HyFOM/FT-FIS para consulta 1 e entradas <i>fuzzy</i> . . . . .	138
6.11	Resultados de fuzzyDL para consulta 1 e entradas <i>fuzzy</i> . . . . .	139
6.12	Graus de pertinência obtidos a partir da satisfação inferida por HyFOM para consulta 2 e entradas <i>fuzzy</i> . . . . .	141
6.13	Resultados de FT-FIS para consulta 2 e entradas <i>fuzzy</i> . . . . .	142
6.14	Resultados de fuzzyDL para consulta 2 e entradas <i>fuzzy</i> . . . . .	143
6.15	Ontologia <i>fuzzy</i> sobre risco químico de analitos utilizada em HyFOM e fuzzyDL.	146
6.16	Ontologia <i>fuzzy</i> sobre risco químico de analitos utilizada em FT-FIS. . . . .	147
6.17	Regras <i>fuzzy</i> para inferir risco agregado do analito. . . . .	149
6.18	Erro médio quadrático de HyFOM/FT-FIS e fuzzyDL. . . . .	153

# SUMÁRIO

<b>CAPÍTULO 1 – INTRODUÇÃO</b>	<b>16</b>
1.1 Contexto e Motivação . . . . .	16
1.2 Objetivos . . . . .	18
1.3 Contribuições . . . . .	19
1.4 Organização da Tese . . . . .	20
<b>I Fundamentação Teórica e Trabalhos Correlatos</b>	<b>21</b>
<b>CAPÍTULO 2 – SISTEMAS DE INFERÊNCIA FUZZY</b>	<b>22</b>
2.1 Considerações Iniciais . . . . .	22
2.2 Conjuntos <i>Fuzzy</i> . . . . .	23
2.3 Variáveis Linguísticas . . . . .	28
2.4 Regras <i>Fuzzy</i> . . . . .	30
2.4.1 Sintaxe de Regras <i>Fuzzy</i> . . . . .	30
2.4.2 Semântica de Regras <i>Fuzzy</i> . . . . .	31
2.5 Raciocínio Aproximado com Regras <i>Fuzzy</i> . . . . .	32
2.5.1 Regra da Inferência Composicional . . . . .	33
2.5.2 Método da Interpolação . . . . .	34
2.5.3 Métodos para Entradas e Saídas Numéricas . . . . .	35
2.6 Considerações Finais . . . . .	39

<b>CAPÍTULO 3 – ONTOLOGIAS FUZZY</b>	<b>40</b>
3.1 Considerações Iniciais . . . . .	40
3.2 Ontologias: Definição e Caracterização . . . . .	40
3.3 Lógicas de Descrição . . . . .	42
3.3.1 Sintaxe e Semântica de $\mathcal{ALC}$ . . . . .	42
3.3.2 Inferências Típicas de DLs . . . . .	44
3.3.3 Raciocínio Baseado em <i>Tableau</i> . . . . .	45
3.3.4 Tipos de Dados Concretos em DLs . . . . .	47
3.4 Ontologias <i>Fuzzy</i> : Definição e Caracterização . . . . .	51
3.5 Lógicas de Descrição <i>Fuzzy</i> . . . . .	52
3.5.1 Sintaxe e Semântica de <i>Fuzzy</i> $\mathcal{ALC}$ . . . . .	53
3.5.2 Inferências Típicas de <i>Fuzzy</i> DLs . . . . .	54
3.5.3 Raciocínio Baseado em <i>Tableau Fuzzy</i> . . . . .	56
3.5.4 Tipos de Dados Concretos <i>Fuzzy</i> . . . . .	59
3.6 Considerações Finais . . . . .	65
<b>CAPÍTULO 4 – TRABALHOS CORRELATOS COMBINANDO ONTOLOGIAS FUZZY E REGRAS FUZZY</b>	<b>66</b>
4.1 Considerações Iniciais . . . . .	66
4.2 Ontologias <i>Fuzzy</i> e Regras <i>Fuzzy</i> Envolvendo Termos Linguísticos . . . . .	67
4.3 Ontologias e Sistemas de Inferência <i>Fuzzy</i> . . . . .	70
4.4 Discussão sobre Abordagens Combinando Ontologias e Regras <i>Fuzzy</i> . . . . .	73
4.5 Considerações Finais . . . . .	76
<b>II Trabalhos Desenvolvidos</b>	<b>77</b>
<b>CAPÍTULO 5 – ABORDAGENS PARA INTEGRAÇÃO DE RACIOCÍNIO DE ON- TOLOGIAS FUZZY E SISTEMAS DE INFERÊNCIA FUZZY</b>	<b>78</b>



5.1	Considerações Iniciais . . . . .	78
5.2	Contexto de Integração de Ontologias <i>Fuzzy</i> e Sistemas de Inferência <i>Fuzzy</i> . .	79
5.3	<i>HyFOM: Hybrid integration of Fuzzy Ontology and Mamdani reasoning</i> . . . .	82
5.3.1	Arquitetura e Componentes de HyFOM . . . . .	83
5.3.2	Abordagem de Integração . . . . .	84
5.3.3	Protótipo HyFOM . . . . .	87
5.4	<i>FT-FIS: Fuzzy Tableau and Fuzzy Inference System</i> . . . . .	89
5.4.1	Representação de Conhecimento . . . . .	89
5.4.2	Arquitetura de Raciocínio de FT-FIS . . . . .	97
5.4.3	Abordagem de Integração . . . . .	99
5.4.4	Protótipo FT-FIS . . . . .	108
5.5	Considerações Finais . . . . .	112
 <b>CAPÍTULO 6 – AVALIAÇÃO DAS ABORDAGENS DESENVOLVIDAS</b>		<b>113</b>
6.1	Considerações Iniciais . . . . .	113
6.2	Recomendação de Atrações Turísticas . . . . .	115
6.2.1	Ontologia <i>Fuzzy</i> e Regras <i>Fuzzy</i> . . . . .	116
6.2.2	Experimentos com Entradas Numéricas . . . . .	121
6.2.3	Experimentos com Entradas <i>Fuzzy</i> . . . . .	134
6.3	Risco Químico em Alimentos . . . . .	144
6.3.1	Ontologia <i>Fuzzy</i> e Regras <i>Fuzzy</i> . . . . .	144
6.3.2	Experimentos com Entradas Numéricas . . . . .	149
6.3.3	Experimentos com Entradas <i>Fuzzy</i> . . . . .	156
6.4	Considerações Finais . . . . .	157
 <b>CAPÍTULO 7 – CONCLUSÕES E TRABALHOS FUTUROS</b>		<b>159</b>
7.1	Considerações Iniciais . . . . .	159

7.2	Contribuições . . . . .	160
7.3	Limitações e Trabalhos Futuros . . . . .	163
	<b>REFERÊNCIAS BIBLIOGRÁFICAS</b>	<b>167</b>
	<b>LISTA DE SIGLAS E ABREVIATURAS</b>	<b>173</b>

# Capítulo 1

## INTRODUÇÃO

---

---

### 1.1 Contexto e Motivação

Ontologias têm sido empregadas para representação de conhecimento em diversas aplicações com o intuito de expressar estruturas semânticas que podem ser compartilhadas e processadas por pessoas e sistemas computacionais. No contexto da Ciência da Computação, uma ontologia define um conjunto de primitivas representacionais como conceitos, atributos, relacionamentos, indivíduos e restrições, que modelam um domínio de conhecimento ou discurso (GRUBER, 2009).

Segundo Uschold e Grüninger (2004), uma gama de especificações podem ser consideradas ontologias: desde simples glossários até rigorosos formalismos lógicos. Existem representações simplificadas chamadas de “ontologias leves” (*lightweight ontologies*), contendo apenas termos com pouca ou nenhuma especificação de significado. Conforme o grau de formalização aumenta e a especificação do significado se torna mais precisa, reduz-se a ambiguidade. Em especial, ontologias formais possibilitam a representação de conhecimento como também oferecem suporte para raciocínio automatizado, sendo fundamentadas por formalismos de representação de conhecimento como a família das Lógicas de Descrição (NARDI; BRACHMAN, 2003; BAADER; NUTT, 2003). As tarefas de raciocínio em ontologias permitem verificar a consistência do conhecimento modelado e revelar associações implícitas a partir de conceitos definidos explicitamente.

Embora as ontologias baseadas em Lógicas de Descrição sejam expressivas e ofereçam suporte para raciocínio, existem limitações que restringem a representação do conhecimento de alguns domínios. De modo geral, como os formalismos que fundamentam ontologias convencionais baseiam-se na lógica clássica, elas são capazes de representar informações apenas

de modo preciso ou exato. Assim, é difícil expressar de forma apropriada conceitos como *homem jovem*, *produto caro* ou *temperatura quente*, para os quais não é possível obter uma definição clara e precisa (STRACCIA, 2006b) (LUKASIEWICZ; STRACCIA, 2008). Portanto, tais ontologias são menos adequadas para modelar domínios que envolvem conceitos de caráter vago ou impreciso.

Em virtude dessa limitação, existe a necessidade de estender ontologias para capturar a imprecisão presente na linguagem, na interpretação e no raciocínio humanos com relação a alguns fenômenos do mundo real. Dentre as teorias que permitem modelar conhecimento não exato, a teoria de conjuntos *fuzzy* e a lógica *fuzzy*, definidas por Zadeh (1965, 1975), estendem conceitos de conjuntos e lógica clássicos com o objetivo de expressar informações que não são totalmente verdadeiras, nem totalmente falsas, mas que possuem um grau de verdade associado (KLIR; YUAN, 1995; PEDRYCZ; GOMIDE, 1998). Neste sentido, a pesquisa em ontologias *fuzzy* visa tratar a representação e o processamento de informações vagas ou imprecisas em ontologias por meio da incorporação de conceitos da teoria de conjuntos *fuzzy* e da lógica *fuzzy*.

Dentre as extensões de ontologias para tratamento de imprecisão, a representação e o processamento de variáveis e termos linguísticos permitem expressar uma transição gradual entre diferentes estados de uma variável, modelados por conjuntos *fuzzy*. Em geral, as variáveis linguísticas são representadas em ontologias *fuzzy* por meio de atributos ou propriedades concretas definidas sobre um domínio numérico, onde é possível definir conjuntos *fuzzy* a partir de funções de pertinência parametrizadas que expressam a transição gradual entre os estados. Os termos linguísticos são representados como predicados *fuzzy* ou como conceitos concretos *fuzzy* (BOBILLO; STRACCIA, 2008), cada qual com sua respectiva função de pertinência, que pode ser do tipo triangular, trapezoidal, entre outros.

O processamento de diferentes variáveis linguísticas combinadas sob a forma de regras *fuzzy*, assim como ocorre em sistemas de inferência *fuzzy* (SIF), pode proporcionar conhecimento relevante para ontologias *fuzzy*, com relação ao raciocínio envolvendo propriedades numéricas vinculadas a termos imprecisos. De acordo com Guillaume e Charnomordic (2012), uma das razões do sucesso dos SIF se deve à habilidade dos sistemas *fuzzy* de incorporar conhecimento de especialistas e suas nuances, capturando a correspondência entre elementos numéricos e simbólicos de uma forma interpretável aos humanos. Ao combinar ontologias *fuzzy* com regras contendo variáveis e termos linguísticos, é possível aplicar métodos de raciocínio aproximado de SIF para realizar inferências associadas a valores de propriedades numéricas.

De fato, existem aplicações que demandam esse tipo de raciocínio integrado, tais como

sistemas de suporte a decisão e de recomendação baseados em ontologias *fuzzy* (LOIA, 2011). Nessas aplicações, geralmente, existem propriedades numéricas cujos valores podem ser explícitos ou então resultar do raciocínio envolvendo o conhecimento representado na ontologia *fuzzy*. Essas propriedades, por sua vez, podem ser combinadas para inferir o valor de outras propriedades numéricas, configurando uma relação que pode ser representada por meio de regras *fuzzy*. Os resultados inferidos a partir das regras *fuzzy* podem ser utilizados diretamente para classificar as respostas mais relevantes ou então determinar a pertinência para outros conceitos da ontologia *fuzzy*.

Segundo Loia (2011), o projeto de sistemas avançados de suporte a decisão requer que o conhecimento impreciso esteja embutido em um motor de inferência de alto nível, como os raciocinadores de ontologias *fuzzy*, que também seja capaz de inferir novo conhecimento ao explorar métodos consagrados na inteligência computacional, como no caso dos métodos de raciocínio de SIF. Diante do exposto, o tratamento de imprecisão, integrando o raciocínio de ontologias *fuzzy* e de SIF, pode resultar em informação significativa para aplicações voltadas para o apoio à tomada de decisão.

## 1.2 Objetivos

Considerando este contexto, o objetivo desta tese de doutorado é abordar a integração do raciocínio de ontologias *fuzzy* e de sistemas de inferência *fuzzy*, a fim de prover inferências expressivas com foco no tratamento de imprecisão envolvendo propriedades numéricas. Especificamente, os principais pontos que nortearam a pesquisa foram:

- **Semântica das inferências baseadas em regras *fuzzy*:** os métodos de raciocínio aproximado, operações *fuzzy* e os métodos de defuzzificação disponibilizados devem corresponder à semântica do domínio para que as inferências baseadas em regras *fuzzy* sejam significativas e atendam aos requisitos das aplicações. Neste sentido, arquiteturas de integração flexíveis, que permitam instanciar diferentes implementações de SIF, devem ser consideradas para oferecer a semântica mais apropriada para o domínio;
- **Raciocínio de SIF contribuindo para o raciocínio de ontologias *fuzzy*:** os resultados do raciocínio de SIF podem ser relevantes para algumas tarefas de raciocínio da ontologia *fuzzy*. Por exemplo, a verificação de pertinência de um indivíduo para um conceito abstrato, definido a partir de restrições sobre propriedades numéricas, pode depender de valores inferidos por meio de regras *fuzzy*. No decorrer da tese, esta característica da abordagem de integração é referenciada como integração  $SIF \rightarrow ontologia\ fuzzy$ ;

- **Raciocínio de ontologias *fuzzy* contribuindo para o raciocínio de SIF:** inferências de ontologias *fuzzy* podem revelar conhecimento implícito associado a valores de propriedades numéricas que correspondem aos valores de entrada para o SIF. Por exemplo, definições de conceitos podem restringir os valores de determinadas propriedades numéricas por meio de predicados *fuzzy* (termos linguísticos), que devem ser considerados para os indivíduos desses conceitos. No decorrer da tese, esta característica da abordagem de integração é referenciada como integração *ontologia fuzzy*  $\rightarrow$  SIF.

### 1.3 Contribuições

A principal contribuição desta tese está na definição e no desenvolvimento de abordagens capazes de integrar raciocínio de ontologias *fuzzy* e SIF, atendendo aos requisitos referentes à semântica do raciocínio baseado em regras *fuzzy* e à colaboração entre as inferências de ambos tipos de raciocínio (SIF  $\rightarrow$  *ontologia fuzzy* e *ontologia fuzzy*  $\rightarrow$  SIF). A combinação dessas características é a inovação da pesquisa desenvolvida, visto que trabalhos correlatos que disponibilizam métodos específicos de raciocínio baseado em regras *fuzzy* não são flexíveis para atender aos requisitos de semântica dos domínios; enquanto outros que oferecem interoperabilidade de SIF não abordam o raciocínio de ontologias *fuzzy* formais, portanto não tratam a integração *ontologia fuzzy*  $\rightarrow$  SIF, conforme discutido no Capítulo 4.

Neste contexto, duas arquiteturas de integração de raciocínio de ontologias *fuzzy* e SIF foram definidas e desenvolvidas:

- **HyFOM**, cujo nome foi inspirado em *Hybrid integration of Fuzzy Ontology and Mamdani reasoning*, pois sua primeira versão oferecia suporte a sistemas de inferência de Mamdani. Posteriormente, foi estendido para ser utilizado com outros tipos de SIF. HyFOM foi concebido e desenvolvido com a cooperação do Prof. Dr. Marek Reformat da Universidade de Alberta, durante o estágio de doutorado sanduíche. Baseado em uma arquitetura híbrida que combina implementações existentes de raciocinadores de ontologias e de SIF, HyFOM permite que diferentes SIF sejam instanciados, com o suporte da *Fuzzy Markup Language* (FML) (ACAMPORA; LOIA, 2005), uma linguagem de especificação de SIF com foco em interoperabilidade. HyFOM possui como contribuição principal a interface e a interação entre seus componentes para a obtenção de inferências integradas de acordo com os objetivos mencionados anteriormente;
- **FT-FIS** (*Fuzzy Tableau and Fuzzy Inference System*), que combina um algoritmo baseado em *tableau fuzzy* com SIF por meio de uma arquitetura flexível inspirada nos trabalhos

de Pan (2007), Wang e Ma (2008), que foi estendida para tratar os requisitos específicos da integração com SIF. O tratamento de imprecisão em FT-FIS é mais expressivo que em HyFOM, pois são explorados resultados do SIF tanto como valores numéricos defuzificados quanto como conjuntos *fuzzy* que podem ser restritos por predicados *fuzzy* modelados na ontologia. A principal contribuição de FT-FIS está na interface desenvolvida entre o algoritmo baseado em *tableau fuzzy* e o SIF, incluindo a definição do predicado *fuzzyRuleReasoning* que permite invocar o raciocínio baseado em regras *fuzzy* quando for necessário para as tarefas de raciocínio da ontologia *fuzzy*.

## 1.4 Organização da Tese

O restante desta tese de doutorado está organizada da seguinte forma.

A Parte I apresenta a fundamentação teórica e os trabalhos correlatos:

- O Capítulo 2 aborda uma breve introdução sobre a teoria de conjuntos *fuzzy* e os principais conceitos relacionados aos sistemas de inferência *fuzzy*, seus componentes e métodos de raciocínio aproximado;
- No Capítulo 3, são apresentados conceitos básicos sobre ontologias e ontologias *fuzzy*, incluindo definição e caracterização, formalismo lógico baseado nas Lógicas de Descrição, algoritmo de raciocínio baseado em *tableau* e as respectivas extensões *fuzzy*;
- O Capítulo 4 analisa abordagens da literatura que investigam a combinação de ontologias *fuzzy* e regras *fuzzy*, com foco em regras contendo variáveis linguísticas associadas a propriedades numéricas modeladas na ontologia *fuzzy*.

A Parte II apresenta os trabalhos desenvolvidos durante o doutorado:

- O Capítulo 5 descreve as abordagens desenvolvidas, HyFOM e FT-FIS, para integração de raciocínio de ontologias *fuzzy* e de SIF, detalhando seus requisitos, arquiteturas e estratégias de integração;
- O Capítulo 6 descreve os experimentos realizados com HyFOM e FT-FIS considerando dois domínios de aplicação, assim como a análise dos resultados obtidos frente aos resultados de abordagens correlatas;
- No Capítulo 7, são apresentadas as conclusões, principais contribuições e limitações das abordagens desenvolvidas, além de direções para trabalhos futuros.

# **Parte I**

## **Fundamentação Teórica e Trabalhos Correlatos**



# Capítulo 2

## SISTEMAS DE INFERÊNCIA *Fuzzy*

---

---

### 2.1 Considerações Iniciais

Para representar conceitos e relacionamentos de um determinado domínio, nem sempre as informações são caracterizadas como exatas ou precisas. Dentre as abordagens que permitem modelar conhecimento não exato, a teoria de conjuntos *fuzzy*, proposta por Zadeh (1965), estende conceitos de conjuntos clássicos visando capturar a imprecisão presente nos fenômenos do mundo real e na linguagem humana.

Uma das aplicações mais conhecidas dos princípios de conjuntos *fuzzy* e lógica *fuzzy* são os *sistemas baseados em regras fuzzy* (SBRF) ou *sistemas de inferência fuzzy* (SIF). De acordo com Guillaume e Charnomordic (2012), uma das razões do sucesso dos SIF se deve à habilidade dos sistemas *fuzzy* de incorporar conhecimento de especialistas humanos e suas nuances, assim como de expressar o comportamento do sistema de uma forma interpretável aos humanos. Ainda segundo Guillaume e Charnomordic (2012), SIF são apropriados para cooperação humano-máquina, pois proveem um *framework* unificado (funções de pertinência, regras, operadores, mecanismos de inferência) que permite capturar a correspondência entre elementos numéricos e simbólicos. Um sistema de inferência *fuzzy* típico é ilustrado na Figura 2.1.

De modo geral, SIF são constituídos por uma base de conhecimento, que contém informações sobre o domínio de aplicação, e por um motor de inferência que deriva conclusões sobre a base de conhecimento, gerando saídas a partir das entradas fornecidas ao sistema. A base de conhecimento é composta por uma base de dados, que contém a definição dos conjuntos *fuzzy* usados no sistema, e por uma base de regras contendo um conjunto de regras *fuzzy* paralelas.

Este capítulo visa apresentar os principais conceitos relacionados aos sistemas de inferência *fuzzy* e seus componentes. As definições apresentadas podem ser encontradas de forma integral

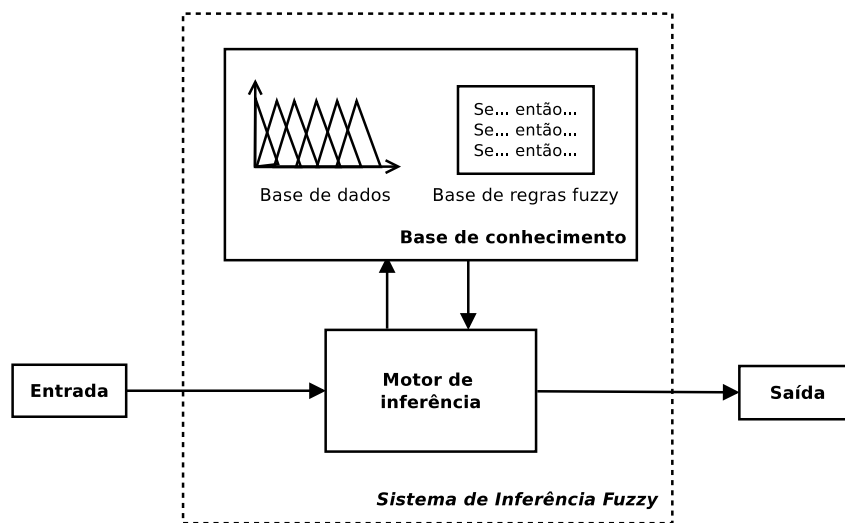


Figura 2.1: Sistema de inferência *fuzzy* (SIF).

nos trabalhos de Klir e Yuan (1995), Pedrycz e Gomide (1998) e Nicoletti e Camargo (2004). Primeiramente, a Seção 2.2 apresenta a definição de conjunto *fuzzy*, tipos de funções de pertinência e operações básicas. Variáveis linguísticas, utilizadas para representar a base de dados de sistemas de inferência *fuzzy*, são definidas na Seção 2.3. Em sequência, a Seção 2.4 descreve a sintaxe e a semântica de regras *fuzzy* que compõem a base de regras *fuzzy*. Por fim, a Seção 2.5 apresenta métodos de raciocínio aproximado frequentemente utilizados para modelar o motor de inferência, assim como sistemas de inferência *fuzzy* voltados a problemas específicos.

## 2.2 Conjuntos *Fuzzy*

De modo geral, o conceito de *conjunto* está associado à organização, categorização, generalização de conhecimento sobre objetos que possuem características em comum. Na teoria de conjuntos clássicos, também chamados de conjuntos *crisp*, objetos pertencem ou não a um conjunto, sendo que a decisão para a classificação de objetos pode ser representada por meio de uma função, denominada função característica. Dado um conjunto de objetos  $X$ , denominado *universo de discurso* ou *conjunto universo*, um conjunto  $A \subseteq X$  é definido por uma função característica  $A : X \rightarrow \{0, 1\}$  (Equação 2.1) que associa cada elemento  $x \in X$  ao valor 1, caso pertença ao conjunto  $A$ , e ao valor 0, caso não pertença (PEDRYCZ; GOMIDE, 1998).

$$A(x) = \begin{cases} 1, & \text{se } x \in A \\ 0, & \text{se } x \notin A \end{cases} \quad (2.1)$$

No entanto, existem situações em que essa classificação binária não é adequada, quando é

necessário representar categorias em que os objetos do mundo real não possuem limites bem definidos. Por exemplo, conceitos como idade *jovem*, temperatura *alta*, velocidade *baixa*, carro *grande*, não possuem uma definição exata de pertinência absoluta e não pertinência absoluta. Conceitos desse tipo demandam uma representação mais flexível, que admita valores intermediários de pertinência aos conjuntos, não restrita aos valores  $\{0, 1\}$ .

Neste sentido, a teoria de conjuntos *fuzzy*, proposta por Zadeh (1965), estabelece a generalização da função característica dos conjuntos clássicos, de modo que elementos pertençam a um conjunto com um determinado grau dentro de um intervalo, geralmente  $[0, 1]$ . Essa função generalizada é denominada *função de pertinência*, e o conjunto definido por ela é denominado *conjunto fuzzy* (KLIR; YUAN, 1995). Formalmente, a função de pertinência de um conjunto *fuzzy*  $A$  é denotada por  $A : X \rightarrow [0, 1]$ , que associa elementos do conjunto universo  $X$  a números reais no intervalo  $[0, 1]$ . Desta forma, é possível representar uma pertinência parcial por um valor entre os valores 0 e 1, que correspondem à exclusão total e à pertinência total de um objeto a um conjunto, respectivamente. Os valores de pertinência expressam os graus com que cada objeto é compatível com as características que definem o conjunto (PEDRYCZ; GOMIDE, 1998).

A princípio qualquer função da forma  $A : X \rightarrow [0, 1]$  descreve uma função de pertinência associada a um conjunto *fuzzy*  $A$ . Entretanto, em algumas situações, o significado capturado por um conjunto *fuzzy* não é tão sensível a variações no formato do conjunto, sendo conveniente utilizar funções mais simples (PEDRYCZ; GOMIDE, 1998). Assim, em aplicações práticas, conjuntos *fuzzy* são geralmente representados por famílias de funções parametrizadas. Segundo Pedrycz e Gomide (1998), algumas das funções de pertinência mais comuns são:

- *Função Triangular* (Equação 2.2 e Figura 2.2a): definida pelos parâmetros  $a$ ,  $m$  e  $b$ , com  $a \leq m \leq b$ :

$$A(x) = \begin{cases} 0, & \text{se } x \leq a \\ \frac{x-a}{m-a}, & \text{se } x \in (a, m) \\ 1, & \text{se } x = m \\ \frac{b-x}{b-m}, & \text{se } x \in (m, b) \\ 0, & \text{se } x \geq b \end{cases} \quad (2.2)$$

Outra notação analítica:  $A(x; a, m, b) = \max\left\{\min\left[\frac{x-a}{m-a}, \frac{b-x}{b-m}\right], 0\right\}$ .

- *Função Trapezoidal* (Equação 2.3 e Figura 2.2b): definida pelos parâmetros  $a$ ,  $m$ ,  $n$  e  $b$ ,

com  $a \leq m < n \leq b$ :

$$A(x) = \begin{cases} 0, & \text{se } x \leq a \\ \frac{x-a}{m-a}, & \text{se } x \in (a, m) \\ 1, & \text{se } x \in [m, n] \\ \frac{b-x}{b-n}, & \text{se } x \in (n, b) \\ 0, & \text{se } x \geq b \end{cases} \quad (2.3)$$

Outra notação analítica:  $A(x; a, m, n, b) = \max\left\{\min\left[\frac{x-a}{m-a}, 1, \frac{b-x}{b-n}\right], 0\right\}$ . Alguns autores consideram tipos específicos de funções trapezoidais, como as funções do tipo *left-shoulder* nas quais  $a = m$  e as funções do tipo *right-shoulder* nas quais  $n = b$ .

- *Função S* (Equação 2.4 e Figura 2.2c): definida pelos parâmetros  $a$  e  $b$ , com o ponto  $m = (a + b)/2$  conhecido como o cruzamento da função S:

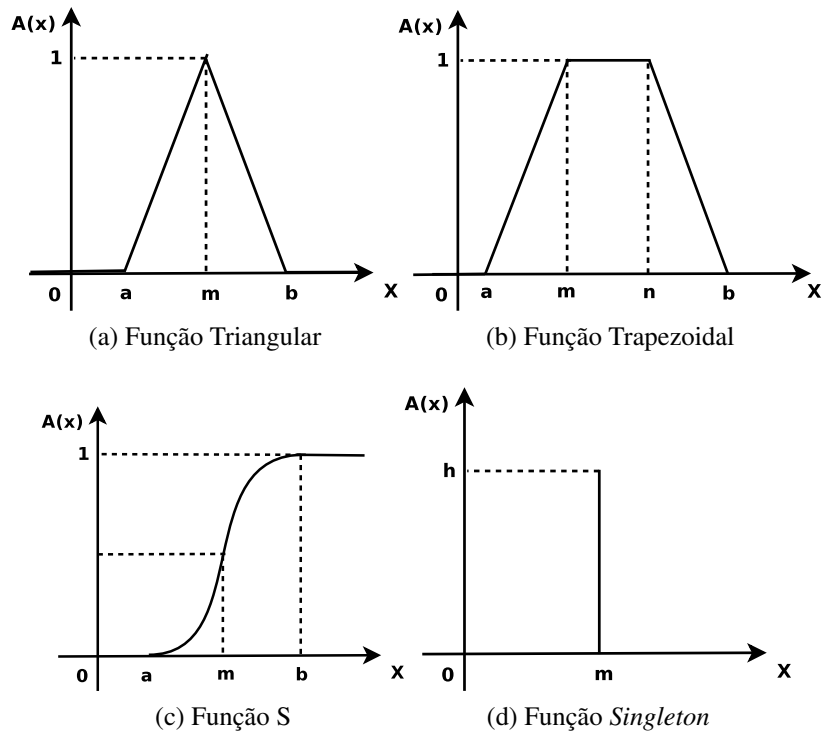
$$A(x) = \begin{cases} 0, & \text{se } x \leq a \\ 2\left(\frac{x-a}{b-a}\right)^2, & \text{se } x \in (a, m) \\ 1 - 2\left(\frac{x-b}{b-a}\right)^2, & \text{se } x \in (m, b) \\ 1, & \text{se } x > b \end{cases} \quad (2.4)$$

- *Função Singleton* (Equação 2.5 e Figura 2.2d): definida pelos parâmetros  $m$  e  $h$ , com  $h \leq 1$ :

$$A(x) = \begin{cases} h, & \text{se } x = m \\ 0, & \text{se } x \neq m \end{cases} \quad (2.5)$$

O conceito de relação na teoria de conjuntos clássicos representa a presença ou ausência de associação, interação ou interconexão entre os elementos de um ou mais conjuntos. Na teoria de conjuntos *fuzzy*, as relações podem ser generalizadas pela atribuição de um valor no intervalo  $[0, 1]$ , que representa a intensidade com que os elementos estão relacionados entre si. Assim, a função característica de uma relação clássica pode ser generalizada para permitir que a associação entre elementos tenha grau de pertinência para uma relação. Pode-se dizer que relações *fuzzy* são conjuntos *fuzzy* definidos em conjuntos universos que são produtos cartesianos. Desta forma, uma *relação fuzzy n-ária* ou multidimensional é um conjunto *fuzzy* definido no produto cartesiano dos conjuntos  $X_1, X_2, \dots, X_n$ , no qual as *n-uplas*  $\langle x_1, x_2, \dots, x_n \rangle$  podem ter graus de pertinência que indicam a força da relação entre os elementos da *n-upla* (KLIR; YUAN, 1995). Formalmente, uma relação *fuzzy* multidimensional  $R$  é denotada por:

$$R : X_1 \times X_2 \times \dots \times X_n \rightarrow [0, 1] \quad (2.6)$$



**Figura 2.2:** Representação gráfica de funções de pertinência.

Operações sobre conjuntos *fuzzy* permitem modificar, combinar e comparar conceitos representados por conjuntos *fuzzy*. Existem diversas generalizações das operações de complemento, interseção e união *fuzzy*; para cada uma dessas operações há uma classe de funções com propriedades específicas: *complementos fuzzy*, *t-normas* para interseções *fuzzy* e *t-conormas* (também conhecidas como *s-normas*) para uniões *fuzzy*.

### Complemento *Fuzzy*

Seja  $A$  um conjunto *fuzzy* definido sobre  $X$ . O complemento *fuzzy* de  $A$  é denotado como  $cA$ , onde  $cA(x)$  pode ser interpretado como o grau com que  $x$  não pertence a  $A$ .  $cA$  é definido por uma função  $c : [0, 1] \rightarrow [0, 1]$  que atribui um valor  $c(A(x))$  para cada grau de pertinência  $A(x)$ , para todo  $x \in X$ . Segundo Klir e Yuan (1995), um dos complementos *fuzzy* frequentemente utilizados em aplicações é o complemento padrão:

$$c(a) = 1 - a \quad (2.7)$$

### Interseção *Fuzzy*

Sejam  $A$  e  $B$  conjuntos *fuzzy* definidos sobre o conjunto universo  $X$ . A interseção *fuzzy* de  $A$  e  $B$ , denotada por  $(A \cap B)(x)$ , corresponde à função de pertinência do conjunto resultante da interseção de  $A$  e  $B$ . A classe de funções *t-normas* (*triangular norm*) possuem propriedades que

garantem que os conjuntos *fuzzy* resultantes da aplicação dessas funções sejam significativos como interseções *fuzzy* (KLIR; YUAN, 1995; NICOLETTI; CAMARGO, 2004). Uma t-norma é uma operação binária  $i : [0, 1]^2 \rightarrow [0, 1]$  que atribui um valor  $(A \cap B)(x)$  considerando os graus de pertinência  $A(x)$  e  $B(x)$ , para todo  $x \in X$ . Segundo Klir e Yuan (1995), algumas t-normas que são frequentemente utilizadas como interseções *fuzzy*:

- *Interseção padrão:*

$$i(a, b) = \min(a, b) \quad (2.8)$$

- *Produto algébrico:*

$$i(a, b) = ab \quad (2.9)$$

- *Diferença limitada:*

$$i(a, b) = \max(0, a + b - 1) \quad (2.10)$$

- *Interseção drástica:*

$$i(a, b) = \begin{cases} a, & \text{quando } b = 1 \\ b, & \text{quando } a = 1 \\ 0, & \text{caso contrário} \end{cases} \quad (2.11)$$

### União Fuzzy

Sejam  $A$  e  $B$  conjuntos *fuzzy* definidos sobre o conjunto universo  $X$ . A união *fuzzy* de  $A$  e  $B$ , denotada por  $(A \cup B)(x)$ , corresponde à função de pertinência do conjunto resultante da união de  $A$  e  $B$ . A classe de funções t-conormas (*triangular co-norm*), também conhecidas como s-normas, possuem propriedades que garantem que os conjuntos *fuzzy* resultantes da aplicação dessas funções sejam significativos como uniões *fuzzy* (KLIR; YUAN, 1995; NICOLETTI; CAMARGO, 2004). Uma s-norma é uma operação binária  $u : [0, 1]^2 \rightarrow [0, 1]$  que atribui um valor  $(A \cup B)(x)$  considerando os graus de pertinência  $A(x)$  e  $B(x)$ , para todo  $x \in X$ . Segundo Klir e Yuan (1995), algumas s-normas que são frequentemente utilizadas como uniões *fuzzy*:

- *União padrão:*

$$u(a, b) = \max(a, b) \quad (2.12)$$

- *Soma algébrica:*

$$u(a, b) = a + b - ab \quad (2.13)$$

- *Soma limitada:*

$$u(a, b) = \min(1, a + b) \quad (2.14)$$

- *União drástica:*

$$u(a,b) = \begin{cases} a, & \text{quando } b = 0 \\ b, & \text{quando } a = 0 \\ 1, & \text{caso contrário} \end{cases} \quad (2.15)$$

Existem, ainda, operações de comparação de conjuntos *fuzzy*. Uma operação frequentemente utilizada é a *medida de possibilidade*, que quantifica a extensão com a qual conjuntos *fuzzy* se sobrepõem. A medida de possibilidade de um conjunto *fuzzy*  $A$  com relação a um conjunto *fuzzy*  $B$ , ambos definidos em  $X$ , denotada como  $Poss(A, B)$ , é definida como:

$$Poss(A, B) = \sup_{x \in X} [\min(A(x), B(x))] \quad (2.16)$$

## 2.3 Variáveis Linguísticas

Ao modelar o conhecimento de um domínio de aplicação, existem situações que demandam a descrição de determinados fenômenos pela caracterização de seus estados possíveis, fato que remete ao conceito de *variável*. Por exemplo, para representar condições ambientais, é possível descrever a *temperatura* utilizando termos linguísticos como *alta*, *baixa* ou *média*, em vez de usar valores numéricos precisos. Em geral, as caracterizações linguísticas são menos específicas que as numéricas, mas podem ser mais apropriadas quando não se sabe qual o valor numérico exato da variável em questão (PEDRYCZ; GOMIDE, 1998). Além disso, as descrições linguísticas são mais próximas da linguagem humana, facilitando a captura e a interpretação do conhecimento com relação aos especialistas humanos.

Segundo Pedrycz e Gomide (1998), para caracterizar fenômenos de uma forma aproximada, os valores de variáveis linguísticas podem ser representados por conjuntos *fuzzy*, de modo que seja possível traduzir as descrições linguísticas para descrições numéricas e computáveis. Formalmente, uma variável linguística é caracterizada por uma quintupla  $\langle X, T(X), \mathbf{X}, G, M \rangle$ , onde:

- $X$  é o nome da variável linguística;
- $T(X)$  é o conjunto de termos de  $X$  cujos elementos são rótulos de valores linguísticos de  $X$ ;
- $\mathbf{X}$  é o domínio ou conjunto universo da variável base  $x$ , que representa os valores numéricos precisos associados à variável linguística  $X$ ;

- $G$  é uma gramática para gerar os termos linguísticos;
- $M$  é uma regra semântica para associar cada rótulo  $L \in T(X)$  a seu significado  $M(L)$ , que é um conjunto *fuzzy* definido no universo  $\mathbf{X}$ .

Por exemplo, considere a variável linguística temperatura, ou seja,  $X = temperatura$ , com  $\mathbf{X} = [0, 50]$ . O conjunto de termos linguísticos associado a  $X$  pode ser  $T(X) = \{muito\ baixa, baixa, média, alta, muito\ alta, \dots\}$ . O valor  $M(L)$  de um rótulo  $L \in T(X)$  é definido para ser a restrição  $L(x)$  sobre a variável base  $x \in \mathbf{X}$ . Assim,  $M(L)$  é um conjunto *fuzzy* em  $\mathbf{X}$  cuja função de pertinência  $L(x)$  expressa a semântica do rótulo  $L$ . A Figura 2.3, adaptada de Pedrycz e Gomide (1998), ilustra a variável temperatura e alguns de seus termos linguísticos definidos como conjuntos *fuzzy* em  $[0, 50]$ .

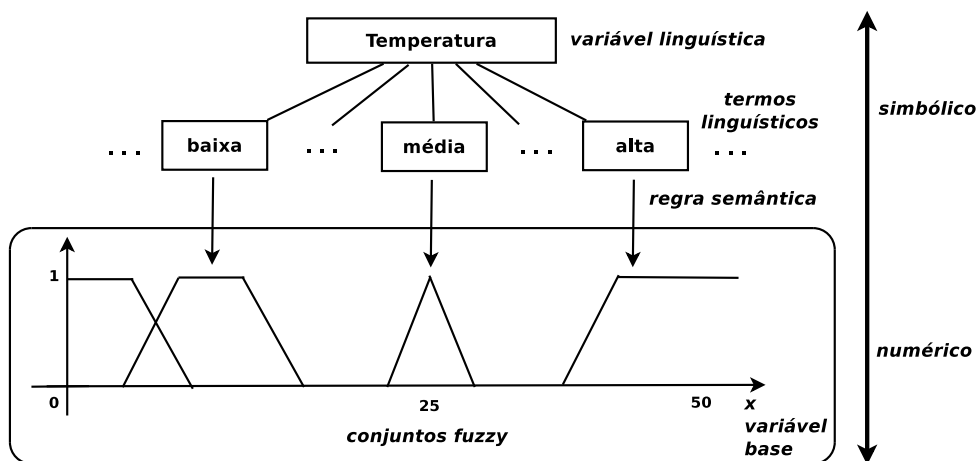


Figura 2.3: Variável linguística temperatura (Figura adaptada de (PEDRYCZ; GOMIDE, 1998)).

Em geral, a maioria das aplicações de variáveis linguísticas baseiam-se em conjuntos *fuzzy* com funções de pertinência parametrizadas (na Figura 2.3 são utilizadas funções triangulares e trapezoidais) e em um número pequeno de valores linguísticos. A quantidade de valores linguísticos define a granularidade do domínio da variável linguística e, portanto, define uma partição *fuzzy*. Um número pequeno de valores linguísticos induz uma partição grossa, enquanto um número grande induz uma partição fina do domínio da variável (PEDRYCZ; GOMIDE, 1998).

Conectivos (*e*, *ou*) e a negação podem ser utilizados para definir termos linguísticos mais complexos. Por exemplo, é possível definir o termo temperatura *não baixa e não alta*, a partir dos termos linguísticos *baixa* e *alta*, conectivo *e* e a negação. Para especificar o significado dos conectivos e da negação, empregam-se operações de interseção, união e complemento *fuzzy* (Seção 2.2). Assim,  $não\ baixa\ e\ não\ alta(x) = t([1 - baixa(x)], [1 - alta(x)])$ , considerando o complemento *fuzzy* padrão e a interseção *fuzzy* como uma *t*-norma  $t$ .



Variáveis linguísticas são frequentemente utilizadas em proposições condicionais ou regras *fuzzy*, de modo a permitir a inferência de conhecimento com base em métodos de raciocínio aproximado.

## 2.4 Regras *Fuzzy*

Segundo Pedrycz e Gomide (1998), regras proveem uma abordagem formal para representar diretivas ou estratégias geralmente apropriadas quando o conhecimento de domínio resulta de associações empíricas ou da experiência de especialistas. Em particular, as regras *fuzzy* são capazes de representar conhecimento com base em variáveis linguísticas, aproximando-se da linguagem humana. Assim, esta seção apresenta a sintaxe e a semântica de regras *fuzzy* utilizadas em sistemas de inferência *fuzzy*.

### 2.4.1 Sintaxe de Regras *Fuzzy*

A unidade básica de informação em regras *fuzzy* é uma proposição *fuzzy*  $p$ :

$$p : X \text{ é } A \quad (2.17)$$

onde  $X$  é uma variável linguística que assume valores de uma variável base  $x$  definida em um universo  $\mathbf{X}$  e  $A$  é um termo linguístico definido por um conjunto *fuzzy* em  $\mathbf{X}$ . Considerando o exemplo da Figura 2.3 sobre a variável linguística  $X = \text{temperatura}$ , uma possível proposição *fuzzy*  $p$  seria:

$$p : \text{temperatura é alta}$$

Proposições compostas podem ser construídas a partir de conjunções ou disjunções de proposições *fuzzy*, formando novas proposições  $p$  e  $q$ :

$$p : X_1 \text{ é } A_1 \text{ e } X_2 \text{ é } A_2 \text{ e } \dots \text{ e } X_n \text{ é } A_n \quad (2.18)$$

$$q : X_1 \text{ é } A_1 \text{ ou } X_2 \text{ é } A_2 \text{ ou } \dots \text{ ou } X_n \text{ é } A_n \quad (2.19)$$

onde  $X_1, X_2, \dots, X_n$  são variáveis linguísticas cujos termos linguísticos são respectivamente  $A_1, A_2, \dots, A_n$  definidos por conjuntos *fuzzy* nos universos  $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n$ . As proposições  $p$  e  $q$  induzem relações  $P$  e  $Q$  sobre  $\mathbf{X}_1 \times \mathbf{X}_2 \times \dots \times \mathbf{X}_n$  tais que:

$$P(x_1, x_2, \dots, x_n) = T_{i=1}^n [A_i(x_i)] \quad (2.20)$$

$$Q(x_1, x_2, \dots, x_n) = S_{i=1}^n [A_i(x_i)] \quad (2.21)$$

onde  $T$  e  $S$  denotam t-normas e s-normas representando operações de conjunção e disjunção, respectivamente.

Em geral, as proposições *fuzzy* condicionais ou regras *fuzzy* possuem o formato:

$$p : \text{se } \textit{antecedente} \text{ então } \textit{consequente}$$

onde *antecedente* e *consequente* são proposições *fuzzy*. O caso mais simples, em que o antecedente e o consequente contêm uma proposição *fuzzy* simples, segue o padrão dado pela Equação 2.22:

$$p : \text{se } X \text{ é } A \text{ então } Y \text{ é } B \quad (2.22)$$

onde  $X$  e  $Y$  são variáveis cujos termos linguísticos são os conjuntos *fuzzy*  $A$  e  $B$  definidos nos universos  $\mathbf{X}$  e  $\mathbf{Y}$ , respectivamente.

Regras *fuzzy* podem conter proposições *fuzzy* compostas, como as proposições das Equações 2.18 e 2.19. Um formato comum nas aplicações são as regras *fuzzy* contendo conjunções no antecedente, conforme ilustrado na Equação 2.23:

$$p : \text{se } X_1 \text{ é } A_1 \text{ e } X_2 \text{ é } A_2 \text{ e } \dots \text{ e } X_n \text{ é } A_n \text{ então } Y \text{ é } B \quad (2.23)$$

onde  $X_1, X_2, \dots, X_n$  e  $Y$  são variáveis cujos termos linguísticos são os conjuntos *fuzzy*  $A_1, A_2, \dots, A_n$  e  $B$  definidos nos universos  $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_n$  e  $\mathbf{Y}$ , respectivamente.

## 2.4.2 Semântica de Regras Fuzzy

Uma regra *fuzzy* descreve uma relação entre as variáveis presentes no antecedente e consequente. A regra *fuzzy* da Equação 2.22 pode ser definida por uma relação *fuzzy*  $R$ , com grau de pertinência  $R(x, y)$  representando o grau com o qual  $(x, y) \in \mathbf{X} \times \mathbf{Y}$  é compatível com a relação entre as variáveis  $X$  e  $Y$  (PEDRYCZ; GOMIDE, 1998). Alguns autores referem-se à relação  $R$  como uma relação que define a *semântica da regra*. Sendo  $A$  e  $B$  conjuntos *fuzzy* definidos nos universos  $\mathbf{X}$  e  $\mathbf{Y}$ , respectivamente, então a relação  $R(x, y)$  em  $\mathbf{X} \times \mathbf{Y}$  é definida por:

$$R(x, y) = f(A(x), B(y)), \forall (x, y) \in \mathbf{X} \times \mathbf{Y} \quad (2.24)$$

onde  $f$  é uma função da forma  $f : [0, 1]^2 \rightarrow [0, 1]$ . Em geral, as relações induzidas por regras *fuzzy* são derivadas de três classes principais de funções: conjunção *fuzzy*, disjunção *fuzzy* e implicação *fuzzy*, sendo a conjunção e a implicação as mais comuns (PEDRYCZ; GOMIDE, 1998). A conjunção *fuzzy* e a disjunção *fuzzy* são associadas às classes de funções t-normas e s-normas, respectivamente. Alguns exemplos de funções t-normas e s-normas são descritos na

Seção 2.2. Algumas funções de implicação *fuzzy* são derivadas da generalização da implicação da lógica clássica, definidas como:

$$f_i(A(x), B(y)) = s(\neg A(x), B(y)), \forall (x, y) \in \mathbf{X} \times \mathbf{Y} \quad (2.25)$$

onde  $\neg A(x)$  é o complemento *fuzzy* de  $A(x)$  e  $s$  é uma  $s$ -norma. Um exemplo típico é a *implicação de Kleene-Dienes*, definida por:

$$f_{kd}(A(x), B(y)) = \max(1 - A(x), B(y)) \quad (2.26)$$

Outro exemplo de implicação *fuzzy* é a *implicação de Gödel*, definida por:

$$f_g(A(x), B(y)) = \begin{cases} 1, & \text{se } A(x) \leq B(y) \\ B(y), & \text{caso contrário} \end{cases} \quad (2.27)$$

No caso de regras *fuzzy* contendo proposições *fuzzy* compostas por conjunção no antecedente (Equação 2.23), a relação induzida  $R$  é definida por:

$$R(x_1, x_2, \dots, x_n, y) = f(P_a(x_1, x_2, \dots, x_n), y) \quad (2.28)$$

onde  $f$  pode ser uma conjunção *fuzzy*, disjunção *fuzzy* ou implicação *fuzzy* e  $P_a$  é a relação induzida pela proposição *fuzzy* composta presente no antecedente da regra. A relação  $P_a$  é definida com base na Equação 2.20, em função do uso de conjunções no antecedente.

## 2.5 Raciocínio Aproximado com Regras Fuzzy

O cálculo baseado em regras *fuzzy* fundamenta-se na semântica das regras e na regra da inferência composicional, permitindo derivar uma conclusão mesmo quando não existe um casamento exato entre os antecedentes das regras e os valores das variáveis *fuzzy* (CORDÓN; JESUS; HERRERA, 1999). Embora custoso, esse processo de inferência pode ser simplificado dependendo dos operadores utilizados para a agregação de múltiplas regras, interseção na regra composicional e semântica das regras (PEDRYCZ; GOMIDE, 1998). Esta seção apresenta os fundamentos do raciocínio aproximado com regras *fuzzy* e os métodos de inferência comumente utilizados em sistemas de inferência *fuzzy*.

### 2.5.1 Regra da Inferência Composicional

Suponha um conjunto de  $N$  regras paralelas, que são uma sequência de proposições fuzzy condicionais que possuem as mesmas variáveis no antecedente e as mesmas variáveis no conseqüente, porém com diferentes combinações de termos linguísticos, consideradas em conjunto para o raciocínio aproximado:

$$\text{se } X \text{ é } A_1 \text{ então } Y \text{ é } B_1$$

$$\text{se } X \text{ é } A_2 \text{ então } Y \text{ é } B_2$$

...

$$\text{se } X \text{ é } A_N \text{ então } Y \text{ é } B_N$$

Conforme descrito na Seção 2.4.2, cada regra induz uma relação  $R_i(x, y) = f(A_i(x), B_i(y))$ , com  $i = 1 \dots N$ . As relações  $R_i$  podem ser agregadas por uma operação de agregação  $A$ , baseada em uma s-norma, t-norma ou operação de média, para se obter uma relação  $R = A_{i=1}^N R_i$ . Dada uma proposição  $p : X \text{ é } A'$ , onde  $A' \neq A_i, i = 1 \dots N$ , é possível concluir que  $Y \text{ é } B'$  pela aplicação da regra da inferência composicional utilizando a relação  $R$ . A regra da inferência composicional estabelece que, uma vez conhecidos  $A'$  e  $R$ , é possível obter  $B'$  pela equação:

$$B'(y) = \sup_{x \in X} \min[A'(x), R(x, y)], \forall y \in Y \quad (2.29)$$

No caso geral, a operação  $\min$  pode ser substituída por uma t-norma  $t$ , configurando uma composição sup-t:

$$B'(y) = \sup_{x \in X} t[A'(x), R(x, y)], \forall y \in Y \quad (2.30)$$

Uma forma alternativa para esse procedimento é aplicar a regra da inferência composicional para cada  $R_i$  e posteriormente agregar os  $B'_i$  inferidos de cada regra para concluir  $B'$ . Assim,  $B'$  é computado como  $B'(y) = A_{i=1}^N B'_i(y)$ , sendo que  $B'_i(y) = \sup_{x \in X} t[A'(x), R_i(x, y)]$ .

Considere  $N$  regras paralelas em que cada uma delas contém proposições fuzzy compostas no antecedente (Equação 2.31):

$$\text{se } X_1 \text{ é } A_1^k \text{ e } X_2 \text{ é } A_2^k \text{ e } \dots \text{ e } X_n \text{ é } A_n^k \text{ então } Y \text{ é } B^k \quad (2.31)$$

Neste caso, o antecedente de cada regra  $k$ , com  $k = 1 \dots N$ , induz uma relação  $P_{ak}(x_1, x_2, \dots, x_n) = T_{i=1}^n [A_i^k(x_i)]$ , onde  $T$  é uma t-norma. Para obter a relação  $R_k$  induzida por cada regra  $k$ , aplica-se uma operação de conjunção fuzzy, disjunção fuzzy ou implicação fuzzy  $f$  sobre a relação  $P_{ak}(x_1, \dots, x_n)$  e o conseqüente  $B(y)$ , resultando em:

$$R_k(x_1, \dots, x_n, y) = f(P_{ak}(x_1, \dots, x_n), B_k(y)) \quad (2.32)$$

Uma vez definida cada relação  $R_k$ , obtém-se a relação  $R(x_1, \dots, x_n, y) = A_{k=1}^N R_k(x_1, \dots, x_n, y)$ , onde  $A$  é uma operação de agregação. Dada uma proposição  $p : X_1 \text{ é } A'_1 \text{ e } \dots \text{ e } X_n \text{ é } A'_n$ , que também induz uma  $P'_a(x_1, \dots, x_n) = T_{i=1}^n [A'_i(x_i)]$ , pode-se concluir  $B'(y)$  como:

$$B'(y) = \sup_{x_1, \dots, x_n} t[P'_a(x_1, \dots, x_n), R(x_1, \dots, x_n, y)] \quad (2.33)$$

Uma discussão mais detalhada sobre o procedimento para inferência a partir de formatos mais generalizados de regras, com combinações de operações no antecedente e no consequente, pode ser vista em (PEDRYCZ; GOMIDE, 1998).

A complexidade do processo de raciocínio aproximado a partir da regra da inferência composicional pode inviabilizar o seu uso em aplicações práticas como, por exemplo, os sistemas de inferência *fuzzy*. Neste sentido, alguns motores de inferência baseiam-se em simplificações da regra da inferência composicional. Além disso, existem métodos de inferência que tratam situações específicas, como inferência sobre entradas numéricas e classificação de objetos. Neste contexto, as próximas seções descrevem métodos de raciocínio aproximado utilizados em contextos práticos, alguns deles inspirados na regra da inferência composicional, visando reduzir a complexidade do mecanismo de inferência.

## 2.5.2 Método da Interpolação

Sejam  $A_k, B_k$  e  $C_k$ , com  $k = 1 \dots N$ , conjuntos *fuzzy* respectivamente definidos nos conjuntos universos  $\mathbf{X}, \mathbf{Y}$  e  $\mathbf{Z}$  e um conjunto de regras paralelas do tipo *se  $X$  é  $A_k$  e  $Y$  é  $B_k$  então  $Z$  é  $C_k$* . Dada uma proposição *fuzzy* de entrada  $p : \text{se } X \text{ é } A' \text{ e } Y \text{ é } B'$ , onde  $A'$  e  $B'$  são conjuntos *fuzzy* respectivamente definidos em  $\mathbf{X}$  e  $\mathbf{Y}$ , pode-se aplicar uma simplificação da regra da inferência composicional, chamada de método da interpolação, para inferir o resultado  $Z \text{ é } C'$ , onde  $C'$  é um conjunto *fuzzy* definido em  $\mathbf{Z}$ . Essa simplificação depende da escolha da  $t$ -norma  $t$  utilizada na conjunção dos antecedentes, da operação  $f$  para as relações  $R_k$  induzidas pelas regras, da operação de composição para a regra da inferência composicional, além da operação  $A$  para agregação das regras paralelas. Um caso particular dessas operações, baseada na composição sup-min,  $A = \max$ ,  $t = \min$  e  $f = \min$ , permite simplificar a regra da inferência composicional para os seguintes passos (prova detalhada descrita por Pedrycz:1998):

1. Calcular a compatibilidade entre a proposição de entrada ( $A', B'$ ) e cada antecedente de cada regra  $k$ , utilizando a medida de possibilidade:

$$m_{Ak} = Poss(A', A_k) = \sup_{x \in \mathbf{X}} [\min(A'(x), A_k(x))], k = 1 \dots N$$

$$m_{Bk} = Poss(B', B_k) = \sup_{y \in Y} [\min(B'(y), B_k(y))], k = 1 \dots N$$

2. Calcular a conjunção dos antecedentes de cada regra  $k$  com a t-norma  $t = \min$ :

$$m_k = \min[m_{A_k}, m_{B_k}], k = 1 \dots N, \text{ onde } m_k \text{ é o grau de disparo da regra } k$$

3. Inferir o resultado  $C'_k$  para cada regra  $k$ , com base na conjunção pela t-norma  $t = \min$ :

$$C'_k(z) = m_k \wedge C_k(z) = \min[m_k, C_k(z)], k = 1 \dots N$$

4. Agregar os resultados das  $N$  regras com a operação de agregação para inferir o resultado final:

$$C'(z) = \mathbf{A}_{k=1}^N C'_k(z) = \max[C'_k(z), k = 1 \dots N], \forall z \in \mathbf{Z}$$

No caso geral, com regras contendo  $n$  proposições *fuzzy* ligadas por conjunção no antecedente, esse procedimento simplificado é estendido no passo 1. A medida de possibilidade  $m_i^k$  é calculada para cada proposição  $X_i$  é  $A_i^k$  do antecedente, com  $i = 1 \dots n$  e  $k = 1 \dots N$ . Além disso, no passo 2, o grau de disparo da regra  $k$  é calculado como  $m_k = \min[m_i^k, i = 1 \dots n]$ .

O método da interpolação inspirou o desenvolvimento de uma alternativa à regra de inferência composicional, denominada *inferência escalonada*, em que são possíveis outras combinações de operações para conjunção, agregação e semântica da regra. Porém, vale ressaltar que não há uma relação direta entre a inferência escalonada e a regra da inferência composicional (PEDRYCZ; GOMIDE, 1998). Um exemplo de método de inferência escalonada é o *produto-soma algébrica*, que utiliza a soma algébrica para agregação e o produto para conjunção e semântica de regra.

### 2.5.3 Métodos para Entradas e Saídas Numéricas

O método da interpolação trabalha com fatos ou entradas *fuzzy*, ou seja, os valores das variáveis são conjuntos *fuzzy*. Por exemplo, na proposição *fuzzy*  $X_1$  é  $A'$  e  $X_2$  é  $B'$ ,  $A'$  e  $B'$  são conjuntos *fuzzy* definidos em  $\mathbf{X}$  e  $\mathbf{Y}$ , respectivamente. Entretanto, existem aplicações que demandam inferências sobre entradas cujos valores das variáveis são numéricos ( $X_1$  é 1.5 e  $X_2$  é 50) com resultados também numéricos ( $Y$  é 2.5). Um exemplo de aplicação são os controladores *fuzzy*, um tipo de sistema de inferência *fuzzy* que controla o valor de uma determinada variável de saída em função dos valores de variáveis de entrada. A Figura 2.4 ilustra os componentes típicos de sistemas de inferência *fuzzy* que tratam entradas e saídas numéricas.

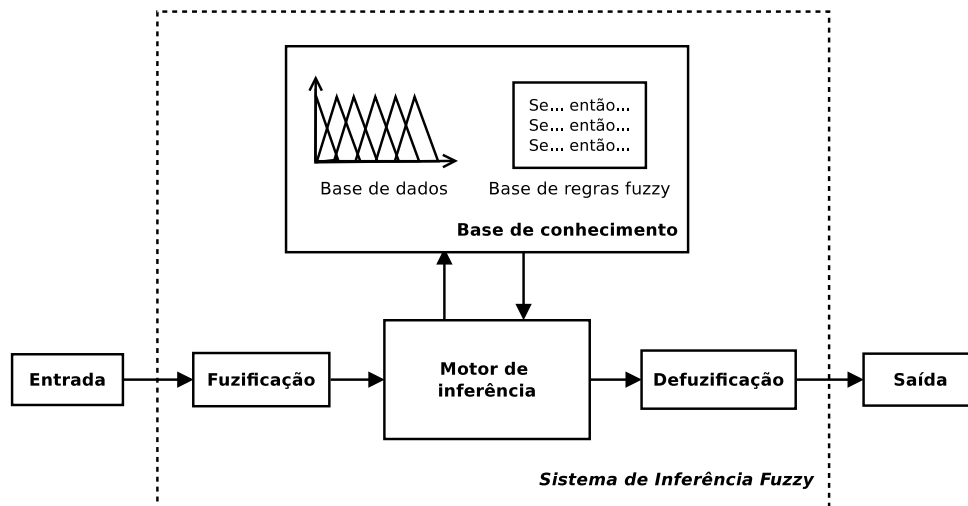


Figura 2.4: Sistema de inferência *fuzzy* com entradas e saídas numéricas.

Comparado a um sistema de inferência *fuzzy* típico (Figura 2.1), o sistema que trata entradas e saídas numéricas contém duas etapas adicionais: *fuzificação* e *defuzificação*. A fuzificação converte os valores numéricos de entrada para conjuntos *fuzzy* e respectivos graus de pertinência, para que sejam processados pelo motor de inferência. Ao final no processo de inferência, e etapa de defuzificação converte o conjunto *fuzzy* inferido para um valor numérico *crisp* que melhor o representa (KLIR; YUAN, 1995). Na sequência, são apresentados métodos de raciocínio aproximado que trabalham com entradas e saídas numéricas, além de métodos de defuzificação frequentemente utilizados em aplicações práticas.

### Método de Mamdani

Um dos métodos de inferência com entradas numéricas mais conhecidos da literatura é o método definido por Mamdani e Assilian (1975). Esse método baseia-se na simplificação da regra da inferência composicional com as operações de mínimo para conjunção e semântica das regras, composição max-min e agregação pela operação de máximo. O método de Mamdani pode ser visto como um caso particular do método da interpolação, em que o passo 1 (fuzificação) corresponde a verificar qual o grau de pertinência da entrada numérica  $a_i$  no conjunto *fuzzy*  $A_i^k$  de cada regra  $k$ . Assim, para  $N$  regras paralelas com  $n$  conjuntos *fuzzy* ( $A_i^k$ ) nos antecedentes e valores numéricos de entrada  $a_i$ , com  $i = 1 \dots n$  e  $k = 1 \dots N$ , o passo 1 é modificado:

$$m_i^k = A_i^k(a_i), i = 1 \dots n, k = 1 \dots N$$

Os demais passos continuam os mesmos: calcular o grau de disparo  $m_k$  de cada regra  $k$  pela operação de mínimo sobre os  $m_i^k$ , inferir o resultado  $C_k'$  para cada regra e agrupá-los usando a operação de máximo. Uma alternativa para o passo 1 é fuzificar as entradas numéricas,

transformando-as, por exemplo, em conjuntos *fuzzy* triangulares com o valor numérico  $a_i$  como centro do triângulo, e então calcular a medida de possibilidade da mesma forma que o método de interpolação.

### Método de Larsen

Outro método de inferência bastante citado na literatura é o método definido por Larsen (1980), que se diferencia do método de Mamdani em função das operações escolhidas: mínimo para conjunção dos antecedentes, máximo para agregação dos resultados parciais e composição max-produto (composição sup-t, com a t-norma produto algébrico). Desta forma, o passo 3, que infere o resultado  $C'_k$  para cada regra  $k$ , é modificado em função da composição max-produto:

$$C'_k(z) = m_k.C_k(z), k = 1 \dots N$$

### Métodos de Defuzzificação

Ambos métodos de Mamdani e Larsen tratam entradas numéricas, porém o resultado do processo de inferência é um conjunto *fuzzy* ( $Z$  é  $C'$ ), cuja interpretação nem sempre é trivial. Para as aplicações que demandam resultados numéricos, deve-se aplicar algum método de *defuzzificação*, que converte a conclusão obtida pelo motor de inferência, em termos de um conjunto *fuzzy*, para um valor numérico único. Existem diversos métodos de defuzzificação na literatura, alguns dos métodos mais comuns são apresentados a seguir, para um conjunto *fuzzy*  $C'$  definido sobre o domínio  $Z$ :

- *Centro de área (Center of Area - COA)*: também chamado de centro de gravidade ou método do centróide, define o valor defuzzificado como o valor de  $Z$  que divide a área sob a função de pertinência de  $C'$  em duas subáreas iguais. Quando  $Z$  é finito e discreto ( $Z = \{z_1, z_2, \dots, z_n\}$ ), o centro de área pode ser calculado como:

$$COA(C') = \frac{\sum_{k=1}^n C'(z_k) \cdot z_k}{\sum_{k=1}^n C'(z_k)} \quad (2.34)$$

- *Moment defuzzification*: um dos métodos de defuzzificação disponibilizados pelo *FuzzyJ Toolkit* (ORCHARD, 2001) que subdivide o conjunto *fuzzy*  $C'$  em  $m$  diferentes formatos ao particioná-lo verticalmente em pontos específicos que resultam em retângulos, triângulos e trapézios. O centro de área  $x_i$  ( $x_i \in Z$ ) e a área  $A_i$  de cada subdivisão  $i$ , com  $i = 1 \dots m$ , são calculados com base nas fórmulas apropriadas para cada formato. O valor



defuzificado é definido como:

$$moment(C') = \frac{\sum_{i=1}^m x_i \cdot A_i}{\sum_{i=1}^m A_i} \quad (2.35)$$

- *Menor dos máximos (Smallest of Maxima - SOM)* : o valor defuzificado é o menor valor de  $\mathbf{Z}$  que possui o grau de pertinência máximo. Quando  $\mathbf{Z}$  é finito e discreto, o valor defuzificado é calculado como:

$$SOM(C') = \min[z_k \mid z_k \in M], \text{ onde } M = \{z_k \mid C'(z_k) = \sup_{x \in \mathbf{X}} C'(x)\} \quad (2.36)$$

- *Maior dos máximos (Largest of Maxima - LOM)* : o valor defuzificado é o maior valor de  $\mathbf{Z}$  que possui o grau de pertinência máximo. Quando  $\mathbf{Z}$  é finito e discreto, o valor defuzificado é calculado como:

$$LOM(C') = \max[z_k \mid z_k \in M], \text{ onde } M = \{z_k \mid C'(z_k) = \sup_{x \in \mathbf{X}} C'(x)\} \quad (2.37)$$

- *Meio dos máximos (Middle of Maxima - MOM)* : o valor defuzificado é a média entre o menor valor e o maior valor de  $\mathbf{Z}$  que possuem o grau de pertinência máximo. Quando  $\mathbf{Z}$  é finito e discreto, o meio de máximos é definido como:

$$MOM(C') = \frac{SOM(C') + LOM(C')}{2} \quad (2.38)$$

- *Média dos máximos (Mean of Maxima)*: o valor defuzificado é a média dos valores de  $\mathbf{Z}$  que possuem o grau de pertinência máximo. Quando  $\mathbf{Z}$  é finito e discreto, o valor defuzificado é calculado como:

$$MeanMaxima(C') = \frac{\sum_{z_k \in M} z_k}{|M|}, \text{ onde } M = \{z_k \mid C'(z_k) = \sup_{x \in \mathbf{X}} C'(x)\} \quad (2.39)$$

Assim, é possível obter um valor numérico a partir do conjunto *fuzzy*  $C'$  inferido pelo motor de inferência por meio da aplicação de algum método de defuzificação.

### Método de Takagi-Sugeno-Kang

Outro método de raciocínio aproximado bastante utilizado é o método Takagi-Sugeno-Kang (TSK) proposto por Takagi e Sugeno (1985). Diferentemente dos demais métodos, o método TSK considera regras contendo funções no consequente, que combinam os valores numéricos das variáveis no antecedente para inferir o valor numérico da variável no consequente.

Neste caso, as regras possuem o seguinte formato:

$$\text{se } X \text{ é } A_k \text{ e } Y \text{ é } B_k \text{ então } z = f_k(x, y) \quad (2.40)$$

onde  $f_k : \mathbf{X} \times \mathbf{Y} \rightarrow \mathbf{Z}$  e  $k = 1 \dots N$ . No caso de uma função  $f_k(x, y) = c$ , onde  $c$  é uma constante, a regra é denominada regra TSK de ordem zero. Caso  $f_k(x, y)$  seja uma função polinomial de primeira ordem, a regra é denominada regra TSK de primeira ordem.

O resultado do processo de inferência é obtido pela combinação ponderada das contribuições das regras individuais, onde o peso de cada regra é o grau de disparo  $m_k$ . Assim, valor numérico do consequente é definido por:

$$z = \frac{\sum_{k=1}^N m_k \cdot f_k(x, y)}{\sum_{k=1}^N m_k} \quad (2.41)$$

## 2.6 Considerações Finais

Neste capítulo, foram apresentados os principais conceitos envolvendo sistemas de inferência *fuzzy*, contemplando fundamentos de conjuntos *fuzzy*, variáveis linguísticas, regras *fuzzy* e métodos de raciocínio aproximado. Essa fundamentação é importante para o desenvolvimento da tese de doutorado assim como para a compreensão dos trabalhos correlatos. No próximo capítulo, são apresentados conceitos básicos sobre ontologias e ontologias *fuzzy*, igualmente essenciais para este trabalho de pesquisa de doutorado.

# Capítulo 3

## ONTOLOGIAS *Fuzzy*

---

---

### 3.1 Considerações Iniciais

Ontologias são utilizadas para expressar conhecimento sobre um determinado domínio ou área de interesse, modelando seus principais conceitos, atributos e relacionamentos. Dependendo do domínio, o conhecimento pode ser caracterizado como impreciso, ou seja, as informações não são exatas ou bem definidas. Como ontologias convencionais baseiam-se em formalismos que modelam informações de modo preciso ou exato, são menos adequadas para representar domínios que envolvem imprecisão. Para expressar esse tipo de informação, diversas pesquisas investigam como estender ontologias com base na teoria de conjuntos *fuzzy* (ZADEH, 1965), resultando em ontologias *fuzzy*.

Assim, este capítulo visa descrever conceitos básicos sobre ontologias e ontologias *fuzzy*, a fim de proporcionar a fundamentação necessária para os trabalhos desenvolvidos no âmbito deste doutorado. Primeiramente, na Seção 3.2, ontologias são definidas e caracterizadas com relação aos seus principais componentes. A Seção 3.3 descreve conceitos básicos sobre Lógicas de Descrição, um dos formalismos lógicos mais utilizados como fundamentação para ontologias, com foco na lógica *ALC*. Em sequência, a Seção 3.4 discute a extensão de ontologias considerando imprecisão. A Seção 3.5 descreve a representação de conhecimento e inferência em Lógicas de Descrição *Fuzzy*, com destaque para a lógica *fuzzy ALC*.

### 3.2 Ontologias: Definição e Caracterização

O termo “ontologia” originou-se na Filosofia como uma teoria que estuda a natureza do ser ou da existência (BUNGE, 1977). Nesse sentido filosófico, um exemplo clássico de ontologia

remete às categorias primitivas de Aristóteles, que classificam os seres de acordo com suas características (GRUBER, 2009). Etimologicamente, a palavra latina “ontologia” deriva da junção dos termos gregos “ont”, particípio do verbo “einai” (ser), e “logia” (estudo), que pode ser traduzida como o “estudo da existência” (GUIZZARDI, 2005).

Especificamente, no contexto da Ciência da Computação, uma das definições de ontologia mais citadas na literatura foi proposta por Gruber (1993): “Uma ontologia é uma especificação formal e explícita de uma conceitualização compartilhada”. Apesar dos esforços de autores como Uschold e Grüninger (2004) em explicitar o significado da definição de Gruber (1993), uma gama de especificações podem ser consideradas ontologias: desde simples glossários até rigorosos formalismos lógicos. Em virtude da discussão sobre as especificações que podem ser consideradas ontologias, Gruber (2009) complementou sua definição anterior com um foco mais voltado para os componentes de ontologias:

No contexto das Ciências da Computação e da Informação, uma ontologia define um conjunto de primitivas representacionais com as quais é possível modelar um domínio de conhecimento ou discurso. As primitivas representacionais são tipicamente classes (ou conjuntos), atributos (ou propriedades), e relacionamentos (relações entre membros das classes). As definições das primitivas representacionais incluem informação sobre seus significados, além de restrições sobre a sua aplicação de modo logicamente consistente.

É possível verificar que a definição de Gruber (2009) enfatiza as primitivas representacionais que tipicamente são utilizadas em ontologias. Independentemente da discussão sobre quais especificações devem ser consideradas ontologias, Gruber (2009) defende que uma ontologia deve definir (ou especificar) conceitos, relacionamentos e outros elementos que sejam relevantes para modelar um domínio. Uschold e Grüninger (2004) detalham o significado dessas primitivas representacionais, considerando conceitos, propriedades (atributos e relacionamentos), além de axiomas e indivíduos. Segundo tais autores, ontologias representam diferentes tipos de coisas em um domínio ou área de interesse, que são modelados como classes ou conceitos organizados em uma hierarquia ou taxonomia. Atributos descrevem características dos conceitos e relacionamentos expressam como os conceitos estão interligados. Axiomas são restrições sobre conceitos e propriedades, cujo principal propósito é expressar significado interpretável por máquinas, para possibilitar mecanismos de raciocínio automático. Por fim, indivíduos representam instâncias ou objetos concretos que seguem as definições modeladas pela ontologia.

No escopo desta tese de doutorado, ontologias são consideradas sob um ponto de vista formal, contendo primitivas representacionais fundamentadas por formalismos lógicos que possibilitam representação de conhecimento e inferência de modo coerente e não ambíguo. Neste

sentido, a próxima seção descreve conceitos básicos sobre Lógicas de Descrição, um formalismo lógico bastante difundido na comunidade de ontologias.

### 3.3 Lógicas de Descrição

Dentre os formalismos que fundamentam ontologias, as Lógicas de Descrição (*Description Logics - DLs*) destacam-se por aliarem expressividade e computabilidade, equipadas com uma semântica formal baseada em lógica. As DLs são consideradas subconjuntos decidíveis da lógica de primeira ordem, em que indivíduos correspondem a constantes, conceitos a predicados unários e relacionamentos a predicados binários (HORROCKS, 2008).

Para descrever os fundamentos de DLs, inferências típicas e o procedimento de raciocínio, as próximas seções baseiam-se na DL  $\mathcal{ALC}$ , que contém os construtores de conceito básicos em Lógicas de Descrição. As definições apresentadas podem ser encontradas de forma completa nos trabalhos de Nardi e Brachman (2003), Baader e Nutt (2003) e Baader (2003).

#### 3.3.1 Sintaxe e Semântica de $\mathcal{ALC}$

As Lógicas de Descrição são baseadas em descrições, razão de seu próprio nome. Descrições elementares são representadas por conceitos e relacionamentos atômicos, que servem de base para a construção de descrições complexas a partir de construtores providos pela lógica. Em notação abstrata, as letras  $A$  e  $B$  são usadas para representar conceitos atômicos;  $R$  e  $S$  para relacionamentos atômicos;  $C$  e  $D$  para descrições de conceito, que representam tanto conceitos atômicos quanto complexos; e letras minúsculas  $a$  e  $b$  para representar indivíduos ou instâncias.

Em geral, uma base de conhecimento  $\mathcal{K}$  de DLs é composta por dois componentes:  $TBox \mathcal{T}$  e  $ABox \mathcal{A}$  (NARDI; BRACHMAN, 2003; BAADER; NUTT, 2003). A  $TBox$  contém conhecimento intencional sob a forma de uma terminologia ou taxonomia, construída por meio de declarações que descrevem propriedades gerais de conceitos. A  $TBox$  introduz o vocabulário do domínio de aplicação, especificando conceitos que denotam conjuntos de indivíduos. Os axiomas típicos de uma  $TBox$  são definições de conceito e axiomas de inclusão ou subsunção que definem a taxonomia de conceitos relacionados por especialização/generalização. A  $ABox$  contém conhecimento relacionado a asserções específicas sobre os indivíduos ou instâncias do domínio de interesse. Os principais construtores de uma  $ABox$  são asserções de conceito, que instanciam indivíduos considerando conceitos da  $TBox$ , e asserções de relacionamento, que instanciam associações entre indivíduos.

Segundo Baader e Nutt (2003), para definir uma semântica formal dos elementos das DLs são consideradas interpretações, em que cada interpretação  $\mathcal{I}$  consiste de um conjunto não vazio  $\Delta^{\mathcal{I}}$ , composto por indivíduos, e uma função de interpretação que atribui para cada conceito atômico  $A$  um conjunto  $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$  e para cada relacionamento atômico  $R$  uma relação binária  $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ . A função de interpretação é estendida para definir indutivamente a semântica das descrições complexas com base no significado de cada construtor e das descrições elementares. A Tabela 3.1, adaptada de (BAADER, 2003), apresenta a sintaxe e a semântica dos construtores de  $\mathcal{ALC}$ , além dos axiomas terminológicos (TBox) e asserções (ABox) que compõem a base de conhecimento de DLs.

**Tabela 3.1: Sintaxe e semântica de  $\mathcal{ALC}$  (BAADER, 2003).**

Construtores de conceito de $\mathcal{ALC}$		
Nome	Sintaxe	Semântica
Conceito universal	$\top$	$\Delta^{\mathcal{I}}$
Conceito vazio	$\perp$	$\emptyset$
Negação de conceito	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
Interseção de conceitos	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
União de conceitos	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
Restrição de valor de conceito	$\forall R.C$	$\{a \in \Delta^{\mathcal{I}} \mid \forall b.(a,b) \in R^{\mathcal{I}} \rightarrow b \in C^{\mathcal{I}}\}$
Quantificação existencial de conceito	$\exists R.C$	$\{a \in \Delta^{\mathcal{I}} \mid \exists b.(a,b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\}$
Axiomas terminológicos (TBox)		
Definição de conceito	$A \equiv D$	$A^{\mathcal{I}} = D^{\mathcal{I}}$
Equivalência de conceitos	$C \equiv D$	$C^{\mathcal{I}} = D^{\mathcal{I}}$
Introdução de conceito primitivo	$A \sqsubseteq D$	$A^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
Inclusão geral de conceitos	$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
Asserções (ABox)		
Asserção de conceito	$C(a)$ ou $a : C$	$a^{\mathcal{I}} \in C^{\mathcal{I}}$
Asserção de relacionamento	$R(a,b)$ ou $\langle a,b \rangle : R$	$(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$

Com relação aos axiomas terminológicos da Tabela 3.1, os axiomas de equivalência e inclusão de conceitos que envolvem um conceito atômico  $A$  do lado esquerdo são denominados definições de conceito e introduções de conceito primitivo, respectivamente. Uma TBox é denominada *definitória* se contém apenas definições de conceito não ambíguas e acíclicas (BAADER; HORROCKS; SATTLE, 2007). As asserções de conceito e relacionamento compõem a ABox, instanciando indivíduos que seguem os axiomas modelados na TBox.

Se uma interpretação  $\mathcal{I}$  satisfaz um axioma terminológico ou uma asserção conforme a semântica mostrada na Tabela 3.1, então  $\mathcal{I}$  é considerada um modelo desse axioma ou asserção. Uma interpretação  $\mathcal{I}$  é um modelo de uma TBox  $\mathcal{T}$  se e somente se  $\mathcal{I}$  satisfaz cada axioma terminológico de  $\mathcal{T}$  ( $\mathcal{I} \models \mathcal{T}$ ). Uma interpretação  $\mathcal{I}$  é um modelo de uma ABox  $\mathcal{A}$  se e somente se  $\mathcal{I}$  satisfaz cada asserção em  $\mathcal{A}$  ( $\mathcal{I} \models \mathcal{A}$ ). Por fim, uma interpretação  $\mathcal{I}$  é um modelo de uma base de conhecimento  $\mathcal{K} = (\mathcal{T}, \mathcal{A})$  se e somente se  $\mathcal{I}$  é um modelo de  $\mathcal{T}$  e  $\mathcal{I}$  é um modelo de  $\mathcal{A}$  ( $\mathcal{I} \models \mathcal{K}$ ) (BAADER; HORROCKS; SATTLE, 2007).

### 3.3.2 Inferências Típicas de DLs

A semântica formal de DLs possibilita inferir conhecimento implícito a partir dos elementos modelados em uma base de conhecimento, composta por TBox e ABox. Segundo Baader e Nutt (2003), as tarefas de inferência típicas para uma terminologia (TBox) são:

- Satisfazibilidade: Um conceito  $C$  é satisfatível com relação a uma TBox  $\mathcal{T}$  se existe uma interpretação ou modelo  $\mathcal{I}$  de  $\mathcal{T}$  tal que o conjunto  $C^{\mathcal{I}}$  não seja vazio, ou seja,  $\mathcal{I}$  é um modelo de  $C$ ;
- Subsunção: Um conceito  $D$  é mais genérico que um conceito  $C$  com relação a uma TBox  $\mathcal{T}$  se  $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$  para qualquer interpretação  $\mathcal{I}$  de  $\mathcal{T}$  ( $\mathcal{T} \models C \sqsubseteq D$ );
- Equivalência: Conceitos  $C$  e  $D$  são equivalentes com relação a uma TBox  $\mathcal{T}$  se  $C^{\mathcal{I}} = D^{\mathcal{I}}$  para qualquer interpretação  $\mathcal{I}$  de  $\mathcal{T}$  ( $\mathcal{T} \models C \equiv D$ );
- Disjunção: Conceitos  $C$  e  $D$  são disjuntos com relação a uma TBox  $\mathcal{T}$  se  $C^{\mathcal{I}} \cap D^{\mathcal{I}} = \emptyset$  para qualquer interpretação  $\mathcal{I}$  de  $\mathcal{T}$ .

Com relação a ABox, os serviços de inferência checam se seu conjunto de asserções é consistente com relação a uma TBox, isto é, se existe uma interpretação  $\mathcal{I}$  que satisfaz ambas TBox  $\mathcal{T}$  e ABox  $\mathcal{A}$  ( $\mathcal{I} \models \mathcal{T}$  e  $\mathcal{I} \models \mathcal{A}$ ). Segundo Baader e Nutt (2003), as tarefas típicas de inferência em ABox são:

- Verificação de instâncias: verifica se uma asserção  $\alpha$  (de conceito ou de relacionamento) é uma implicação lógica de uma ABox  $\mathcal{A}$  ( $\mathcal{A} \models \alpha$ ) se toda interpretação  $\mathcal{I}$  que satisfaz  $\mathcal{A}$  também satisfaz  $\alpha$ ;
- Consistência da base de conhecimento: verifica se cada descrição de conceito em uma base de conhecimento  $\mathcal{K}$  admite pelo menos um indivíduo, ou seja, não é contraditória. Uma ABox  $\mathcal{A}$  é consistente com relação a uma TBox  $\mathcal{T}$ , se existe uma interpretação  $\mathcal{I}$  que é um modelo de ambos  $\mathcal{A}$  e  $\mathcal{T}$  ( $\mathcal{I} \models \mathcal{K}$ );
- Recuperação: recupera os indivíduos na base de conhecimento que são instâncias de um determinado conceito - dada uma ABox  $\mathcal{A}$  e um conceito  $C$ , verifica todos os indivíduos  $a$  tais que  $\mathcal{A} \models C(a)$ ;
- Realização: determina os conceitos mais específicos ao qual um indivíduo pertence - dado um indivíduo  $a$ , verifica os conceitos mais específicos  $C$  tais que  $\mathcal{A} \models C(a)$ .

Esses serviços de raciocínio para ABox podem ser reduzidos a uma única tarefa. A recuperação e a realização podem ser efetuadas por uma sequência de verificações de instâncias, embora de forma não otimizada. A verificação de instâncias, por sua vez, pode ser reduzida à verificação de consistência da base de conhecimento: dada uma ABox  $\mathcal{A}$ ,  $\mathcal{A} \models C(a)$  se e somente se  $\mathcal{A} \cup \{\neg C(a)\}$  é inconsistente (BAADER; NUTT, 2003). Portanto, os serviços de inferência típicos para ABox resumem-se à verificação da (in)consistência da base de conhecimento, sendo suficiente o desenvolvimento de algoritmos de raciocínio para essa tarefa.

As tarefas de inferência para TBox também podem ser reduzidas ao problema de verificação de consistência (BAADER; NUTT, 2003). A satisfazibilidade de conceitos é reduzida da seguinte forma: dada uma ABox  $\mathcal{A}$ , um conceito  $C$  é satisfável se e somente se  $\mathcal{A} \cup \{C(a)\}$  é consistente, onde  $a$  é um novo indivíduo arbitrário. Ao provar que  $\mathcal{A} \cup \{C(a)\}$  é consistente, obtém-se uma interpretação  $\mathcal{I}$  válida que satisfaz o conceito  $C$ , ou seja, uma interpretação  $\mathcal{I}$  em que  $C$  não é vazio (não é contraditório). Ainda segundo Baader e Nutt (2003), como as tarefas de subsunção, equivalência e disjunção de conceitos podem ser reduzidas para a satisfazibilidade de conceitos, a verificação de consistência da base de conhecimento também resolve os principais serviços de raciocínio para TBox, desde que a Lógica de Descrição possua os construtores de interseção e negação de conceitos.

### 3.3.3 Raciocínio Baseado em *Tableau*

Como as tarefas de raciocínio típicas de DLs podem ser reduzidas para a tarefa de verificação de consistência da base de conhecimento, um método de inferência que trate essa tarefa é suficiente para também realizar as demais. Um dos métodos mais utilizados na literatura de DLs são os algoritmos baseados em *tableau* para raciocínio em TBox e ABox, projetados para verificar a consistência de bases de conhecimento. Em Lógicas de Descrição, o primeiro algoritmo baseado em *tableau* foi proposto por Schmidt-Schauß e Smolka (1991) para a lógica  $\mathcal{ALC}$ . Desde então, essa abordagem tem sido empregada para obter algoritmos corretos, completos e finitos para realizar as tarefas de inferência em diversas DLs (BAADER; NUTT, 2003).

Segundo Baader, Horrocks e Sattler (2007), um algoritmo baseado em *tableau* verifica a consistência de uma base de conhecimento ao tentar construir um modelo ou uma interpretação  $\mathcal{I}$  em que cada conceito não é vazio. Um *tableau* é um grafo dirigido que representa tal modelo, cujos nós correspondem a indivíduos (elementos de  $\Delta^{\mathcal{I}}$ ) e arestas correspondem a relacionamentos entre indivíduos (elementos de  $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ ) (HORROCKS, 2003). Cada nó  $x$  é rotulado com um conjunto de conceitos  $\mathcal{L}(x) = \{C_1, \dots, C_n\}$ , sendo que cada conceito  $C_i$  pertence a  $\mathcal{L}(x)$  se o indivíduo representado pelo nó  $x$  pertencer à interpretação de  $C_i$ . Cada



aresta  $\langle x, y \rangle$  entre os nós  $x$  e  $y$  é rotulada com um relacionamento  $R$  ( $\mathcal{L}(\langle x, y \rangle) = R$ ) caso a tupla de indivíduos correspondente a  $\langle x, y \rangle$  pertencer à interpretação de  $R$ . Como pré-requisito para o algoritmo, todas as descrições de conceito devem ser representadas segundo a *forma normal negativa* (*negation normal form* - NNF), em que negações são aplicadas somente a conceitos atômicos. A conversão para a forma normal negativa pode ser realizada pela aplicação das leis de De Morgan e das equivalências  $\neg(\exists R.C) \Leftrightarrow (\forall R.\neg C)$  e  $\neg(\forall R.C) \Leftrightarrow (\exists R.\neg C)$ .

Para verificar a consistência de uma base de conhecimento  $\mathcal{K}$  com uma TBox vazia e uma ABox  $\mathcal{A} = \{C_1(x)\}$ , onde  $C_1$  pode ser uma descrição de conceito complexa, o algoritmo inicializa o grafo com um nó  $x$  contendo o rótulo  $\mathcal{L}(x) = \{C_1\}$ , para então aplicar regras de expansão que estendem os rótulos dos nós ou adicionam novos nós, arestas e rótulos ao *tableau*. A Tabela 3.2 mostra o conjunto de regras de expansão para a lógica  $\mathcal{ALC}$ , onde  $C$  e  $D$  representam conceitos e  $R$  relacionamentos binários. Uma contradição (*clash*) é detectada quando  $\{C, \neg C\} \subseteq \mathcal{L}(x)$  para algum conceito  $C$  e algum nó  $x$ . Se não houver outra alternativa sem contradição (no caso da *regra*  $-\sqcup$ ), o algoritmo finaliza a execução e conclui que não é possível obter uma interpretação para o conceito  $C_1$ , ou seja, a base de conhecimento  $\{C_1(x)\}$  é inconsistente. Caso não sejam detectadas contradições, o algoritmo finaliza a execução quando não for mais possível aplicar as regras de expansão, obtendo-se, assim, um *tableau* correspondente a uma interpretação  $\mathcal{I}$  que satisfaz o conceito  $C_1$ , ou seja, a base de conhecimento formada por  $\{C_1(x)\}$  é consistente.

**Tabela 3.2: Regras de expansão *tableau* para  $\mathcal{ALC}$  (extraída de (HORROCKS, 2003)).**

<i>regra</i> $-\sqcap$	se $(C \sqcap D) \in \mathcal{L}(x)$ e $\{C, D\} \not\subseteq \mathcal{L}(x)$ então $\mathcal{L}(x) \longrightarrow \mathcal{L}(x) \cup \{C, D\}$
<i>regra</i> $-\sqcup$	se $(C \sqcup D) \in \mathcal{L}(x)$ e $\{C, D\} \cap \mathcal{L}(x) = \emptyset$ então $\mathcal{L}(x) \longrightarrow \mathcal{L}(x) \cup \{C\}$ ou $\mathcal{L}(x) \longrightarrow \mathcal{L}(x) \cup \{D\}$
<i>regra</i> $-\exists$	se $\exists R.C \in \mathcal{L}(x)$ e não existe nó $y$ tal que $\mathcal{L}(\langle x, y \rangle) = R$ e $C \in \mathcal{L}(y)$ então crie um novo nó $y$ e a aresta $\langle x, y \rangle$ com $\mathcal{L}(y) = \{C\}$ e $\mathcal{L}(\langle x, y \rangle) = R$
<i>regra</i> $-\forall$	se $\forall R.C \in \mathcal{L}(x)$ e existe algum nó $y$ tal que $\mathcal{L}(\langle x, y \rangle) = R$ e $C \notin \mathcal{L}(y)$ então $\mathcal{L}(y) \longrightarrow \mathcal{L}(y) \cup \{C\}$

Para resolver as outras tarefas de raciocínio, devem ser aplicadas as reduções apropriadas para se obter um problema de verificação de consistência de base de conhecimento correspondente, conforme descrito na Seção 3.3.2. No caso de bases de conhecimento com uma TBox  $\mathcal{T}$  não vazia e definitiva, pode ser aplicada a técnica de *unfolding* (HORROCKS, 2003), que consiste em substituir, para cada definição  $A \equiv C$ , as ocorrências de  $A$  por sua definição  $C$

nas descrições de conceitos complexas de  $\mathcal{K}$ . Para as inclusões de conceitos do tipo  $A \sqsubseteq C$ , as ocorrências de  $A$  são substituídas por  $A' \sqcap C$ , sendo que  $A'$  é um novo conceito atômico que representa as características que diferenciam o conceito  $A$  do conceito  $C$ . Como resultado do *unfolding*, as descrições de conceito complexas de  $\mathcal{K}$  são formadas apenas por conceitos atômicos, correspondendo a uma base de conhecimento com uma TBox vazia.

### 3.3.4 Tipos de Dados Concretos em DLs

Uma importante extensão em DLs são os domínios concretos, que permitem incorporar características concretas tais como números reais, inteiros, intervalos de tempo ou cadeias de caracteres (*strings*) em DLs (LUTZ, 2003). A partir dos domínios concretos, é possível modelar propriedades concretas dos indivíduos, como a idade de uma pessoa, o peso de um objeto, a temperatura do ambiente, além da comparação dos valores desses atributos. Por exemplo, ontologias usadas em aplicações de comércio eletrônico podem classificar itens de acordo com o peso e ainda inferir que itens pesando mais de 50kg requerem frete especial (HORROCKS; SATTLER, 2001).

Conforme descrito por Lutz (2003), um domínio concreto  $\mathcal{D}$  é um par  $(\Delta_{\mathcal{D}}, \Phi_{\mathcal{D}})$  onde  $\Delta_{\mathcal{D}}$  é um conjunto e  $\Phi_{\mathcal{D}}$  um conjunto de nomes de predicados. Cada nome de predicado  $P \in \Phi_{\mathcal{D}}$  é associado com uma aridade  $n$  e um predicado  $n$ -ário  $P^{\mathcal{D}} \subseteq \Delta_{\mathcal{D}}^n$ . Para ilustrar um exemplo de domínio concreto numérico,  $\Delta_{\mathcal{D}}$  pode ser o conjunto dos números racionais  $\mathbb{Q}$  e o conjunto  $\Phi_{\mathcal{D}}$  pode ser composto por predicados unários  $P_q$ , com  $P \in \{<, \leq, =, \neq, \geq, >\}$ ,  $q \in \mathbb{Q}$  e  $(P_q)^{\mathcal{D}} = \{q' \in \mathbb{Q} \mid q' P q\}$ . Por exemplo, o predicado  $<_{18}$  denota o conjunto de números racionais menores que 18, com  $(<_{18})^{\mathcal{D}} = \{q' \in \mathbb{Q} \mid q' < 18\}$ .

Segundo Baader, Horrocks e Sattler (2007), os domínios concretos em sua forma irrestrita podem prejudicar a decidibilidade e a complexidade computacional das DLs. Por esta razão, uma forma mais restrita de domínios concretos, conhecida como *tipos de dados concretos* (HORROCKS; SATTLER, 2001), é frequentemente utilizada em aplicações práticas. Uma dessas restrições considera que  $\Delta_{\mathcal{D}}$  é definido como o domínio de todos os tipos de dados concretos, ao invés de considerar algum domínio concreto específico  $\Delta_{\mathcal{D}}$  (PAN; HORROCKS, 2002). Assim, segundo Pan e Horrocks (2002),  $\Delta_{\mathcal{D}}$  é o domínio de tipos de dados cobrindo todos os tipos de dados concretos. Um conjunto de tipos de dados concretos  $d_1, \dots, d_m$  pode ser definido sobre  $\Delta_{\mathcal{D}}$ , sendo que o domínio de cada tipo de dado é um subconjunto de  $\Delta_{\mathcal{D}}$ , ou seja  $\forall i : 1 \leq i \leq m, d_i^{\mathcal{D}} \subseteq \Delta_{\mathcal{D}}$ . Predicados de tipos de dados  $P_1, \dots, P_r$  podem ser definidos sobre esses tipos de dados e cada predicado  $P_j$  está associado com uma aridade  $n_j$ , onde  $\forall j : 1 \leq j \leq r, P_j^{\mathcal{D}} \subseteq \Delta_{\mathcal{D}}^{n_j}$ . Cada predicado  $P_j(v_1, \dots, v_{n_j})$  é uma função booleana, onde  $v_1, \dots, v_{n_j}$

são suas variáveis de entrada, retornando verdadeiro se e somente se existe uma tupla de valores de dados  $t_1, \dots, t_{n_j}$  em  $\Delta_D$  tal que  $\forall i : 1 \leq i \leq n_j, v_i^D = t_i$  e  $\langle t_1, \dots, t_{n_j} \rangle \in P_j^D$ . Pan e Horrocks (2002) exemplificam um tipo de dado, INT+, que possui o conjunto de inteiros não negativos como seu domínio  $d_{INT+} \subseteq \Delta_D$ . Predicados binários como  $\geq$  e predicados unários  $\geq_n$  podem ser definidos sobre INT+. Para denotar o suporte a tipos de dados concretos, adiciona-se o sufixo (D) ao nome da DL, por exemplo,  $\mathcal{ALC}(D)$ ,  $\mathcal{SHIF}(D)$ .

Dependendo da semântica do domínio, tipos de dados e predicados concretos personalizados podem ser necessários para expressar o conhecimento de forma adequada. Neste sentido, Pan (2007) definiu o conceito de grupos de tipos de dados e expressões de tipos de dados que permitem personalizar tipos de dados e predicados concretos. Um grupo de tipos de dados  $G$  é um grupo de predicados embutidos, definidos sobre um conjunto de tipos de dados primitivos, provendo um formalismo unificado para tratar tipos de dados e predicados de tipos de dados. Para denotar o suporte a grupos de tipos de dados, adiciona-se o sufixo (G) ao nome da DL, por exemplo,  $\mathcal{ALC}(G)$ ,  $\mathcal{SHIF}(G)$ .

Segundo Pan (2007), um predicado de tipo de dado  $p$  é caracterizado por uma aridade  $a(p)$  e uma extensão de predicado  $E(p)$ . Por exemplo,  $=^{int+}$  é um predicado binário com aridade  $a(=^{int+}) = 2$  e extensão  $E(=^{int+}) = \{\langle i_1, i_2 \rangle \in V(INT+)^2 \mid i_1 = i_2\}$ , onde  $V(d)$  representa o conjunto de valores do tipo de dado  $d$ . Para cada predicado de tipo de dado  $p$  existe um tipo de dado primitivo  $d$  tal que  $E(p) \subseteq V(d)^{a(p)}$ , com  $V(d) \subseteq \Delta_D$ . Os tipos de dados primitivos são disjuntos, ou seja, para quaisquer dois tipos de dados primitivos  $d_1, d_2$ ,  $V(d_1) \cap V(d_2) = \emptyset$  (PAN, 2007).

Em um grupo de tipos de dados, expressões de tipos de dados podem ser usadas para representar tipos de dados e predicados personalizados. A Tabela 3.3 mostra a sintaxe e a semântica do conjunto de expressões de tipos de dados  $\mathbf{Dexp}(G)$ .

**Tabela 3.3: Sintaxe e semântica de expressões de tipos de dados.**

Expressão	Sintaxe	Semântica
expressões atômicas	$p$	$p^D$
expressões negadas relativizadas	$\bar{p}$	$\Delta_D \setminus p^D$
expressões enumeradas	$\{l_1, \dots, l_n\}$	$\{l_1^D\} \cup \dots \cup \{l_n^D\}$
expressões de domínio	$[u_1, \dots, u_n]$	$u_1^D \times \dots \times u_n^D$
expressões conjuntivas	$E_1 \wedge \dots \wedge E_n$	$E_1^D \cap \dots \cap E_n^D$
expressões disjuntivas	$E_1 \vee \dots \vee E_n$	$E_1^D \cup \dots \cup E_n^D$

DLs com suporte a grupos de tipos de dados contêm *atributos de tipos de dados*, também denominados *propriedades concretas*, que correspondem a relacionamentos entre indivíduos  $a \in \Delta^I$  e valores concretos  $x \in \Delta_D$ , além de construtores de conceito adicionais que permitem descrever conceitos a partir de restrições sobre valores concretos usando expressões de tipos de

dados. Assim, o conjunto de indivíduos é composto por  $I_A$ , que denota o conjunto de indivíduos em  $\Delta^{\mathcal{I}}$ , denominados indivíduos abstratos, e por  $I_D$ , que denota o conjunto de indivíduos que correspondem a valores concretos em  $\Delta_D$ , denominados indivíduos concretos, sendo  $I_A$  e  $I_D$  disjuntos. O mesmo ocorre para o conjunto de relacionamentos:  $R_A$  denota o conjunto de relacionamentos entre indivíduos abstratos, denominados relacionamentos abstratos, e  $R_D$  denota o conjunto de atributos de tipos de dados que relacionam indivíduos abstratos a valores concretos, sendo  $R_A$  e  $R_D$  também disjuntos. Sejam  $a \in I_A$ ,  $x, x_i \in I_D$ ,  $T, T_i \in R_D$ ,  $1 \leq i \leq n$  e  $E \in \mathbf{Dexp}(G)$ , a função de interpretação também é estendida para definir a semântica dos atributos de tipos de dados e dos construtores de conceito adicionais (PAN, 2007):

- $T^{\mathcal{I}} : \Delta^{\mathcal{I}} \rightarrow \Delta_D$ ;
- $(\exists T_1, \dots, T_n.E)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \exists \langle a, x_i \rangle \in T_i^{\mathcal{I}} \text{ e } \langle x_1, \dots, x_n \rangle \in E^D, \text{ com } 1 \leq i \leq n\}$ ;
- $(\forall T_1, \dots, T_n.E)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \forall \langle a, x_i \rangle \in T_i^{\mathcal{I}} \rightarrow \langle x_1, \dots, x_n \rangle \in E^D, \text{ com } 1 \leq i \leq n\}$ .

Utilizando esses construtores adicionais, é possível definir conceitos a partir das expressões de tipos de dados. Por exemplo, pode-se definir o conceito:

$$\text{ItemPequeno} \equiv \text{Item} \sqcap \exists \text{soma, altura, largura, profundidade.menorQue15} \quad (3.1)$$

onde o predicado personalizado *menorQue15* possui extensão  $X$ :

$$X(\text{menorQue15}) = \{\langle i_0, i_1, i_2, i_3 \rangle \in V(\text{INT}^+)^4 \mid i_0 = i_1 + i_2 + i_3 \wedge i_0 < 15\} \quad (3.2)$$

Pan (2007) também definiu uma arquitetura flexível para raciocínio em DLs com suporte a grupos de tipos de dados  $G$ , ilustrada na Figura 3.1. A principal característica dessa arquitetura é separação entre o procedimento de raciocínio das DLs baseado em *tableau*, realizado pelo raciocinador de conceitos, e o processamento das expressões de tipos de dados, realizado pelo raciocinador de tipos de dados. Existem dois tipos de raciocinadores de tipos de dados: gerenciador de tipos de dados e verificador de tipos de dados. Assim, os tipos de dados são tratados independentemente da parte abstrata ou simbólica, o que confere flexibilidade para adicionar novos tipos de dados e predicados concretos, de acordo com a necessidade do domínio.

Nessa arquitetura, o algoritmo baseado em *tableau* invoca o raciocinador de tipos de dados somente quando necessário, ou seja, quando são utilizados construtores de conceito contendo as expressões de tipos de dados  $E \in \mathbf{Dexp}(G)$  listadas na Tabela 3.3. Para tanto, o *tableau* é estendido com *nós de tipos de dados* que correspondem a nós-folha não rotulados, além dos

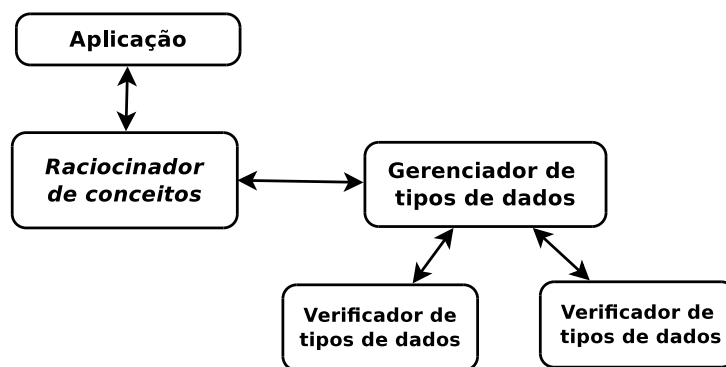


Figura 3.1: Arquitetura para raciocínio proposta por Pan (2007).

nós rotulados, denominados nós abstratos. Um nó de tipo de dado  $v$  é chamado de *sucessor- $T$*  de  $x$  se  $T \in \mathcal{L}(\langle x, v \rangle)$ , onde  $T$  é uma propriedade concreta e  $x$  é um indivíduo abstrato. Uma tupla de nós de tipos de dados  $\langle v_1, \dots, v_n \rangle$  é chamada de *sucessor- $\langle T_1, \dots, T_n \rangle$*  de  $x$  se para todo  $1 \leq i \leq n$ ,  $v_i$  é um sucessor- $T_i$  de  $x$ . Para armazenar as expressões de tipos de dados  $E$  que devem ser satisfeitas com relação aos sucessores- $\langle T_1, \dots, T_n \rangle$  de um nó abstrato  $x$ , é utilizado um conjunto  $DC(x)$  que contém elementos da forma:

$$(\langle v_1, \dots, v_n \rangle, E) \text{ ou } (\langle v_{i1}, \dots, v_{in} \rangle, \langle v_{j1}, \dots, v_{jn} \rangle, \neq) \quad (3.3)$$

onde  $\langle v_1, \dots, v_n \rangle, \langle v_{i1}, \dots, v_{in} \rangle, \langle v_{j1}, \dots, v_{jn} \rangle$  são tuplas de sucessores de tipo de dado de  $x$ . Também são definidas regras de expansão adicionais, chamadas de regras- $G$  (Tabela 3.4), para expandir o *tableau* e armazenar as expressões de tipos de dados  $E$  em  $DC(x)$ .

Tabela 3.4: Regras- $G$  para tratar expressões de tipos de dados (PAN, 2007).

<i>regra</i> – $\exists_p$	se $\exists T_1, \dots, T_n. E \in \mathcal{L}(x)$ e não existe um sucessor- $\langle T_1, \dots, T_n \rangle \langle v_1, \dots, v_n \rangle$ de $x$ tal que $(\langle v_1, \dots, v_n \rangle, E) \in DC(x)$ então crie um sucessor- $\langle T_1, \dots, T_n \rangle \langle v_1, \dots, v_n \rangle$ de $x$ com $\mathcal{L}(\langle x, v_i \rangle) = \{T_i\}$ , $1 \leq i \leq n$ e $DC(x) \longrightarrow DC(x) \cup \{(\langle v_1, \dots, v_n \rangle, E)\}$
<i>regra</i> – $\forall_p$	se $\forall T_1, \dots, T_n. E \in \mathcal{L}(x)$ e existe algum sucessor- $\langle T_1, \dots, T_n \rangle \langle v_1, \dots, v_n \rangle$ de $x$ tal que $(\langle v_1, \dots, v_n \rangle, E) \notin DC(x)$ então $DC(x) \longrightarrow DC(x) \cup \{(\langle v_1, \dots, v_n \rangle, E)\}$

Além das regras- $G$ , uma nova condição de *clash* é adicionada: um *tableau* contém um *clash* se, para algum nó abstrato  $x$ ,  $DC(x)$  é insatisfável. Por meio das regras- $G$ , o algoritmo *tableau* invoca o raciocinador de tipos de dados como um subprocedimento para verificar a satisfazibilidade de  $DC(x)$ . Assim,  $DC(x)$  atua como interface entre o algoritmo baseado em *tableau* e os raciocinadores de tipos de dados, para o nó abstrato  $x$ . As expressões de tipos de dados não são processadas pelo algoritmo baseado em *tableau*, portanto não afetam a estrutura do *tableau*, permitindo a separação entre o raciocínio da DL e o raciocínio de grupos de tipos de dados.

O gerenciador de tipos de dados, um dos tipos de raciocinador de tipos de dados, trabalha entre o raciocinador de conceitos e um conjunto de verificadores de tipos de dados. Ele é responsável por reduzir o problema de satisfazibilidade de conjunções de expressões de tipos de dados ao problema de satisfazibilidade de várias conjunções de predicados de tipos de dados. Ou seja, o gerenciador de tipos de dados decompõe as expressões de tipos de dados contidas em  $DC(x)$  para disjunções contendo conjunções mais simplificadas formadas por predicados atômicos, subdivididas por tipo de dado primitivo. Além disso, ele realiza algumas verificações de satisfazibilidade que independem da avaliação dos predicados. Por exemplo, verifica se existe alguma conjunção  $p \wedge \bar{p}$  em  $DC(x)$  ou se dois predicados, definidos por diferentes tipos de dados primitivos, compartilham variáveis.

Os verificadores de tipos de dados, por sua vez, verificam a satisfazibilidade de conjunções de predicados de tipos de dados definidos sobre um tipo de dado primitivo. Ou seja, cada verificador de tipo de dado é específico para tratar um tipo de dado primitivo. Como os tipos de dados primitivos são disjuntos por definição, os verificadores podem trabalhar de forma independente, sendo que os resultados de suas verificações são reunidos pelo gerenciador de tipos de dados, que então retorna o resultado final sobre a satisfazibilidade de  $DC(x)$  para o raciocinador de conceitos.

### 3.4 Ontologias Fuzzy: Definição e Caracterização

Embora ontologias fundamentadas por Lógicas de Descrição sejam capazes de expressar o significado das características mais comuns da representação do conhecimento, elas são menos apropriadas para os domínios nos quais a informação a ser representada caracteriza-se pela *incerteza* e/ou *imprecisão* (LUKASIEWICZ; STRACCIA, 2008). Por exemplo, informação *incerta* pode ser da forma *João é um professor com grau de certeza 0.3 ou um aluno com grau de certeza 0.7*, ou seja, João ou é um professor ou um aluno, porém é mais provável que seja um aluno. Informação *vaga* ou *imprecisa*, por sua vez, pode ser da forma *João é alto com grau de verdade 0.9*, ou seja, João é *quase* alto. A representação e o raciocínio de conhecimento incerto e/ou impreciso proporcionam mais expressividade às ontologias por meio do tratamento de informações incompletas e da intensidade associada a fatos, conceitos e relacionamentos.

Um exemplo clássico que esclarece a diferença entre incerteza e imprecisão é o exemplo de uma garrafa com líquido, citado por Lukasiewicz e Straccia (2008). Em termos de valores de verdade binários, uma garrafa é vista como cheia ou vazia. Porém, para descrever a quantidade de líquido presente na garrafa, pode-se dizer, por exemplo, que a garrafa está *meio cheia*. Neste

caso, o conceito *garrafa cheia* é considerado *impreciso*, e o grau de verdade associado reflete a quantidade de líquido na garrafa, constatado como um fato. Por outro lado, quando não se sabe se a garrafa está cheia ou vazia (informação incompleta), tendo em vista que somente um desses dois estados é o verdadeiro, pode-se expressar que existe uma *probabilidade* de que a garrafa esteja cheia, caracterizando *incerteza*. Portanto, expressar que uma garrafa esteja meio cheia não significa necessariamente que a probabilidade de que a garrafa esteja cheia é de 0.5. Especificamente, nesta tese de doutorado, o foco está voltado ao tratamento de imprecisão em ontologias, que é um importante aspecto para expressar de forma apropriada alguns domínios de aplicação e elementos da linguagem humana, que possui muitas características imprecisas.

Para tratar informações vagas ou imprecisas, a teoria de conjuntos *fuzzy* e lógica *fuzzy* (ZADEH, 1965; KLIR; YUAN, 1995; PEDRYCZ; GOMIDE, 1998) proveem um formalismo que permite tanto representar como também raciocinar sobre conhecimento impreciso. Com a popularidade das Lógicas de Descrição como formalismo de ontologias no contexto da *Web Semântica*, aliada à necessidade pelo tratamento de imprecisão em diversos domínios, muitas pesquisas têm sido realizadas para a extensão de Lógicas de Descrição considerando imprecisão. Neste sentido, as Lógicas de Descrição *fuzzy* (*Fuzzy Description Logics - fuzzy DLs*) (STRACCIA, 2001) foram definidas com base na teoria de conjuntos *fuzzy* e lógica *fuzzy*, sendo um dos principais formalismos para fundamentar as ontologias *fuzzy* formais e seus serviços de inferência. A próxima seção descreve as principais características das *fuzzy DLs*, assim como tarefas de raciocínio e a teoria de tipos de dados *fuzzy*, que permite expressar termos linguísticos como predicados *fuzzy*.

### 3.5 Lógicas de Descrição Fuzzy

Na literatura, existem diversas pesquisas sobre *fuzzy DLs*, com diferentes níveis de expressividade em função das características de imprecisão incorporadas. Uma das primeiras extensões, proposta por Straccia (2001), é *fuzzy ALC*, que combina a DL *ALC* com a lógica *fuzzy* de Zadeh. A *fuzzy DL f-SHIN* (STOILLOS et al., 2006, 2007, 2008) adiciona construtores e axiomas a *fuzzy ALC*, uma vez que a DL *SHIN* é mais expressiva que *ALC*. Existem também extensões que consideram tipos de dados concretos *fuzzy*, como *fuzzy SHOIN(D)* (STRACCIA, 2006b), *fuzzy ALCF(D)* (BOBILLO; STRACCIA, 2009), *fuzzy SROIQ(D)* (BOBILLO et al., 2009a), *F-ALC(G)* (WANG; MA, 2008), *F-SHOIQ(G)* (WANG et al., 2008) e *F-SROIQ(G)* (YAN; ZHANG; MA, 2012).

Para descrever os fundamentos do formalismo de representação e as tarefas de inferências

típicas de *Fuzzy DLs*, as próximas seções baseiam-se em *Fuzzy  $\mathcal{ALC}$*  (STRACCIA, 2001). Além disso, para explicar a extensão *fuzzy* de tipos de dados concretos em DLs, é considerada a *Fuzzy DL F- $\mathcal{ALC}(G)$*  e sua arquitetura de raciocínio (WANG; MA, 2008; WANG et al., 2008; WANG; MA; YIN, 2009; YAN; ZHANG; MA, 2012).

### 3.5.1 Sintaxe e Semântica de *Fuzzy $\mathcal{ALC}$*

Em *fuzzy  $\mathcal{ALC}$* , um conceito  $C$  é interpretado como um conjunto *fuzzy* e asserções de conceito do tipo  $a : C$  passam a ter um grau de verdade em  $[0, 1]$ , definido como o grau de pertinência do indivíduo  $a$  no conjunto *fuzzy*  $C$ . Desta forma, a interpretação de  $\mathcal{ALC}$  é estendida para uma interpretação *fuzzy*  $\mathcal{I}$ , que consiste de um conjunto domínio não vazio  $\Delta^{\mathcal{I}}$  e uma função de interpretação que mapeia:

- Um conceito  $C$  para uma função de pertinência  $C^{\mathcal{I}} : \Delta^{\mathcal{I}} \rightarrow [0, 1]$ ;
- Um relacionamento  $R$  para uma função de pertinência  $R^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \rightarrow [0, 1]$ .

Adicionalmente, a função de interpretação é estendida para construtores de conceito, conforme ilustrado na Tabela 3.5. Os construtores de  $\mathcal{ALC}$  são estendidos segundo as operações da lógica *fuzzy* de Zadeh correspondentes à negação, interseção, união e implicação (LUKASIEWICZ; STRACCIA, 2008). Os quantificadores existencial ( $\exists$ ) e universal ( $\forall$ ) são interpretados como uma disjunção e uma conjunção sobre os elementos do domínio  $\Delta^{\mathcal{I}}$ , razão do uso das operações de supremo (sup) e ínfimo (inf) em  $(\exists R.C)^{\mathcal{I}}$  e  $(\forall R.C)^{\mathcal{I}}$ , respectivamente.

**Tabela 3.5: Construtores de conceito estendidos de *fuzzy  $\mathcal{ALC}$*  (STRACCIA, 2001).**

Nome	Sintaxe	Semântica
<i>Conceito universal</i>	$\top$	$\top^{\mathcal{I}}(x) = 1$
<i>Conceito vazio</i>	$\perp$	$\perp^{\mathcal{I}}(x) = 0$
<i>Negação</i>	$\neg C$	$(\neg C)^{\mathcal{I}}(x) = 1 - C^{\mathcal{I}}(x)$
<i>Interseção</i>	$C \sqcap D$	$(C \sqcap D)^{\mathcal{I}}(x) = \min\{C^{\mathcal{I}}(x), D^{\mathcal{I}}(x)\}$
<i>União</i>	$C \sqcup D$	$(C \sqcup D)^{\mathcal{I}}(x) = \max\{C^{\mathcal{I}}(x), D^{\mathcal{I}}(x)\}$
<i>Restrição de valor</i>	$\forall R.C$	$(\forall R.C)^{\mathcal{I}}(x) = \inf_{y \in \Delta^{\mathcal{I}}} \max\{1 - R^{\mathcal{I}}(x, y), C^{\mathcal{I}}(y)\}$
<i>Quantificação existencial</i>	$\exists R.C$	$(\exists R.C)^{\mathcal{I}}(x) = \sup_{y \in \Delta^{\mathcal{I}}} \min\{R^{\mathcal{I}}(x, y), C^{\mathcal{I}}(y)\}$

Em *fuzzy  $\mathcal{ALC}$* , uma ABox *fuzzy* contém asserções *fuzzy* de conceito e relacionamento com o formato  $\langle \alpha \bowtie n \rangle$ , onde  $\alpha$  é uma asserção de  $\mathcal{ALC}$ ,  $\bowtie \in \{<, \leq, >, \geq\}$  e  $n \in [0, 1]$ . Uma asserção  $\langle \alpha \geq n \rangle$ , chamada de restrição *fuzzy*, restringe o grau de verdade de  $\alpha$  para que seja maior ou igual a  $n$ . Formalmente, uma interpretação  $\mathcal{I}$  satisfaz uma asserção  $\langle a : C \geq n \rangle$  se e somente se  $C^{\mathcal{I}}(a^{\mathcal{I}}) \geq n$ , de forma análoga para os demais operadores  $\bowtie$ . Para asserções de relacionamento, uma interpretação  $\mathcal{I}$  satisfaz  $\langle (a, b) : R \geq n \rangle$  se e somente se  $R^{\mathcal{I}}(a^{\mathcal{I}}, b^{\mathcal{I}}) \geq n$ , de



forma análoga para os demais operadores  $\bowtie$ . A combinação de ambas  $\langle \alpha \geq m \rangle$  e  $\langle \alpha \leq n \rangle$ , com  $m \leq n$ , restringem o grau de verdade de  $\alpha$  no intervalo  $[m, n]$ .

Straccia (2001) também define extensões de axiomas terminológicos *fuzzy* de uma TBox  $\mathcal{T}$  com base na inclusão e na equivalência de conjuntos *fuzzy*. Desta forma, uma interpretação *fuzzy*  $\mathcal{I}$  satisfaz axiomas de inclusão de conceito  $C \sqsubseteq D$  se e somente se  $C^{\mathcal{I}}(x) \leq D^{\mathcal{I}}(x), \forall x \in \Delta^{\mathcal{I}}$  e de definição de conceito  $C \equiv D$  se e somente se  $C^{\mathcal{I}}(x) = D^{\mathcal{I}}(x), \forall x \in \Delta^{\mathcal{I}}$ . Note que, nesta definição de TBox, um axioma de inclusão de conceitos é uma subsunção *crisp* de conceitos *fuzzy*. Existem outras *fuzzy* DLs, como as descritas por Lukasiewicz e Straccia (2008) e Bobillo e Straccia (2009), que definem uma subsunção *fuzzy* de conceitos do tipo  $\langle C \sqsubseteq D, n \rangle$ , onde  $C, D$  são conceitos *fuzzy* e  $n \in (0, 1]$ . Neste caso, todas as instâncias do conceito  $C$  são instâncias do conceito  $D$  no mínimo com um grau  $n$ . De acordo com Stoilos, Stamou e Pan (2010), não é muito claro como a subsunção *fuzzy* de conceitos poderia ser usada de um ponto de vista prático, pois é difícil para os especialistas do domínio modelarem axiomas de subsunção *fuzzy* consistentes. Por esta razão, e também para manter a representação mais simplificada, será considerada a subsunção *crisp* de conceitos *fuzzy* nos axiomas de inclusão de conceitos.

A base de conhecimento de *fuzzy*  $\mathcal{ALC}$ , denotada por  $\mathcal{K}$ , é um conjunto finito de asserções *fuzzy* (ABox) e axiomas terminológicos *fuzzy* (TBox), sendo que um conceito atômico aparece no máximo uma única vez do lado esquerdo de definições de conceito acíclicas (TBox definitiva). Uma interpretação *fuzzy*  $\mathcal{I}$  satisfaz  $\mathcal{K}$  se e somente se ela satisfaz cada elemento de  $\mathcal{K}$ . Um conceito  $C$  é satisfatível se e somente se existe uma interpretação *fuzzy*  $\mathcal{I}$  e um indivíduo  $x \in \Delta^{\mathcal{I}}$  tal que  $C^{\mathcal{I}}(x) > 0$ . A noção da satisfazibilidade de um axioma *fuzzy* por uma interpretação *fuzzy*  $\mathcal{I}$  é definida como:

$$\begin{aligned} \mathcal{I} \models A \sqsubseteq C & \quad \text{se e somente se } \forall a \in \Delta^{\mathcal{I}}, A^{\mathcal{I}}(a) \leq C^{\mathcal{I}}(a) \\ \mathcal{I} \models A \equiv C & \quad \text{se e somente se } \forall a \in \Delta^{\mathcal{I}}, A^{\mathcal{I}}(a) = C^{\mathcal{I}}(a) \\ \mathcal{I} \models \langle \alpha \bowtie n \rangle & \quad \text{se e somente se } \alpha^{\mathcal{I}} \bowtie n, \text{ onde } \alpha \text{ é uma asserção de } \mathcal{ALC} \end{aligned}$$

Uma base de conhecimento *fuzzy*  $\mathcal{K}$  implica uma asserção *fuzzy*  $\psi$  ( $\mathcal{K} \models \psi$ ) se e somente se toda interpretação  $\mathcal{I}$  de  $\mathcal{K}$  também satisfizer  $\psi$ . Além disso,  $\mathcal{K}$  implica um axioma  $C \sqsubseteq D$  ( $\mathcal{K} \models C \sqsubseteq D$ ) se e somente se toda interpretação  $\mathcal{I}$  de  $\mathcal{K}$  satisfizer  $C^{\mathcal{I}}(x) \leq D^{\mathcal{I}}(x), \forall x \in \Delta^{\mathcal{I}}$ .

### 3.5.2 Inferências Típicas de *Fuzzy* DLs

Além das tarefas típicas de raciocínio em DLs (Seção 3.3.2), as *Fuzzy* DLs possuem alguns problemas de inferência específicos, como a verificação do melhor limite de grau de verdade e a recuperação *fuzzy*. Straccia (2006a), Lukasiewicz e Straccia (2008) listam as principais tarefas de raciocínio para uma base de conhecimento *fuzzy*  $\mathcal{K}$ :

- *Consistência de base de conhecimento fuzzy*: verifica se  $\mathcal{K}$  é consistente, ou seja, se existe uma interpretação ou modelo  $\mathcal{I}$  que satisfaz todos os elementos de  $\mathcal{K}$  ( $\mathcal{I} \models \mathcal{K}$ );
- *Satisfazibilidade de conceitos*: verifica se um conceito  $C$  é satisfatível se e somente se existe uma interpretação  $\mathcal{I}$  e um indivíduo  $x \in \Delta^{\mathcal{I}}$  tal que  $C^{\mathcal{I}}(x) > 0$ ;
- *Subsunção*: computa a taxonomia de conceitos, de modo que  $C \sqsubseteq D$  com relação a  $\mathcal{K}$  se e somente se  $C^{\mathcal{I}}(x) \leq D^{\mathcal{I}}(x), \forall x \in \Delta^{\mathcal{I}}$  para qualquer interpretação  $\mathcal{I}$  de  $\mathcal{K}$  ( $\mathcal{K} \models C \sqsubseteq D$ );
- *Verificação de instâncias*: verifica se uma asserção fuzzy  $\psi$  ( $\langle a : C \bowtie n \rangle$  ou  $\langle (a, b) : R \bowtie n \rangle$ ) é uma implicação lógica de  $\mathcal{K}$  se toda interpretação  $\mathcal{I}$  que satisfaz  $\mathcal{K}$  também satisfaz  $\psi$  ( $\mathcal{K} \models \psi$ );
- *Melhor limite de grau de verdade (best truth value bound - BTVB)*: computa o melhor limite inferior e superior de graus de verdade para uma asserção  $\alpha$  que não envolva transitividade. O maior limite inferior (*greatest lower bound*) de  $\alpha$  com relação a  $\mathcal{K}$  é definido como  $glb(\mathcal{K}, \alpha) = \sup\{n \mid \mathcal{K} \models \langle \alpha \geq n \rangle\}$ . O menor limite superior (*least upper bound*) de  $\alpha$  com relação a  $\mathcal{K}$  é definido como  $lub(\mathcal{K}, \alpha) = \inf\{n \mid \mathcal{K} \models \langle \alpha \leq n \rangle\}$ ;
- *Recuperação Fuzzy*: Recuperar os primeiros  $k$  indivíduos (recuperação top- $k$ ) que instanciam um conceito  $C$ , classificados de acordo com o melhor limite de grau de verdade, resultando no conjunto  $\{x, glb(\mathcal{K}, x : C)\}$ .

Straccia (2001) demonstra que as tarefas de raciocínio típicas de *fuzzy* DLs podem ser reduzidas para a verificação de consistência da bases de conhecimento *fuzzy*, assim como ocorre nas DLs tradicionais. A seguir, são descritas algumas formas de reduções encontradas na literatura (STRACCIA, 2001, 2006a; STOILOS; STAMOU; PAN, 2010):

- *Satisfazibilidade de conceitos*: um conceito  $C$  é satisfatível se e somente se existe pelo menos uma interpretação  $\mathcal{I}$  para um novo indivíduo  $x \in \Delta^{\mathcal{I}}$  tal que a base de conhecimento  $\mathcal{K}' = \mathcal{K} \cup \{\langle x : C > 0 \rangle\}$  é consistente. Um conceito  $C$  é *n-satisfatível* se e somente se existe pelo menos uma interpretação  $\mathcal{I}$  para um novo indivíduo  $x \in \Delta^{\mathcal{I}}$  tal que a base de conhecimento  $\mathcal{K}' = \mathcal{K} \cup \{\langle x : C \geq n \rangle\}$  é consistente;
- *Subsunção*:  $\mathcal{K} \models C \sqsubseteq D$  se e somente se  $\mathcal{K}' = \mathcal{K} \cup \{\langle x : C \geq n \rangle, \langle x : D < n \rangle\}$  é *inconsistente* para um novo indivíduo  $x \in \Delta^{\mathcal{I}}$  e para todo  $n \in [0, 1]$ . Ou seja, não existe uma interpretação  $\mathcal{I}$  que satisfaça  $\mathcal{K}'$ . Note que verificar a inconsistência para  $\mathcal{K}'$  e todo grau  $n \in [0, 1]$  é um problema inviável em termos computacionais. Uma simplificação demonstrada por Straccia (2001) para a classe específica de *fuzzy* DLs  $f_{KD}$  estabelece que  $\mathcal{K} \models C \sqsubseteq D$  se

e somente se  $\mathcal{K}' = \mathcal{K} \cup \{\langle x : C \geq n \rangle, \langle x : D < n \rangle\}$  é inconsistente para cada  $n \in \{n_1, n_2\}$ , onde  $n_1 \in (0, 0.5]$  e  $n_2 \in (0.5, 1]$ . A notação  $f_{KD}$  refere-se a *fuzzy* DLs que utilizam o complemento de Lukasiewicz ( $1 - x$ ), a t-norma e a s-norma de Gödel ( $\min(x, y)$  e  $\max(x, y)$ , respectivamente), e a implicação de Kleene-Dienes ( $\max(1 - x, y)$ ) (STOILLOS; STAMOU; PAN, 2010);

- *Verificação de instâncias*:  $\mathcal{K} \models \langle a : C \geq n \rangle$  se e somente se  $\mathcal{K}' = \mathcal{K} \cup \{\langle a : C < n \rangle\}$  é inconsistente. Se existir um modelo  $\mathcal{I}$  para  $\mathcal{K}'$ , ele representa um contra-exemplo para a asserção  $\langle a : C \geq n \rangle$ ;
- *Melhor limite de grau de verdade*: uma abordagem proposta por Straccia (2001) para resolver o *glb* de uma asserção  $\alpha$  consiste em realizar sucessivas verificações de instâncias considerando um conjunto de graus de pertinência  $N^{\mathcal{K}} = \{0, 0.5, 1\} \cup \{n : \langle \psi \triangleright n \rangle \in \mathcal{K}\} \cup \{1 - n : \langle \psi \triangleleft n \rangle \in \mathcal{K}\}$  presentes na ABox de  $\mathcal{K}$ . Em sequência, são realizadas verificações de instâncias para  $\langle \alpha \geq n \rangle$  para todo  $n \in N^{\mathcal{K}}$  de modo a obter o *glb* como  $glb(\mathcal{K}, \alpha) = \sup\{n \mid \mathcal{K} \models \alpha \geq n\}$ . Para facilitar a busca por  $n$ , pode-se realizar verificações de instâncias considerando uma busca binária sobre o conjunto  $N^{\mathcal{K}}$  devidamente ordenado (STRACCIA, 2001). Outra abordagem proposta por Straccia (2005a) considera o uso de *Mixed Integer Linear Programming* (MILP) para definir restrições lineares de modo que  $glb(\mathcal{K}, \alpha)$  é dado pelo mínimo  $x$  tal que  $\mathcal{K} \cup \{\alpha \leq x\}$  seja consistente. Segundo Straccia (2001), o *lub* pode ser definido a partir do *glb* por  $lub(\mathcal{K}, \alpha) = 1 - glb(\mathcal{K}, \neg\alpha)$ ;
- *Recuperação Fuzzy*: uma abordagem não otimizada, citada por Straccia (2006a), computa o  $glb(\mathcal{K}, x : C)$  para todo  $x \in \Delta^{\mathcal{I}}$  e então ordena os indivíduos de acordo com o *glb* para selecionar os  $k$  primeiros indivíduos. Existem otimizações, como a proposta por Straccia (2006a), para viabilizar a recuperação *fuzzy* em bases de conhecimento com um grande número de indivíduos.

Considerando a tarefa de verificação de consistência de bases de conhecimento *fuzzy*, Straccia (2001) desenvolveu uma extensão do algoritmo baseado em *tableau* (Seção 3.3.3) para *fuzzy*  $\mathcal{ALC}$ , que pode ser utilizado como procedimento de inferência básico em *Fuzzy* DLs.

### 3.5.3 Raciocínio Baseado em *Tableau Fuzzy*

As principais extensões do algoritmo baseado em *tableau* definidas por Straccia (2001) estão relacionadas à representação do *tableau*, às regras de expansão e às condições que configuram uma contradição, pois devem levar em consideração a semântica estendida dos construtores

de conceito e os graus de verdade das asserções *fuzzy*. A representação do *tableau* é estendida para um *tableau fuzzy* de modo que o conjunto  $\mathcal{L}(x)$  de rótulos de um nó  $x$  contenha um conceito  $\langle C_i \bowtie n \rangle$  se  $C_i^{\mathcal{I}}(x^{\mathcal{I}}) \bowtie n$ . Cada aresta  $\langle x, y \rangle$  entre os nós  $x$  e  $y$  deve ser rotulada com um relacionamento  $\langle R_i \bowtie n \rangle$  caso  $R_i^{\mathcal{I}}(x^{\mathcal{I}}, y^{\mathcal{I}}) \bowtie n$ , para  $\bowtie \in \{<, \leq, >, \geq\}$ . Neste caso, o nó  $y$  é chamado de *sucessor- $R_i$*  do nó  $x$ .

As regras de expansão são estendidas de acordo com a semântica dos construtores de *fuzzy*  $\mathcal{ALC}$  e dos operadores relacionais  $\bowtie$ , adicionando rótulos aos nós e arestas do *tableau*. As situações que configuram contradições também são modificadas, considerando intervalos de graus de verdade disjuntos em asserções envolvendo um mesmo conceito ou relacionamento. Por exemplo, caso um nó  $x$  contenha dois rótulos  $\langle x : C > n \rangle$  e  $\langle x : C < m \rangle$ , com  $n > m$ , há mais de um intervalo possível de graus de verdade para a asserção  $x : C$ , o que configura uma contradição na base de conhecimento.

O algoritmo estendido considera algumas suposições quanto à notação. As descrições de conceito devem estar na NNF, assim como no algoritmo original. O símbolo  $\triangleleft$  corresponde a um dos operadores relacionais  $\{<, \leq\}$  e  $\triangleright$  a um dos operadores  $\{>, \geq\}$ , enquanto  $\bowtie$  se refere a qualquer um desses operadores relacionais. Os símbolos  $\triangleleft^{-1}, \triangleright^{-1}, \bowtie^{-1}$  denotam suas reflexões, sendo que  $\triangleleft^{-1}$  indica a substituição de  $<$  por  $>$ ,  $\leq$  por  $\geq$ , de modo análogo para  $\triangleright^{-1}$  (YAN; ZHANG; MA, 2012).

Para uma asserção  $\psi$  de *fuzzy*  $\mathcal{ALC}$ ,  $\psi^C$  denota um conjugado de  $\psi$ . De acordo com Straccia (2001), um *par conjugado* é um conjunto de duas inequações restringindo o grau de verdade de uma mesma asserção  $\alpha$  que não possuem uma sobreposição. A Tabela 3.6 mostra as condições em que os pares de asserções *fuzzy* são pares conjugados. Vale ressaltar que um conjugado de uma asserção *fuzzy* pode não ser único; por exemplo ambos  $\langle C(a) < 0.6 \rangle$  e  $\langle C(a) \leq 0.7 \rangle$  são conjugados de  $\langle C(a) \geq 0.8 \rangle$ .

**Tabela 3.6: Situações que definem pares conjugados (STRACCIA, 2001).**

	$\langle \alpha < m \rangle$	$\langle \alpha \leq m \rangle$
$\langle \alpha \geq n \rangle$	$n \geq m$	$n > m$
$\langle \alpha > n \rangle$	$n \geq m$	$n \geq m$

Iniciando por uma ABox  $\mathcal{A}$ , o algoritmo inicializa o *tableau fuzzy* da seguinte forma:

- Define um nó para cada indivíduo  $x$  que ocorre em  $\mathcal{A}$ , rotulado com um conjunto de rótulos  $\mathcal{L}(x) = \{\langle C \bowtie n \rangle\}$  para cada asserção *fuzzy*  $\langle C(x) \bowtie n \rangle$  em  $\mathcal{A}$ ;
- Define uma aresta  $\langle x, y \rangle$ , rotulada com um conjunto  $\mathcal{L}(\langle x, y \rangle) = \{\langle R \bowtie n \rangle\}$  para cada asserção  $\langle R(x, y) \bowtie n \rangle$  em  $\mathcal{A}$ .

Depois dessa inicialização, o algoritmo expande o *tableau fuzzy* pela aplicação das regras de expansão. Cada regra de expansão do algoritmo tradicional ( $\sqcap, \sqcup, \forall, \exists$ ), juntamente com a negação ( $\neg$ ), foi estendida para contemplar os operadores relacionais  $\triangleleft \in \{<, \leq\}$ ,  $\triangleright \in \{>, \geq\}$  e  $\bowtie$ , em função da semântica dos construtores. A Tabela 3.7 apresenta as regras de expansão de *fuzzy ALC* definidas por Straccia (2001).

**Tabela 3.7: Regras de expansão para *fuzzy ALC* extraídas de (STRACCIA, 2001).**

1) regra- $\neg$	se $\langle \neg C \bowtie n \rangle \in \mathcal{L}(x)$ então $\mathcal{L}(x) \rightarrow \mathcal{L}(x) \cup \{ \langle C \bowtie^{-1} 1 - n \rangle \}$
2) regra- $\sqcap \triangleright$	se $\langle C_1 \sqcap C_2 \triangleright n \rangle \in \mathcal{L}(x)$ então $\mathcal{L}(x) \rightarrow \mathcal{L}(x) \cup \{ \langle C_1 \triangleright n \rangle, \langle C_2 \triangleright n \rangle \}$
3) regra- $\sqcup \triangleleft$	se $\langle C_1 \sqcup C_2 \triangleleft n \rangle \in \mathcal{L}(x)$ então $\mathcal{L}(x) \rightarrow \mathcal{L}(x) \cup \{ \langle C_1 \triangleleft n \rangle, \langle C_2 \triangleleft n \rangle \}$
4) regra- $\sqcup \triangleright$	se $\langle C_1 \sqcup C_2 \triangleright n \rangle \in \mathcal{L}(x)$ então $\mathcal{L}(x) \rightarrow \mathcal{L}(x) \cup \{ \langle C_1 \triangleright n \rangle \}$ ou $\mathcal{L}(x) \rightarrow \mathcal{L}(x) \cup \{ \langle C_2 \triangleright n \rangle \}$
5) regra- $\sqcap \triangleleft$	se $\langle C_1 \sqcap C_2 \triangleleft n \rangle \in \mathcal{L}(x)$ então $\mathcal{L}(x) \rightarrow \mathcal{L}(x) \cup \{ \langle C_1 \triangleleft n \rangle \}$ ou $\mathcal{L}(x) \rightarrow \mathcal{L}(x) \cup \{ \langle C_2 \triangleleft n \rangle \}$
6) regra- $\forall \triangleright$	se (1) $\langle \forall R.C \triangleright n \rangle \in \mathcal{L}(x)$ e (2) $x$ tem um sucessor- $R$ y tal que $\psi^C \in \mathcal{L}(\langle x, y \rangle)$ com $\psi = \langle R, \triangleright^{-1}, 1 - n \rangle$ e $\langle C \triangleright n \rangle \notin \mathcal{L}(y)$ então $\mathcal{L}(y) \rightarrow \mathcal{L}(y) \cup \{ \langle C \triangleright n \rangle \}$
7) regra- $\exists \triangleleft$	se (1) $\langle \exists R.C \triangleleft n \rangle \in \mathcal{L}(x)$ e (2) $x$ tem um sucessor- $R$ y tal que $\psi^C \in \mathcal{L}(\langle x, y \rangle)$ com $\psi = \langle R \triangleleft n \rangle$ e $\langle C \triangleleft n \rangle \notin \mathcal{L}(y)$ então $\mathcal{L}(y) \rightarrow \mathcal{L}(y) \cup \{ \langle C \triangleleft n \rangle \}$
8) regra- $\exists \triangleright$	se (1) $\langle \exists R.C \triangleright n \rangle \in \mathcal{L}(x)$ e (2) $x$ não tem um sucessor- $R$ y tal que $\langle R \triangleright n \rangle \in \mathcal{L}(\langle x, y \rangle)$ e $\langle C \triangleright n \rangle \in \mathcal{L}(y)$ então crie um novo sucessor- $R$ y de $x$ tal que $\mathcal{L}(\langle x, y \rangle) = \{ \langle R \triangleright n \rangle \}$ e $\mathcal{L}(y) = \{ \langle C \triangleright n \rangle \}$
9) regra- $\forall \triangleleft$	se (1) $\langle \forall R.C \triangleleft n \rangle \in \mathcal{L}(x)$ e (2) $x$ não tem um sucessor- $R$ y tal que $\langle R \triangleleft^{-1} 1 - n \rangle \in \mathcal{L}(\langle x, y \rangle)$ e $\langle C \triangleleft n \rangle \in \mathcal{L}(y)$ então crie um novo sucessor- $R$ y de $x$ tal que $\mathcal{L}(\langle x, y \rangle) = \{ \langle R \triangleleft^{-1} 1 - n \rangle \}$ e $\mathcal{L}(y) = \{ \langle C \triangleleft n \rangle \}$

É possível notar que as regras combinam a semântica de cada construtor de conceito ( $\neg, \sqcap, \sqcup, \forall, \exists$ ) com os operadores relacionais presentes em asserções *fuzzy*. A regra 1 trata a negação, complementando tanto o operador relacional ( $\bowtie^{-1}$ ) quanto o grau de verdade ( $1 - n$ ). As regras de 2 a 5 combinam a semântica dos construtores  $\sqcap, \sqcup$  com os operadores relacionais. Como a interpretação de  $C \sqcap D$  é definida pela operação de mínimo, ambos os graus de verdade de  $C$  e  $D$  devem ser maiores que  $n$ , no caso dos operadores do tipo  $\triangleright$ . Por outro lado, para os operadores do tipo  $\triangleleft$ , é suficiente que  $C$  ou  $D$  tenham grau de verdade menor que  $n$ . Esse raciocínio é análogo para as regras do construtor  $\sqcup$ , definido pela operação de máximo.

Nas regras 6 e 7,  $\psi^C$  corresponde a um conjugado de  $\psi$ , em que os intervalos de graus de verdade de  $\psi$  e  $\psi^C$  são disjuntos. Considerando a regra 6, para que  $\langle x : \forall R.C \triangleright n \rangle$  seja consistente, é suficiente que  $1 - R^{\mathcal{I}}(x, y)$  ou  $C^{\mathcal{I}}(y)$  sejam maiores que  $n$ , em função da semântica de  $(\forall R.C)^{\mathcal{I}}(x) = \inf_{y \in \Delta^{\mathcal{I}}} \max\{1 - R^{\mathcal{I}}(x, y), C^{\mathcal{I}}(y)\}$ . Como  $\langle 1 - R^{\mathcal{I}}(x, y) \triangleright n \rangle$  corresponde a  $\langle R^{\mathcal{I}}(x, y) \triangleright^{-1} 1 - n \rangle$ , as asserções de relacionamento  $\psi = \langle R(x, y) \triangleright^{-1} 1 - n \rangle$  tornam a asserção  $\langle x : \forall R.C \triangleright n \rangle$  consistente, independentemente de  $C^{\mathcal{I}}(y)$ , pela combinação das operações *max*

e *inf*. No entanto, na presença de conjugados de  $\psi$  ( $\psi^C$ ), é preciso garantir que  $\langle y : C \triangleright n \rangle$  para que a asserção  $\langle x : \forall R.C \triangleright n \rangle$  seja consistente. Portanto, neste caso, a regra 6 adiciona o rótulo  $\langle y : C \triangleright n \rangle$  para manter a consistência da base de conhecimento. Esse raciocínio é análogo para a regra 7, que considera a semântica do construtor  $\exists R.C$  e os operadores do tipo  $\triangleleft$ .

Com relação à regra 8 ( $\exists \triangleright$ ), ambas restrições  $\langle R(x,y) \triangleright n \rangle$  e  $\langle C(y) \triangleright n \rangle$  devem ocorrer, para algum  $x, y \in \Delta^{\mathcal{I}}$ , para que  $\langle x : \exists R.C \triangleright n \rangle$  seja consistente. Assim, caso essa situação não exista no *tableau*, a regra 8 deve adicioná-la na base de conhecimento para manter a consistência. Esse raciocínio é análogo para a regra 9, considerando a semântica do construtor  $\forall R.C$  e os operadores relacionais do tipo  $\triangleleft$ .

As situações que configuram contradições também são estendidas em função dos graus de verdade presentes nos rótulos dos nós e das arestas do *tableau fuzzy*. Segundo Straccia (2001),  $\mathcal{L}(x)$  contém uma contradição se  $\mathcal{L}(x)$  contiver um par conjugado (Tabela 3.6) ou se  $\mathcal{L}(x)$  contiver alguma das seguintes situações:

- $\langle \perp \geq n \rangle$ , onde  $n > 0$ ;
- $\langle \top \leq n \rangle$ , onde  $n < 1$ ;
- $\langle \perp > n \rangle$ ,  $\langle \top < n \rangle$ ,  $\langle C < 0 \rangle$ ,  $\langle C > 1 \rangle$ , onde  $n \in [0, 1]$ .

Por exemplo, caso um nó  $x$  contenha um par conjugado em  $\mathcal{L}(x)$  como  $\langle C > n \rangle$  e  $\langle C < m \rangle$ , com  $n > m$ , então existe mais de um intervalo possível de graus de verdade para  $x : C$ , o que configura uma contradição.

O algoritmo baseado em *tableau fuzzy* finaliza a execução quando o *tableau* está completo, ou seja, nenhuma regra de expansão pode ser mais aplicada, ou então existe algum nó  $x$  contendo uma contradição. Um *tableau fuzzy* é denominado *livre de contradição* se nenhum de seus nós contém uma contradição. Por fim, se as regras de expansão podem ser aplicadas de modo a resultar em um *tableau fuzzy* completo e livre de contradições, então a base de conhecimento verificada por ele é consistente. O *tableau fuzzy* resultante representa uma interpretação  $\mathcal{I}$  que prova que a  $\mathcal{K}$  é consistente. Caso o *tableau fuzzy* contenha alguma contradição, o algoritmo retorna que a base de conhecimento é inconsistente.

### 3.5.4 Tipos de Dados Concretos Fuzzy

Os tipos de dados concretos em DLs, descritos na Seção 3.3.4, também são estendidos para tipos de dados concretos *fuzzy*. Assim, propriedades concretas dos indivíduos podem

estar associadas tanto a valores precisos como também a valores imprecisos, definidos por conjuntos *fuzzy*. No caso de atributos definidos sobre domínios numéricos, podem ser utilizadas funções de pertinência parametrizadas como triangular, trapezoidal, entre outras. Por exemplo, o atributo *idade* de uma pessoa pode assumir valores imprecisos como *jovem*, *adulto* e *idoso*, cada qual com sua respectiva função de pertinência. A partir desses conceitos, é possível representar atributos com termos linguísticos como *idade jovem*, *velocidade muito rápida*, *estatura média*, que não possuem uma definição exata ou precisa.

Formalmente, uma teoria de tipos de dados *fuzzy*  $D = (\Delta^D, \cdot^D)$  consiste de um domínio de tipos de dados  $\Delta^D$  e uma função  $\cdot^D$ , que atribui uma relação *fuzzy*  $n$ -ária sobre  $\Delta^D$  para todo predicado de tipo de dado *fuzzy*  $n$ -ário (LUKASIEWICZ; STRACCIA, 2008). *Fuzzy* DLs com suporte a tipos de dados *fuzzy* também são denotadas com o sufixo ( $D$ ) adicionado ao nome da DL: *fuzzy*  $\mathcal{ALC}(D)$  (STRACCIA, 2005a, 2005b), *fuzzy*  $\mathcal{ALCF}(D)$  (BOBILLO; STRACCIA, 2009), *fuzzy*  $\mathcal{SHOIN}(D)$  (STRACCIA, 2006b; STOILOS; STAMOU; PAN, 2010), *fuzzy*  $\mathcal{SROIQ}(D)$  (BOBILLO; STRACCIA, 2010). Existem motores de inferência de *fuzzy* DLs que oferecem suporte aos tipos de dados *fuzzy*, como *fuzzyDL* (BOBILLO; STRACCIA, 2008), que trata *fuzzy*  $\mathcal{SHIF}(D)$ , e DeLorean (BOBILLO; DELGADO; GÓMEZ-ROMERO, 2012). com suporte a *fuzzy*  $\mathcal{SROIQ}(D)$ .

Um exemplo de predicado de tipo de dado *fuzzy* é  $Jovem \subseteq \Delta^D$ , que denota o grau com o qual o valor da idade é considerado jovem (LUKASIEWICZ; STRACCIA, 2008). O significado desse predicado unário pode ser definido por uma função de pertinência trapezoidal parametrizada do tipo *left shoulder* como  $Jovem(x) = ls(x; 10, 30)$  (BOBILLO; STRACCIA, 2008). A partir desse predicado *fuzzy*, é possível modelar conceitos como  $Pessoa \sqcap \exists idade.Jovem$ , que envolvem propriedades concretas e valores imprecisos.

Os tipos de dados e predicados concretos personalizados também são estendidos considerando imprecisão. Neste sentido, grupos de tipos de dados e expressões de tipos de dados definidos por Pan (2007) são estendidos para grupos de tipos de dados *fuzzy* e expressões de tipos de dados *fuzzy* (WANG; MA, 2008; WANG et al., 2008; WANG; MA; YIN, 2009; YAN; ZHANG; MA, 2012). De acordo com Wang e Ma (2008), um grupo de tipos de dados *fuzzy* é um grupo de predicados *fuzzy* embutidos, definidos sobre um conjunto de tipos de dados base, provendo um formalismo unificado para tratar tipos de dados *fuzzy* e predicados de tipos de dados *fuzzy*. Para denotar o suporte a grupos de tipos de dados *fuzzy*, também se adiciona o sufixo ( $G$ ) ao nome da DL, por exemplo,  $F\text{-}\mathcal{ALC}(G)$  (WANG; MA, 2008),  $F\text{-}\mathcal{SHOIN}(G)$  (WANG et al., 2008),  $F\text{-}\mathcal{SROIQ}(G)$  (YAN; ZHANG; MA, 2012). O motor de inferência FRESG (WANG; MA; YIN, 2009) oferece suporte a grupos de tipos de dados *fuzzy* com base

em  $F\text{-}\mathcal{ALC}(G)$ .

Segundo Wang e Ma (2008), um predicado de tipo de dado *fuzzy*  $p$  é caracterizado por uma aridade  $a(p)$ , ou por uma aridade mínima  $a_{min}(p)$  se  $p$  puder ter múltiplas aridades, e uma extensão de predicado  $X(p)$ . O predicado  $p$  é interpretado por  $(\Delta_D, \cdot^D)$ , onde  $\Delta_D$  é o domínio de tipos de dados *fuzzy* e  $\cdot^D$  mapeia  $p$  para uma função  $p^D : \Delta_D^n \rightarrow [0, 1]$ , que é uma relação *fuzzy*  $n$ -ária sobre  $\Delta_D$ . Cada predicado de tipo de dado *fuzzy*  $p$  é definido sobre um tipo de dado base  $d$  tal que  $X(p) \subseteq V(d)^{a(p)}$ , onde  $V(d)$  é conjunto de valores de  $d$ . Um tipo de dado base  $d$  é um tipo de dado primitivo, correspondendo aos tipos de dados comuns às linguagens de programação como números inteiros, números reais, tipos booleanos e cadeias de caracteres. Cada tipo de dado base possui espaço de valores disjunto dos demais, ou seja, para quaisquer dois tipos de dados base  $d_1, d_2$ ,  $V(d_1) \cap V(d_2) = \emptyset$ .

Conforme Wang et al. (2008), um *grupo de tipo de dado fuzzy*  $G$  é definido como uma tripla  $(\phi_G, D_G, dom)$ , onde  $\phi_G$  é um conjunto de predicados embutidos,  $D_G$  é o conjunto de tipos de dados base e  $dom(p)$  é o domínio do predicado de tipo de dado *fuzzy*  $p$ . Cada  $p \in \phi_G$  deve ser definido sobre um tipo de dado base  $d \in D_G$  tal que  $X(p) \subseteq dom(p)$ , onde  $dom(p) = V(d)^{a(p)}$ . Por exemplo, o predicado de tipo de dado *fuzzy* unário *pequeno* pode ser definido com a extensão:

$$X(\text{pequeno}) = \{x \in V(\text{real}) \mid \text{pequeno}^D(x) = \max(0, 1 - 0.00068x^2)\} \quad (3.4)$$

Em um grupo de tipos de dados *fuzzy*, expressões de tipos de dados *fuzzy* podem ser usadas para representar tipos de dados e predicados *fuzzy* personalizados. A Tabela 3.8 mostra a sintaxe e a semântica de algumas expressões de tipos de dados *fuzzy* definidas por Wang et al. (2008). O conjunto de expressões de tipo de dado *fuzzy* é denotado por  $\mathbf{Dexp}(G)$ .

**Tabela 3.8: Sintaxe e semântica de expressões de tipos de dados *fuzzy*.**

Expressão	Sintaxe	Semântica
expressões atômicas	$p$	$p^D(v_1, v_2, \dots, v_n) \rightarrow [0, 1]$
expressões negadas relativizadas	$\bar{p}$	$\bar{p}^D(v_1, v_2, \dots, v_n) = 1 - p^D(v_1, v_2, \dots, v_n)$
expressões conjuntivas	$E_1 \wedge \dots \wedge E_n$	$(E_1 \wedge \dots \wedge E_n)^D = (E_1)^D \wedge \dots \wedge (E_n)^D$
expressões disjuntivas	$E_1 \vee \dots \vee E_n$	$(E_1 \vee \dots \vee E_n)^D = (E_1)^D \vee \dots \vee (E_n)^D$

Um exemplo ilustrando o uso de expressões de tipos de dados *fuzzy* é descrito por Wang et al. (2008), considerando o contexto de uma loja virtual que precisa classificar seus itens de acordo com o tamanho definido em função da soma das medidas de altura, largura e comprimento. Ao modelar esse domínio com um grupo de tipo de dado *fuzzy*  $G = (\phi_G, D_G, dom)$ , os predicados *soma* e *pequeno* podem ser definidos como predicados embutidos em  $\phi_G$ . O predicado *soma* possui  $a_{min}(\text{soma}) = 3$  e extensão  $X(\text{soma}) = \{\langle i_0, i_1, i_2, \dots, i_{n-1} \rangle \in V(\text{real})^n \mid i_0 = i_1 + i_2 + \dots + i_{n-1}\}$ . O predicado *fuzzy* unário *pequeno* pode ser definido com a extensão



exemplificada na Equação 3.4. Assim, para definir o conceito de *item-pequeno*, os predicados *soma* e *pequeno* podem ser combinados por uma expressão conjuntiva, resultando no predicado de tipo de dado fuzzy personalizado  $p_{item-pequeno}$  com aridade  $a(p_{item-pequeno}) = 4$  e extensão:

$$X(p_{item-pequeno}) = \{ \langle i_0, i_1, i_2, i_3 \rangle \in V(real)^4 \mid soma(i_0, i_1, i_2, i_3) \wedge pequeno(i_0) \} \quad (3.5)$$

DLs com suporte a grupos de tipos de dados fuzzy também contêm *propriedades de tipos de dados fuzzy*, também consideradas *propriedades concretas*, que correspondem a relacionamentos entre indivíduos  $a \in \Delta^{\mathcal{I}}$  e valores concretos  $x \in \Delta_D$ , além de construtores de conceito adicionais que permitem descrever conceitos a partir de restrições sobre valores concretos usando expressões de tipos de dados fuzzy. Assim, o conjunto de indivíduos é composto por  $I_A$ , que denota o conjunto de indivíduos em  $\Delta^{\mathcal{I}}$  (indivíduos abstratos), e por  $I_D$ , que denota o conjunto de indivíduos que correspondem a valores de tipos de dados em  $\Delta_D$  (indivíduos concretos), sendo  $I_A$  e  $I_D$  disjuntos. O mesmo ocorre para o conjunto de relacionamentos:  $R_A$  denota o conjunto de relacionamentos entre indivíduos abstratos (relacionamentos abstratos), e  $R_D$  denota o conjunto de propriedades de tipos de dados fuzzy que relacionam indivíduos abstratos a valores concretos, sendo  $R_A$  e  $R_D$  também disjuntos. Para cada atributo  $T$  envolvendo tipos de dados fuzzy, existe uma função  $T^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^D \rightarrow [0, 1]$ . Considerando  $a \in I_A$ ,  $v, v_i \in I_D$ ,  $T, T_i \in R_D$ ,  $1 \leq i \leq n$  e  $E \in \mathbf{Dexp}(G)$ , a função de interpretação também é estendida para definir a semântica dos propriedades de tipos de dados e de construtores de conceito adicionais:

$$\begin{aligned} (\exists T_1, \dots, T_n.E)^{\mathcal{I}}(a) &= \sup_{v_1, \dots, v_n \in \Delta_D} (\bigotimes_{i=1}^n T_i^{\mathcal{I}}(a, v_i)) \otimes E^D(v_1, \dots, v_n) \\ (\forall T_1, \dots, T_n.E)^{\mathcal{I}}(a) &= \inf_{v_1, \dots, v_n \in \Delta_D} (\bigotimes_{i=1}^n T_i^{\mathcal{I}}(a, v_i)) \rightarrow E^D(v_1, \dots, v_n) \end{aligned}$$

Segundo a semântica da lógica fuzzy de Zadeh, as operações fuzzy  $x \otimes y = \min(x, y)$  e  $x \rightarrow y = \max(1 - x, y)$ , onde  $x, y \in [0, 1]$  (LUKASIEWICZ; STRACCIA, 2008). Utilizando esses construtores adicionais, é possível definir conceitos a partir das expressões de tipos de dados fuzzy. Por exemplo, a partir do predicado fuzzy personalizado  $p_{item-pequeno}$ , o conceito de *item-pequeno* pode ser definido:

$$item-pequeno = item \sqcap \exists soma, altura, largura, comprimento. p_{item-pequeno} \quad (3.6)$$

Wang et al. (2008) também estenderam a arquitetura para raciocínio definida por Pan (2007) considerando grupos de tipos de dados fuzzy, ilustrada na Figura 3.2. A arquitetura separa o procedimento de raciocínio das fuzzy DLs baseado em *tableau fuzzy*, realizado pelo raciocinador de conceitos fuzzy, do processamento das expressões de tipos de dados fuzzy, realizado pelo raciocinador de tipos de dados fuzzy. Existem dois tipos de raciocinadores de tipos de dados fuzzy: gerenciador de tipos de dados fuzzy e verificador de tipos de dados fuzzy. Assim,

os predicados de tipos de dados *fuzzy* são tratados independentemente da parte abstrata ou simbólica das *fuzzy* DLs, o que confere flexibilidade para adicionar novos tipos de dados e predicados *fuzzy*.

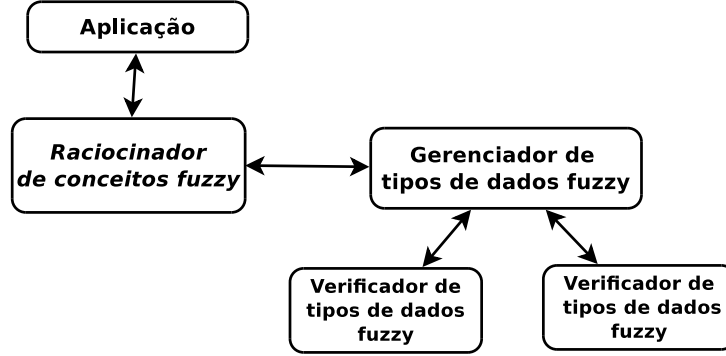


Figura 3.2: Arquitetura para raciocínio proposta por Wang et al. (2008).

O funcionamento geral da arquitetura de raciocínio no caso de grupos de tipos de dados *fuzzy* é bem similar ao caso *crisp*, exceto pelo tratamento de imprecisão presente nos predicados de tipos de dados *fuzzy*. O algoritmo baseado em *tableau fuzzy* invoca o raciocinador de tipos de dados *fuzzy* quando são utilizados construtores de conceito contendo as expressões de tipos de dados *fuzzy*  $E \in \mathbf{Dexp}(G)$  listadas na Tabela 3.8. Os conceitos de *nó de tipo de dado*  $v$ , *sucessor- $T$*  de um nó abstrato  $x$ , *sucessor- $\langle T_1, \dots, T_n \rangle$*  de um nó abstrato  $x$  são similares aos definidos por Pan (2007), sendo que  $v$  é um *sucessor- $T$*  de um nó abstrato  $x$  se  $\langle T \bowtie n \rangle \in \mathcal{L}(\langle x, v \rangle)$ . As expressões de tipos de dados  $E$  que devem ser satisfeitas com relação aos sucessores- $\langle T_1, \dots, T_n \rangle$  de um nó abstrato  $x$  são armazenadas em um conjunto  $DC(x)$  que contém elementos da forma:

$$(\langle E(v_1, \dots, v_n) \bowtie k \rangle) \text{ ou } (\langle v_{i1}, \dots, v_{in} \rangle, \langle v_{j1}, \dots, v_{jn} \rangle, \neq) \quad (3.7)$$

onde  $E \in \mathbf{Dexp}(G)$ ,  $\bowtie \in \{<, \leq, >, \geq\}$ ,  $k \in [0, 1]$  e  $\langle v_1, \dots, v_n \rangle, \langle v_{i1}, \dots, v_{in} \rangle, \langle v_{j1}, \dots, v_{jn} \rangle$  são tuplas de sucessores de tipos de dados *fuzzy* de  $x$ . Também são definidas regras de expansão para os construtores de conceito relacionados a expressões de tipos de dados *fuzzy*, chamadas de regras- $G$  (Tabela 3.9), para expandir o *tableau fuzzy* e armazenar as expressões de tipos de dados *fuzzy*  $E$  em  $DC(x)$ .

Por meio das regras- $G$ , o raciocinador de conceitos *fuzzy* invoca o raciocinador de tipos de dados *fuzzy* como um subprocedimento para verificar a satisfazibilidade de  $DC(x)$ . Assim,  $DC(x)$  atua como uma interface entre o o raciocinador de conceitos *fuzzy* e os raciocinadores de tipos de dados.  $DC(x)$  é satisfatível se a seguinte consulta envolvendo expressões de tipos de dados *fuzzy* é satisfatível:

$$\bigwedge_{\langle E(v_{i1}, \dots, v_{in}) \bowtie k \rangle \in DC(x)} \langle E(v_{i1}, \dots, v_{in}) \bowtie k \rangle \quad (3.8)$$

**Tabela 3.9: Regras-G para tratar expressões de tipos de dados fuzzy (WANG; MA, 2008; WANG et al., 2008).**

Regra- $\exists_{p \triangleright}$	se (1) $\langle \exists T_1, \dots, T_n. E \triangleright k \rangle \in \mathcal{L}(x)$ e (2) $x$ não possui um sucessor- $\langle T_1, \dots, T_n \rangle \langle v_1, \dots, v_n \rangle$ tal que $\langle T_i \triangleright k \rangle \in \mathcal{L}(\langle x, v_i \rangle)$ , $1 \leq i \leq n$ e $\langle E(v_1, \dots, v_n) \triangleright k \rangle \in DC(x)$ então crie um novo sucessor- $\langle T_1, \dots, T_n \rangle \langle v_1, \dots, v_n \rangle$ de $x$ tal que $\mathcal{L}(\langle x, v_i \rangle) = \{ \langle T_i \triangleright k \rangle \}$ e $DC(x) \rightarrow DC(x) \cup \{ \langle E(v_1, \dots, v_n) \triangleright k \rangle \}$
Regra- $\forall_{p \triangleleft}$	se (1) $\langle \forall T_1, \dots, T_n. E \triangleleft k \rangle \in \mathcal{L}(x)$ e (2) $x$ não possui um sucessor- $\langle T_1, \dots, T_n \rangle \langle v_1, \dots, v_n \rangle$ tal que $\langle T_i \triangleleft^{-1} 1 - k \rangle \in \mathcal{L}(\langle x, v_i \rangle)$ , $1 \leq i \leq n$ e $\langle E(v_1, \dots, v_n) \triangleleft k \rangle \in DC(x)$ então crie um novo sucessor- $\langle T_1, \dots, T_n \rangle \langle v_1, \dots, v_n \rangle$ de $x$ tal que $\mathcal{L}(\langle x, v_i \rangle) = \{ \langle T_i \triangleleft^{-1} 1 - k \rangle \}$ e $DC(x) \rightarrow DC(x) \cup \{ \langle E(v_1, \dots, v_n) \triangleleft k \rangle \}$
Regra- $\exists_{p \triangleleft}$	se (1) $\langle \exists T_1, \dots, T_n. E \triangleleft k \rangle \in \mathcal{L}(x)$ e (2) $x$ possui um sucessor- $\langle T_1, \dots, T_n \rangle \langle v_1, \dots, v_n \rangle$ tal que $\psi_i^C \in \mathcal{L}(\langle x, v_i \rangle)$ , com $\psi_i = \langle T_i \triangleleft k \rangle$ , $1 \leq i \leq n$ e $\langle E(v_1, \dots, v_n) \triangleleft k \rangle \notin DC(x)$ então $DC(x) \rightarrow DC(x) \cup \{ \langle E(v_1, \dots, v_n) \triangleleft k \rangle \}$
Regra- $\forall_{p \triangleright}$	se (1) $\langle \forall T_1, \dots, T_n. E \triangleright k \rangle \in \mathcal{L}(x)$ e (2) $x$ possui um sucessor- $\langle T_1, \dots, T_n \rangle \langle v_1, \dots, v_n \rangle$ tal que $\psi_i^C \in \mathcal{L}(\langle x, v_i \rangle)$ , com $\psi_i = \langle T_i \triangleright^{-1} 1 - k \rangle$ , $1 \leq i \leq n$ e $\langle E(v_1, \dots, v_n) \triangleright k \rangle \notin DC(x)$ então $DC(x) \rightarrow DC(x) \cup \{ \langle E(v_1, \dots, v_n) \triangleright k \rangle \}$

Além das condições de *clash* do algoritmo baseado em *tableau fuzzy*, uma nova condição é adicionada: um *tableau fuzzy* contém um *clash* se, para algum nó abstrato  $x$ ,  $DC(x)$  é insatisfatível.

O gerenciador de tipos de dados *fuzzy* trabalha entre o raciocinador de conceitos *fuzzy* e um conjunto de verificadores de tipos de dados *fuzzy*. De forma geral, o gerenciador de tipos de dados *fuzzy* decompõe uma conjunção de expressões de tipos de dados *fuzzy* em  $DC(x)$  (formato ilustrado na Equação 3.8) em disjunções de conjunções de predicados de tipos de dados *fuzzy*, de acordo com a semântica das expressões descrita na Tabela 3.8. Então, cada conjunção é subdividida em sub-conjunções, onde cada sub-conjunção contém predicados *fuzzy* atômicos de um determinado tipo de dado base (por exemplo, *string*, *real*). Essas sub-conjunções são enviadas aos verificadores de tipos de dados *fuzzy* correspondentes, responsáveis por verificar a satisfazibilidade delas. Se todas as sub-conjunções forem satisfatíveis, então a conjunção de predicados de tipos de dados *fuzzy* é satisfatível. Se pelo menos uma conjunção pertencente à disjunção for satisfatível, então  $DC(x)$  é satisfatível. O gerenciador de tipos de dados *fuzzy* também realiza algumas verificações de satisfazibilidade que independem da avaliação dos predicados. Por exemplo, verifica se  $v_1, \dots, v_n$  são argumentos de ambos  $p$  e  $\bar{p}$  ou se dois predicados *fuzzy*, definidos por diferentes tipos de dados base, compartilham variáveis.

Os verificadores de tipos de dados *fuzzy*, por sua vez, verificam a satisfazibilidade de conjunções de predicados de tipos de dados *fuzzy* definidos sobre um tipo de dado base  $d \in D_G$ . Como os tipos de dados base são disjuntos por definição, os verificadores podem trabalhar de

forma independente, sendo que os resultados de suas verificações são reunidos pelo gerenciador de tipos de dados *fuzzy*, que então retorna o resultado final para o raciocinador de conceitos. Assim, um verificador de tipo de dado *fuzzy* recebe como entrada uma conjunção finita de predicados *fuzzy*  $\mathcal{C}$  da forma (WANG et al., 2008):

$$\mathcal{C} = \bigwedge_{j=1}^m \langle p_j(v_1^j, \dots, v_{n_j}^j) \bowtie k \rangle \quad (3.9)$$

onde  $v_i^j$  são variáveis e  $p_j$  é um predicado de tipo de dado *fuzzy* em  $G$  definido sobre  $d$  com aridade  $n_j$ . O verificador para o tipo de dado  $d$  verifica se  $\mathcal{C}$  é satisfatível ou não, de acordo com a semântica dos predicados definidos sobre  $d$ .  $\mathcal{C}$  é insatisfatível quando ocorre um par conjugado considerando um mesmo predicado *fuzzy*  $p$  sobre o mesmo conjunto de variáveis  $v_1, \dots, v_n$ . Por exemplo,  $\langle p(v_1, \dots, v_n) \triangleright r \rangle \in \mathcal{C}$  e  $\langle p(v_1, \dots, v_n) \triangleright^{-1} r' \rangle \in \mathcal{C}$  com  $r > r'$ . Outra situação em que  $\mathcal{C}$  é insatisfatível ocorre quando uma de suas restrições *fuzzy* é insatisfatível, ou seja, quando  $\langle p(v_1, \dots, v_n) \bowtie r \rangle \in \mathcal{C}$  mas a verificação do grau de verdade de  $p(v_1, \dots, v_n)$  não satisfaz a restrição  $\bowtie r$ . Por exemplo,  $\langle p(v_1, \dots, v_n) < 0.7 \rangle \in \mathcal{C}$  e a avaliação do predicado  $p$  com os valores de específicos de  $v_1, \dots, v_n$  resulta em  $p(v_1, \dots, v_n) = 0.9$ .

Os tipos de dados e predicados *fuzzy*, assim como a arquitetura de raciocínio proposta por Wang et al. (2008), que separa o raciocínio de *fuzzy* DLs e de grupos de tipos de dados *fuzzy* de forma flexível, inspiraram o desenvolvimento da integração de raciocínio de ontologias *fuzzy* e sistemas de inferência *fuzzy* definida nesta tese de doutorado.

## 3.6 Considerações Finais

Neste capítulo, foram apresentados os principais conceitos relacionados à representação e ao raciocínio de ontologias *fuzzy*. Primeiramente, foram descritos os conceitos básicos de ontologias convencionais e posteriormente a extensão para tratamento de imprecisão, por meio da incorporação de conceitos da teoria de conjuntos *fuzzy* e lógica *fuzzy*. Esses conceitos são fundamentais para as abordagens definidas nesta tese de doutorado, que tratam a integração de raciocínio de ontologias *fuzzy* e sistemas de inferência *fuzzy*. O próximo capítulo descreve trabalhos correlatos que visam combinar ontologias *fuzzy* e regras *fuzzy*, discutindo suas contribuições e limitações mais relevantes.

# Capítulo 4

## TRABALHOS CORRELATOS COMBINANDO ONTOLOGIAS *Fuzzy* E REGRAS *Fuzzy*

---

---

### 4.1 Considerações Iniciais

Conforme apresentado no Capítulo 3, ontologias *fuzzy* permitem representar e processar conhecimento impreciso por meio de construtores expressivos e mecanismos de raciocínio lógico. Dentre as extensões *fuzzy* de ontologias na literatura, os recursos de modelagem correspondentes a variáveis e termos linguísticos são relevantes por expressarem a imprecisão inerente a algumas situações reais assim como ao vocabulário e ao raciocínio humanos.

Combinar o formalismo das ontologias *fuzzy*, que estrutura o conhecimento por meio de conceitos, construtores de conceitos e axiomas, com a representação e o raciocínio baseado em regras *fuzzy* pode resultar em uma maior expressividade com relação às inferências envolvendo informações imprecisas. Em especial, ao combinar ontologias *fuzzy* com regras contendo variáveis e termos linguísticos, é possível aplicar métodos de raciocínio aproximado de SIF para realizar inferências associadas a propriedades numéricas vinculadas a termos imprecisos, que ocorrem com frequência na linguagem humana.

Neste sentido, diversas pesquisas têm investigado abordagens para integração de ontologias *fuzzy* e regras *fuzzy* envolvendo variáveis linguísticas. Ao analisar as abordagens propostas na literatura, existem algumas questões a serem consideradas:

1. A semântica das inferências baseadas em regras *fuzzy* corresponde às necessidades do domínio ou da aplicação? É importante disponibilizar construtores, operadores e métodos de raciocínio apropriados para que as inferências baseadas em regras *fuzzy* sejam significativas e atendam aos requisitos da aplicação;

2. As inferências baseadas em regras *fuzzy* contribuem para a realização de outras tarefas de raciocínio da ontologia *fuzzy*? Algumas tarefas de raciocínio da ontologia *fuzzy* podem depender das inferências providas pelas regras *fuzzy* como, por exemplo, a verificação da pertinência de uma instância para um conceito definido com base em valores de propriedades inferidos por meio de regras *fuzzy*;
3. O conjunto de inferências *fuzzy* disponibilizadas compreende tanto o raciocínio de ontologias *fuzzy* quanto o raciocínio baseado em regras *fuzzy*? Algumas aplicações demandam inferências *fuzzy* baseadas tanto nas definições da ontologia *fuzzy* quanto no processamento de regras *fuzzy*, que podem ser complementares e significativas para o domínio. Por exemplo, os valores das propriedades correspondentes aos atributos presentes em regras *fuzzy* podem ser obtidos por meio do raciocínio de ontologias *fuzzy*.

Assim, o objetivo deste capítulo é descrever abordagens da literatura que tratam a integração de ontologias *fuzzy* e regras *fuzzy*, com foco no processamento de variáveis linguísticas associadas a propriedades numéricas modeladas na ontologia *fuzzy*. A Seção 4.2 descreve trabalhos que abordam ontologias *fuzzy* e regras *fuzzy* envolvendo propriedades numéricas e termos linguísticos, mas que não consideram sistemas de inferência *fuzzy*. Na Seção 4.3, são apresentadas abordagens que combinam ontologias e, especificamente, sistemas de inferência *fuzzy* para o processamento de regras *fuzzy*. Por fim, a Seção 4.4 discute de forma resumida os trabalhos correlatos descritos no decorrer deste capítulo, sob a ótica das questões mencionadas.

## 4.2 Ontologias *Fuzzy* e Regras *Fuzzy* Envolvendo Termos Linguísticos

Segundo Lukasiewicz e Straccia (2008), uma das tendências de pesquisa em ontologias *fuzzy* é a integração com linguagens baseadas em regras. Existem diversos trabalhos que tratam regras *fuzzy* cujos antecedentes e consequentes contêm conceitos e relacionamentos *fuzzy* definidos na ontologia, levando em consideração graus de verdade das asserções *fuzzy* e até pesos que permitem especificar quais condições e regras são mais relevantes. Neste contexto, estão trabalhos como f-SWRL (PAN et al., 2006), *fuzzy* CARIN (MAILIS; STOILLOS; STAMOU, 2007), Vague-SWRL (WANG et al., 2008) e *fuzzy* RuleML (DAMÁSIO et al., 2008). Entretanto, as pesquisas que consideram esse tipo de regra *fuzzy* não abordam especificamente regras *fuzzy* que contêm variáveis linguísticas e, portanto, não tratam o raciocínio aproximado envolvendo propriedades numéricas e termos linguísticos.

Diversas pesquisas voltadas para ontologias *fuzzy* também oferecem suporte a regras *fuzzy* que combinam propriedades correspondentes a variáveis linguísticas. Agarwal e Hitzler (2005) propõem a representação de regras *fuzzy* por meio de construtores da DL  $\mathcal{ALC}(D)$  com suporte a tipos de dados concretos e operações de agregação (soma, média, máximo, entre outros). Como os próprios construtores da DL  $\mathcal{ALC}(D)$  são utilizados para modelar variáveis linguísticas e regras *fuzzy*, a sintaxe dessa DL não é modificada, o que permite o uso de linguagens de ontologias e raciocinadores *crisp* para realizar a inferência baseada em regras *fuzzy*, desde que tratem  $\mathcal{ALC}(D)$  e operações de agregação. Com relação ao raciocínio aproximado, Agarwal e Hitzler (2005) propõem a aplicação da regra da inferência composicional (Seção 2.5.1) para cada regra e posteriormente a agregação dos resultados inferidos, com as operações *max* para agregação, *min* para a conjunção dos antecedentes e semântica da regra, além do método de centro de área para defuzificação. No entanto, o foco das inferências *fuzzy* está voltado somente para as regras *fuzzy*, pois o formalismo para representação da ontologia é baseado em uma lógica de descrição *crisp* ( $\mathcal{ALC}(D)$ ).

Um dos motores de inferência mais completos para ontologias *fuzzy* é fuzzyDL (BOBILLO; STRACCIA, 2008), que baseia-se na *fuzzy* DL  $\mathcal{SHIF}(D)$ . O raciocinador fuzzyDL oferece suporte a propriedades sobre tipos de dados como números reais, inteiros e cadeias de caracteres, além de permitir a definição de conceitos *fuzzy* a partir de funções de pertinência. Assim, é possível representar termos linguísticos definidos por funções de pertinência parametrizadas, como funções triangulares e trapezoidais, com base na teoria de tipos de dados *fuzzy*. Com relação às inferências, *fuzzyDL* combina um algoritmo baseado em *tableau* estendido para *fuzzy*  $\mathcal{SHIF}(D)$  e otimização baseada em MILP (*Mixed Integer Linear Programming*), que resolvem as tarefas de verificação de instâncias, subsunção de conceitos, verificação de satisfazibilidade e verificação de melhor grau de verdade (Seção 3.5.2 do Capítulo 3). Com relação à combinação de ontologias *fuzzy* e regras *fuzzy* envolvendo variáveis linguísticas, fuzzyDL permite representar regras *fuzzy* e realizar inferências segundo diferentes abordagens:

- Implicações *fuzzy*: regras *fuzzy* podem ser representadas por meio da operação de implicação *fuzzy*, correspondendo a regras implicativas, cuja semântica depende das operações de conjunção e implicação *fuzzy* consideradas. Em fuzzyDL, é possível escolher qual o tipo de semântica das operações *fuzzy*, como a semântica de Zadeh e de Lukasiewicz (LUKASIEWICZ; STRACCIA, 2008). No caso da semântica de Zadeh, mais comum entre as aplicações, utiliza-se a operação de mínimo para conjunções e a implicação de Kleene-Dienes ou a inclusão de conjuntos de Zadeh para as implicações *fuzzy*. Conforme descrito por Guillaume e Charnomordic (2012), as regras implicativas são combinadas por conjunção, em que diversas regras levam a um conjunto mais restrito de situações

possíveis do que uma única regra;

- Conceitos *fuzzy* e operações *fuzzy*: também é possível representar regras *fuzzy* similares às regras de SIF segundo o método de Mamdani por meio de construtores de conceitos e operações *fuzzy*. Para inferir valores numéricos a partir das regras, fuzzyDL disponibiliza três tipos de consulta correspondentes aos métodos de defuzzificação SOM, MOM e LOM (Seção 2.5.3). Assim, os próprios construtores de conceito da ontologia *fuzzy* podem ser usados para modelar regras *fuzzy* e, juntamente com as consultas de defuzzificação, reproduzir um sistema de inferência *fuzzy*. Vale ressaltar que fuzzyDL não utiliza um SIF específico, mas é possível simular um SIF ao considerar os construtores de conceito *fuzzy* e consultas disponíveis. Também é possível obter o valor inferido a partir de regras *fuzzy* por meio de consultas do tipo *show-concrete-fillers-for* combinadas a consultas por verificação de satisfazibilidade ou de melhor grau de verdade.

Por ser baseado em uma lógica de descrição *fuzzy*, fuzzyDL obtém inferências *fuzzy* envolvendo tanto a ontologia *fuzzy* quanto regras *fuzzy* modeladas segundo as abordagens mencionadas.

Reformat e Golmohammadi (2009) definem um *framework* para sistemas de computação com palavras baseados em ontologias, que possibilita a representação de conceitos relacionados às lógicas possibilística, probabilística e *fuzzy*. Com relação à representação de imprecisão, os autores definem uma ontologia que modela propriedades contendo termos linguísticos, associando-os a funções de pertinência parametrizadas, como funções trapezoidais, função S, entre outras. Quanto ao processamento de informações imprecisas, o *framework* proposto considera regras contendo termos linguísticos, porém as inferências realizadas não levam em consideração métodos de raciocínio aproximado baseados em regras *fuzzy*, como os citados na Seção 2.5 do Capítulo 2.

Bragaglia et al. (2010) propõem um motor de inferência híbrido que combina motores de inferência existentes para processar regras, ontologias tradicionais e ontologias *fuzzy*. Respetivamente, os motores de inferência utilizados são Drools, Pellet e fuzzyDL. Neste trabalho, embora as regras *fuzzy* contenham termos linguísticos, elas combinam outros elementos de ontologias, como conceitos e relacionamentos abstratos. Assim, as regras *fuzzy* não tratam especificamente o raciocínio envolvendo variáveis linguísticas para obter valores de propriedades numéricas.

DeLorean (BOBILLO; DELGADO; GÓMEZ-ROMERO, 2012) é um motor de inferência que oferece suporte à linguagem fuzzy OWL 2, correspondente à *fuzzy DL SROIQ(D)*. Mais precisamente, DeLorean é um tradutor de uma linguagem de ontologias *fuzzy* para uma



linguagem de ontologias convencionais, sendo que o raciocínio é realizado por um motor de inferência de ontologias *crisp*. A grande contribuição de DeLorean está, portanto, na abordagem de tradução entre essas linguagens, que alia a expressividade de *fuzzy* DLs à interoperabilidade de motores de inferência de ontologias convencionais. Embora até seja possível expressar regras *fuzzy* envolvendo termos linguísticos usando alguns construtores de *fuzzy*  $SR\mathcal{OIQ}(D)$ , DeLorean não disponibiliza uma interface de consulta sobre os valores inferidos das propriedades numéricas, tampouco métodos de defuzificação.

Enfim, de modo geral, as abordagens que integram ontologias e regras *fuzzy* visam aumentar a expressividade com relação à representação e ao processamento de imprecisão, tão comum em diversos domínios. Dentre os trabalhos analisados, devem-se destacar as pesquisas que tratam regras contendo termos linguísticos, pois permitem representar um conhecimento mais próximo da linguagem humana, facilitando tanto a modelagem quanto a interpretação do conhecimento por especialistas. Em especial, *fuzzyDL* é capaz de realizar inferências *fuzzy* não somente referentes ao conhecimento modelado na ontologia *fuzzy* mas também permite processar regras *fuzzy* envolvendo variáveis *fuzzy* para inferir propriedades numéricas.

### 4.3 Ontologias e Sistemas de Inferência Fuzzy

Uma outra linha de pesquisa considera a combinação de ontologias e sistemas de inferência *fuzzy* para realizar inferências envolvendo propriedades numéricas e termos linguísticos. Neste contexto, a ontologia é responsável por representar e realizar inferências associadas aos conceitos, axiomas e relacionamentos do domínio, enquanto o SIF trata especificamente o raciocínio baseado em regras *fuzzy* relacionado a propriedades numéricas.

Existem trabalhos que consideram uma ontologia tradicional associada a regras *fuzzy*. Neste caso, o raciocínio envolvendo imprecisão é limitado às regras *fuzzy*, uma vez que a ontologia não representa conceitos nem relacionamentos *fuzzy*. Bobillo et al. (2009b) combinam ontologias *crisp* e regras *fuzzy* segundo o modelo de Mamdani em um sistema de suporte a decisão no domínio de gerenciamento de negócios. A ontologia modela as principais categorias do domínio como perspectivas (Negócio, Cliente, Ambiente, entre outras), assim como variáveis linguísticas, com base em uma meta-ontologia proposta por Blanco et al. (2005). O raciocínio de ontologias é utilizado principalmente para verificação de consistência com relação a restrições de cardinalidade de relacionamentos e de disjunção de classes, não envolvendo conhecimento impreciso. Os valores a serem enviados como entradas para o SIF são representados explicitamente na ontologia, e os valores inferidos pelo SIF são apresentados diretamente à aplicação,

não sendo utilizados em outros conceitos da ontologia.

Outro trabalho que considera ontologias e regras *fuzzy* é SWRL-F (WLODARCZYK et al., 2010). Em SWRL-F, a ontologia modela conceitos e relacionamentos que representam conjuntos *fuzzy*, variáveis e termos linguísticos utilizando os construtores da própria linguagem OWL DL. Para editar as regras e realizar inferências, Wlodarczyk et al. (2010) modificam o *plug-in* SWRLJessTab da ferramenta Protégé, com base no motor de inferência FuzzyJess e no FuzzyJ Toolkit (ORCHARD, 2001). As inferências *fuzzy* em SWRL-F são inspiradas nos controladores *fuzzy*, envolvendo as etapas de fuzificação das entradas, aplicação do método de raciocínio aproximado e defuzificação dos resultados. Entretanto, as inferências *fuzzy* em SWRL-F limitam-se às regras *fuzzy*, portanto as tarefas típicas de raciocínio para ontologias *fuzzy* não são consideradas.

Diversos trabalhos têm utilizado a *Fuzzy Markup Language* (FML) (ACAMPORA; LOIA, 2005) em conjunto com ontologias. FML permite especificar informações associadas a SIF: conjuntos de regras *fuzzy*, variáveis linguísticas, métodos de raciocínio e de defuzificação. Segundo Lee et al. (2010a), embora FML tenha sido projetada para o contexto de sistemas de controle *fuzzy*, ela tem sido combinada com ontologias *fuzzy* em diversos domínios de aplicação. De forma geral, a ontologia tem o papel de representar os principais conceitos do domínio, assim como as propriedades numéricas correspondentes a variáveis linguísticas, enquanto FML descreve os dados mais específicos do raciocínio de SIF, de modo interoperável e independente de implementação. Huang et al. (2011) propõem um sistema de apoio a decisão baseado em ontologias e FML para predição de comportamento de programas maliciosos, utilizando regras *fuzzy* para combinar as variáveis deste domínio, como o número de conexões IP abertas e o número de modificações no sistema de arquivos para determinar o risco associado aos programas. Wang et al. (2013) também desenvolveram um sistema de apoio a decisão usando ontologias e FML no domínio energético, com foco no fornecimento e na demanda de energia solar com base em características ambientais e de consumo. No entanto, estes trabalhos consideram ontologias *crisp*, portanto a representação e o raciocínio de conhecimento impreciso está limitado às regras *fuzzy* processadas por um SIF.

Ainda usando FML porém combinada com ontologias *fuzzy*, Lee et al. (2009) propõem um agente inteligente voltado para domínio médico, envolvendo a recomendação de dietas mais apropriadas para o tratamento de diabetes. Neste mesmo contexto, mas com foco em alimentos típicos da culinária japonesa, Kurozumi et al. (2013) utilizam uma ontologia *fuzzy* para representar categorias e propriedades relacionadas aos alimentos e regras *fuzzy* para avaliar se as dietas são saudáveis de acordo com recomendações de associações médicas japonesas.

Algumas pesquisas têm utilizado FML e ontologias *fuzzy* com suporte a conjuntos *fuzzy* tipo-2 para tratar a incerteza associada à definição dos termos linguísticos. Seguindo essa abordagem, Lee et al. (2010a) também realizam recomendações de dietas saudáveis e Lee et al. (2010b) propõem um agente inteligente para o domínio de jogos, que avalia as melhores jogadas com base no raciocínio aproximado de regras *fuzzy*. Embora considerem ontologias *fuzzy*, o foco dos trabalhos mencionados está no raciocínio *fuzzy* baseado em regras *fuzzy*, em função dos resultados do SIF. Nestes trabalhos, a ontologia *fuzzy* não é fundamentada por algum formalismo lógico que possibilite a inferência de conhecimento impreciso como ocorre nas lógicas de descrição *fuzzy*, portanto os trabalhos não incluem as inferências providas por motores de inferência de ontologias *fuzzy*.

OWL-FC (MAIO et al., 2012) também representa variáveis linguísticas, regras *fuzzy* e especifica métodos de raciocínio de SIF por meio de uma ontologia de alto nível no contexto de controle *fuzzy*. Os autores argumentam que essa abordagem é mais apropriada que FML por considerar um meta-modelo que pode ser instanciado diretamente por linguagens de representação de ontologias. Assim, os conceitos do domínio e suas propriedades são associadas às informações do SIF por meio de relacionamentos modelados na própria ontologia, de acordo com o meta-modelo definido. OWL-FC é baseada em OWL, seguindo padrões da *Web Semântica* (BERNERS-LEE; HENDLER; LASSILA, 2001), com foco na interoperabilidade entre diferentes aplicações que lidam com ontologias e controle *fuzzy*. Entretanto, OWL-FC considera ontologias *crisp* para representar a semântica conceitual do domínio, também explorando somente as inferências baseadas em regras *fuzzy* com relação ao raciocínio envolvendo imprecisão.

Em geral, os trabalhos analisados que combinam ontologias e SIF possuem diversas características em comum. A ontologia é utilizada para estruturar o conhecimento com ênfase na interoperabilidade de informações semânticas entre aplicações do domínio, enquanto o processamento de regras *fuzzy*, realizado por SIF, é responsável por prover as principais inferências envolvendo imprecisão. Mesmo em abordagens que consideram ontologias *fuzzy*, o potencial do raciocínio baseado em construtores de conceitos e axiomas *fuzzy* de ontologias não é devidamente explorado, desprezando inferências que podem ser relevantes para alguns domínios de aplicação.

Uma vez apresentadas abordagens da literatura que tratam ontologias *fuzzy* e regras *fuzzy* envolvendo variáveis linguísticas, a próxima seção analisa suas características com relação aos aspectos discutidos no início deste capítulo.

## 4.4 Discussão sobre Abordagens Combinando Ontologias e Regras *Fuzzy*

Ao analisar as questões apresentadas no início deste capítulo, é possível verificar que os diversos trabalhos da literatura possuem algumas contribuições e limitações em comum.

Quanto à semântica do raciocínio baseado em regras *fuzzy* (questão 1), existem abordagens de ontologias *fuzzy* que tratam regras *fuzzy* combinando diversos elementos como conceitos e relacionamentos abstratos assim como propriedades concretas. Dependendo da semântica que se pretende representar com as regras *fuzzy*, esse tipo de regra é bastante apropriado, principalmente para tratar situações como a inferência de relacionamentos *fuzzy* abstratos a partir da combinação de outros relacionamentos, que não é possível em algumas lógicas de descrição *fuzzy* por serem fortemente baseadas em raciocínio a partir de construtores e axiomas envolvendo conceitos. Nestes casos, as regras *fuzzy* agregam mais expressividade por permitir esse tipo de inferência, como em alguns trabalhos citados na Seção 4.2: (PAN et al., 2006; MAILIS; STOILOS; STAMOU, 2007; WANG et al., 2008; DAMÁSIO et al., 2008; REFORMAT; GOLMOHAMMADI, 2009; BRAGAGLIA et al., 2010). No entanto, quando as aplicações demandam especificamente raciocínio baseado em regras *fuzzy* que combinam variáveis linguísticas para obter valores de propriedades numéricas, como no caso dos métodos de raciocínio de SIF, esses trabalhos são menos apropriados porque não abordam esse tipo de raciocínio. Como o foco da tese está voltado para regras *fuzzy* envolvendo variáveis linguísticas e inferências associadas a propriedades numéricas, o tipo de semântica considerado é diferente desses trabalhos.

Tanto em fuzzyDL (BOBILLO; STRACCIA, 2008) quanto em DeLorean (BOBILLO; DELGADO; GÓMEZ-ROMERO, 2012), os construtores de conceito e operações *fuzzy* possibilitam a representação de regras implicativas e regras *fuzzy* segundo a semântica de Mamdani. Embora as regras implicativas possuam uma semântica distinta dos métodos de raciocínio de SIF, são capazes de inferir valores de propriedades numéricas com base em termos linguísticos. No entanto, DeLorean não disponibiliza consultas aos valores inferidos de propriedades concretas numéricas, nem métodos de defuzificação que permitam gerar um valor numérico em função do raciocínio baseado em regras *fuzzy*.

Em fuzzyDL, os resultados inferidos a partir das regras de Mamdani podem ser defuzificados por três diferentes métodos (SOM, MOM e LOM). Existe ainda a opção de obter valores numéricos com a consulta *show-concrete-fillers* associada a tarefas de raciocínio da ontologia *fuzzy* envolvendo os conceitos que representam as regras *fuzzy*. Entretanto, as abordagens

disponibilizadas por fuzzyDL para tratar regras *fuzzy* podem ser incompatíveis dependendo da semântica do domínio, em função dos métodos de raciocínio restritos a regras implicativas, semântica de Mamdani e métodos de defuzzificação baseados em intervalos de grau máximo. Com relação à integração das inferências, como as regras *fuzzy* são representadas por meio dos próprios construtores de conceito, axiomas e operadores *fuzzy* de fuzzyDL, as inferências das regras estão integradas ao conhecimento modelado pela ontologia *fuzzy* (questão 2).

Considerando a questão 3, como fuzzyDL e DeLorean são fundamentados por *fuzzy* DLs, são capazes de raciocinar a partir das definições de conceito e asserções *fuzzy* presentes na ontologia *fuzzy* e, portanto, o conjunto de inferências *fuzzy* compreende tanto as inferências da ontologia *fuzzy* quanto das regras *fuzzy* modeladas.

Dentre os trabalhos correlatos, também foram identificadas pesquisas que processam regras *fuzzy* segundo métodos de raciocínio de SIF, porém consideram ontologias *crisp* (AGARWAL; HITZLER, 2005; BOBILLO et al., 2009b; WLODARCZYK et al., 2010; HUANG et al., 2011; WANG et al., 2013; MAIO et al., 2012). Nesses trabalhos, as inferências baseadas em conhecimento impreciso são limitadas ao raciocínio baseado em regras *fuzzy*, pois não incluem as inferências associadas à imprecisão representada em ontologias. Essa limitação está relacionada à questão 3, que enfatiza que as inferências relacionadas às ontologias *fuzzy*, combinadas às regras *fuzzy*, podem ser relevantes para alguns domínios de aplicação.

Embora alguns trabalhos baseados em FML (LEE et al., 2009, 2010a, 2010b; KUROZUMI et al., 2013) argumentem que consideram ontologias *fuzzy* combinadas com SIF, o foco das inferências envolvendo imprecisão está voltado para as regras *fuzzy*. Nesses trabalhos, como as ontologias *fuzzy* não seguem um formalismo que viabilize inferências baseadas em conceitos e asserções *fuzzy*, a questão 3 não é tratada por completo. Neste contexto, a ontologia *fuzzy* é usada para estruturar o conhecimento do domínio, geralmente voltada para interoperabilidade de informação semântica, porém o raciocínio não é devidamente explorado como em abordagens formais baseadas em lógicas de descrição *fuzzy*. Conforme mencionado no início deste capítulo, as inferências envolvendo elementos de ontologias *fuzzy* são também importantes para expressar a imprecisão presente em alguns domínios de aplicação, inclusive para a definição de valores a serem considerados no raciocínio baseado em regras *fuzzy*.

As questões analisadas com relação aos principais trabalhos correlatos são sumarizadas na Tabela 4.1. Note que a tabela somente lista as abordagens que tratam regras *fuzzy* contendo termos linguísticos para inferência de propriedades numéricas. Dentre os trabalhos considerados, existem diferentes tipos de métodos de raciocínio para regras *fuzzy*, desde regras implicativas até Mamdani e Larsen, combinados a diversos tipos de defuzzificação. Vale ressaltar que a coluna

*Arquitetura de raciocínio flexível* refere-se especificamente à flexibilidade de instanciação de métodos de raciocínio baseado em regras *fuzzy* que sejam mais apropriados para as aplicações.

Especificamente, no caso das abordagens baseadas em FML e OWL-FC, a semântica das regras *fuzzy* a ser considerada é aquela que mais se adequa às necessidades da aplicação, pois esses trabalhos abordam arquiteturas de raciocínio flexíveis que permitem especificar as informações relativas às regras *fuzzy* de forma independente de implementação de SIF. Portanto, caso seja apropriada a semântica de Mamdani, de Larsen ou de algum outro método de raciocínio, basta especificá-la apropriadamente usando FML ou OWL-FC e acoplar a implementação de SIF correspondente com o suporte de um programa tradutor entre o formato genérico e a linguagem específica do SIF.

Apesar de todos os trabalhos listados na Tabela 4.1 considerarem o resultado das inferências das regras *fuzzy* na ontologia, apenas fuzzyDL e DeLorean abordam o raciocínio baseado em ontologias *fuzzy*. Por outro lado, os métodos de raciocínio baseado em regras *fuzzy* específicos disponíveis nesses motores de inferência podem ser menos apropriados para os requisitos do domínio, dependendo da aplicação.

**Tabela 4.1: Características analisadas nos principais trabalhos correlatos.**

Trabalhos correlatos	Raciocínio baseado em regras <i>fuzzy</i>	Defuzificação	Inferências de regras <i>fuzzy</i> integradas à ontologia	Raciocínio de ontologias <i>fuzzy</i> formais	Arquitetura de raciocínio flexível
(AGARWAL; HITZLER, 2005)	Mamdani	COA	sim	não	não
fuzzyDL (BOBILLO; STRACCIA, 2008)	regras implicativas e Mamdani	SOM, MOM e LOM	sim	sim	não
(BOBILLO et al., 2009b)	Mamdani	COA	sim	não	não
SWRL-F (WLODARCZYK et al., 2010)	Mamdani, Larsen	MOM, COA e Moment	sim	não	não
Abordagens baseadas em FML (LEE et al., 2009, 2010a, 2010b; HUANG et al., 2011; WANG et al., 2013; KUROZUMI et al., 2013)	dependente da aplicação	dependente da aplicação	sim	não	sim
OWL-FC (MAIO et al., 2012)	dependente da aplicação	dependente da aplicação	sim	não	sim
DeLorean (BOBILLO; DELGADO; GÓMEZ-ROMERO, 2012)	regras implicativas e Mamdani	não	sim	sim	não

Desta forma, considerando as características discutidas, é necessário combinar de forma efetiva o raciocínio de ontologias *fuzzy* com inferências baseadas em regras *fuzzy*, proporcionando uma semântica que atenda às necessidades das aplicações. É importante considerar o raciocínio de ontologias *fuzzy*, que provê inferências relevantes com relação aos construtores de conceitos e asserções *fuzzy* do domínio. Com relação às regras *fuzzy*, é interessante investigar arquiteturas de raciocínio flexíveis, como nas abordagens que utilizam FML e OWL-FC, para possibilitar a integração com implementações de SIF que disponibilizem os métodos de raciocínio e de defuzificação mais apropriados à semântica do domínio.

## 4.5 Considerações Finais

Neste capítulo, foram descritos os principais trabalhos correlatos que investigam a integração de ontologias *fuzzy* e regras *fuzzy*. Especificamente, as regras *fuzzy* envolvem o processamento de variáveis linguísticas, assim como é realizado em sistemas de inferência *fuzzy*, associadas a propriedades numéricas modeladas na ontologia *fuzzy*. A Seção 4.4 apresenta uma discussão sobre a revisão da literatura, identificando limitações nas abordagens existentes com relação aos aspectos de integração apresentados no início do capítulo. Com base nessa análise, foram definidas e desenvolvidas duas abordagens para integração de raciocínio de ontologias *fuzzy* e de sistemas de inferência *fuzzy*, descritas com detalhes no próximo capítulo.

## **Parte II**

# **Trabalhos Desenvolvidos**



# Capítulo 5

## ABORDAGENS PARA INTEGRAÇÃO DE RACIOCÍNIO DE ONTOLOGIAS FUZZY E SISTEMAS DE INFERÊNCIA FUZZY

---

---

### 5.1 Considerações Iniciais

O principal objetivo de integrar o raciocínio de ontologias *fuzzy* e de sistemas de inferência *fuzzy* é prover inferências associadas a propriedades numéricas, derivadas do processamento de regras *fuzzy*. Para tanto, as propriedades numéricas envolvidas nesse processo de inferência devem estar associadas a termos linguísticos representados por conjuntos *fuzzy*.

Neste sentido, quando aplicações baseadas em ontologias processarem consultas que envolvem conhecimento relacionado a propriedades numéricas, um sistema de inferência *fuzzy* pode ser invocado para inferir os valores numéricos necessários, a partir de valores de entrada obtidos da ontologia *fuzzy*. As saídas inferidas são então retornadas à ontologia *fuzzy*, contribuindo para a realização de suas tarefas de raciocínio. Em especial, neste trabalho analisou-se como a tarefa de verificação de instâncias envolvendo restrições sobre valores de propriedades numéricas beneficia-se dos resultados providos por SIF.

Estudos preliminares com foco na representação de ontologias *fuzzy* resultaram na definição de uma meta-ontologia para modelar conceitos, relacionamentos e propriedades *fuzzy* associada a sistemas de classificação baseados em regras *fuzzy* (YAGUINUMA et al., 2010a, 2010b, 2012, 2014). Embora as questões específicas à integração de inferências não tenham sido abordadas, os estudos evidenciaram a necessidade por investigar a combinação de ontologias *fuzzy* e sistemas de inferência *fuzzy*. Assim, a partir dos requisitos identificados, foram definidas e desenvolvidas duas abordagens para integração de raciocínio de ontologias *fuzzy* e de sistemas de inferência *fuzzy* no contexto desta tese de doutorado.

A primeira abordagem desenvolvida resultou no motor de inferência *HyFOM*, cujo nome foi inspirado em *Hybrid integration of Fuzzy Ontology and Mamdani reasoning*. Sua primeira versão oferecia suporte a sistemas de inferência de Mamdani, mas posteriormente foi estendido para oferecer suporte a SIF em geral. *HyFOM* possui uma arquitetura híbrida por combinar motores de inferência existentes na literatura de forma não-intrusiva, com foco na interface e na interação entre os mesmos. Assim, implementações existentes de racionadores de ontologias e de sistemas de inferência *fuzzy* são combinadas para oferecer inferências associadas a propriedades numéricas.

A segunda abordagem desenvolvida foi *FT-FIS*, referente a *Fuzzy Tableau and Fuzzy Inference System*. Esta abordagem, mais fortemente acoplada que *HyFOM*, integra um algoritmo baseado em *tableau fuzzy* a um SIF, invocando a inferência de regras *fuzzy* somente quando necessário, além de considerar uma maior expressividade com relação às inferências *fuzzy* que *HyFOM*. Embora sejam baseadas em diferentes arquiteturas, ambas abordagens são capazes de integrar ontologias *fuzzy* e sistemas de inferência *fuzzy*, cada qual com vantagens e desvantagens decorrentes da estratégia de integração. Além disso, ambas abordagens consideram uma arquitetura flexível com relação à instanciação de SIF que seja mais adequado para a semântica das aplicações.

Para descrever as abordagens desenvolvidas, este capítulo está organizado da seguinte forma. Primeiramente, a Seção 5.2 apresenta qual o contexto específico em que *HyFOM* e *FT-FIS* realizam a integração, considerando requisitos de representação e de raciocínio envolvendo ontologias *fuzzy* e sistemas de inferência *fuzzy*. Na sequência, a Seção 5.3 descreve o motor de inferência *HyFOM*, sua arquitetura e a abordagem de integração de seus componentes. Por fim, na Seção 5.4 é apresentada a abordagem *FT-FIS*, com foco na interação entre um algoritmo baseado em *tableau fuzzy* e um sistema de inferência *fuzzy*.

## 5.2 Contexto de Integração de Ontologias Fuzzy e Sistemas de Inferência Fuzzy

Conforme mencionado anteriormente, a integração de raciocínio de ontologias *fuzzy* e de sistemas de inferência *fuzzy* envolve propriedades numéricas e termos linguísticos. Neste contexto, existem alguns requisitos de representação e de raciocínio em aplicações baseadas em ontologias que evidenciam a necessidade por esse tipo de integração. Tais requisitos podem ocorrer em diversos domínios, assemelhando-se aos casos de uso de padrões de projeto ontológicos (GANGEMI; PRESUTTI, 2009), que correspondem a soluções de modelagem para

resolver problemas recorrentes de projeto de ontologias.

Em termos gerais, uma situação típica em ontologias *fuzzy* que demanda inferência baseada em regras *fuzzy* é a definição de conceitos abstratos em função de valores de propriedades concretas numéricas. Neste caso, seja uma propriedade ou atributo concreto  $T$  definido no domínio dos números reais,  $p1$  e  $p2$  predicados que restringem os valores de  $T$ . Um conceito  $C1$  pode ser definido a partir de um predicado  $p1$  que corresponde a uma restrição *crisp* definida por  $p = \bowtie n$ , onde  $\bowtie$  pode ser algum operador relacional em  $\{\leq, <, \geq, >, =, \neq\}$  e  $n \in \mathbb{R}$  (Equação 5.1). Um conceito  $C2$  pode ser definido com base em um predicado  $p2$ , que corresponde a um termo linguístico  $t$  representado por um conjunto *fuzzy* (Equação 5.2).

$$C1 = \exists T.p1 \mid p1 = \bowtie n, \bowtie \in \{\leq, <, \geq, >, =, \neq\} \text{ e } n \in \mathbb{R} \quad (5.1)$$

$$C2 = \exists T.p2 \mid p2 = t \text{ e } t \text{ é definido por um conjunto } \textit{fuzzy} \text{ em } \mathbb{R} \quad (5.2)$$

Os conceitos  $C1$  e  $C2$ , modelados na ontologia *fuzzy*, dependem dos valores da propriedade concreta numérica  $T$  para que a pertinência das instâncias da ontologia a esses conceitos seja conhecida. Os valores de  $T$  podem ser determinados com base em regras *fuzzy* que combinam outras propriedades concretas numéricas  $T_1, \dots, T_n$ , também modeladas na ontologia. Dependendo da semântica do domínio, as regras *fuzzy* podem ser adequadas por serem capazes de tratar a imprecisão por meio dos termos linguísticos e do processo de inferência, que leva em consideração operações *fuzzy* tradicionais da teoria de conjuntos *fuzzy*. Neste sentido, o uso de regras *fuzzy* pode ser mais apropriado que o uso de regras que combinam restrições de intervalos numéricos rígidos.

Para realizar a inferência com base nas regras *fuzzy*, um sistema de inferência *fuzzy* pode ser acionado, retornando o valor inferido da propriedade  $T$  a partir de valores das propriedades  $T_1, \dots, T_n$  obtidas de instâncias da ontologia *fuzzy*. Tais valores podem ser explícitos ou obtidos por meio de inferências da ontologia *fuzzy*. Para exemplificar valores de propriedades derivados de inferências da ontologia *fuzzy*, suponha a definição de conceito *fuzzy*  $C3 = \exists T_1.high$  juntamente com a asserção  $C3(ind1) \geq 1.0$ , que permitem inferir que a instância  $ind1$  possui valor  $high \geq 1.0$  para a propriedade  $T_1$ . O valor retornado pelo SIF deve ser considerado para determinar a pertinência dos indivíduos da ontologia para os conceitos  $C1$  e  $C2$ , contribuindo para responder consultas das aplicações à ontologia *fuzzy*.

Para exemplificar essa situação em um domínio específico, suponha que uma loja de comércio eletrônico possua uma regra de negócio que defina o preço do frete de um pedido a partir de outras propriedades, tais como peso, volume e a distância para o local de entrega, todas

modeladas em uma ontologia *fuzzy* como propriedades de tipo de dado numérico real. Com base na experiência dos especialistas nesse domínio, regras *fuzzy* podem ser definidas combinando essas propriedades para se obter o preço do frete, conforme exemplificado na Tabela 5.1.

**Tabela 5.1: Regras *fuzzy* para obter o preço do frete em função de outras propriedades.**

se o peso é leve e o volume é baixo e a distância é pequena então o preço do frete é baixo se o peso é leve e o volume é alto e a distância é longa então o preço do frete é médio se o peso é pesado e o volume é alto e a distância é longa então o preço do frete é alto
---

Suponha também uma instância *pedido1* da ontologia *fuzzy*, que representa um pedido específico de um cliente. Essa instância pode ter asserções referentes às propriedades *peso* e *volume*, além de pertencer à categoria de *PedidoExterior*, que se refere a pedidos para serem entregues em outros países:

$$\begin{aligned}
 & \textit{PedidoExterior} = \exists \textit{distancia.muitoLonga} \\
 & \textit{peso}(\textit{pedido1}, 2.0) \\
 & \textit{volume}(\textit{pedido1}, 2.5) \\
 & \textit{PedidoExterior}(\textit{pedido1}) \geq 1.0
 \end{aligned}$$

Assim, para inferir qual o preço do frete para a instância *pedido1* com base nas regras da Tabela 5.1, os valores das propriedades *peso*, *volume* e *distancia* podem ser obtidos da ontologia *fuzzy*. Neste exemplo, os valores numéricos das duas primeiras propriedades estão modelados explicitamente, mas o valor da distância pode ser derivado da aplicação da regra de inferência do quantificador existencial presente na definição do conceito *PedidoExterior*. Assim, o valor da propriedade *distancia* para a instância *pedido1* é *muitoLonga* com grau  $\geq 1.0$ , conforme a inferência provida por raciocinadores de ontologia *fuzzy*, sendo que *muitoLonga* é um termo linguístico definido por um conjunto *fuzzy*. Com isso, as entradas para um SIF podem ser definidas a fim de se obter o valor da propriedade *precoFrete* para a instância *pedido1*.

Além disso, um conceito *BonusProximaCompra* pode ser definido na ontologia *fuzzy* para representar o bônus que um cliente pode obter para a próxima compra com base no preço do frete (Equação 5.3). Neste caso, a restrição é *fuzzy*, pois *alto* é definido como um termo linguístico da propriedade *precoFrete*.

$$\textit{BonusProximaCompra} = \textit{Cliente} \sqcap \exists \textit{possuiPedido} . (\exists \textit{precoFrete.alto}) \quad (5.3)$$

Assim, para definir qual a pertinência das instâncias do conceito *Cliente* para o conceito *BonusProximaCompra*, é necessário saber qual o valor do frete dos pedidos. É nesse tipo de situação que as abordagens de integração desenvolvidas (HyFOM e FT-FIS) atuam, permitindo que os resultados do FIS estejam disponíveis para a ontologia *fuzzy* quando necessário. Os valores de entrada do SIF são obtidos da ontologia *fuzzy* de forma direta (asserções) ou por meio de

inferências de raciocinadores de ontologias *fuzzy*, mostrando que os dois tipos de raciocínio são necessários para satisfazer aos requisitos das aplicações.

### 5.3 *HyFOM: Hybrid integration of Fuzzy Ontology and Mamdani reasoning*

O motor de inferência híbrido HyFOM foi desenvolvido para combinar as inferências de ontologias *fuzzy* com os resultados do raciocínio de sistemas de inferência *fuzzy*, com foco em valores de propriedades numéricas derivados a partir de regras *fuzzy*. A primeira versão de HyFOM foi desenvolvida com suporte a SIF do tipo Mamdani (origem do nome HyFOM), conforme descrito por Yaguinuma et al. (2013, 2013b, 2013a). Posteriormente, HyFOM foi estendido para o uso de SIF genéricos, com suporte para métodos de inferência de Mamdani e Larsen. HyFOM foi projetado a fim de contemplar tanto a integração de inferências de ontologias *fuzzy* para SIF quanto dos resultados do SIF para a ontologia *fuzzy*, conforme discutido no Capítulo 4:

- *Ontologia fuzzy*  $\rightarrow$  *SIF*: as instâncias (ou indivíduos) do domínio estão representadas na ontologia *fuzzy*, associadas a propriedades concretas numéricas. Algumas dessas propriedades são utilizadas em um SIF tanto como atributos de entrada quanto como atributos de saída. Os valores das propriedades correspondentes aos atributos de entrada do SIF em questão podem ser obtidos da ontologia *fuzzy*, para cada instância, caso sejam representados como asserções explícitas ou resultem de inferências de um raciocinador de ontologias. Desta forma, as entradas do SIF são providas pela ontologia *fuzzy*;
- *SIF*  $\rightarrow$  *ontologia fuzzy*: os resultados do raciocínio do SIF são incorporados à ontologia *fuzzy* como novas asserções de propriedades numéricas, associando as instâncias correspondentes aos valores inferidos a partir de regras *fuzzy*. Essas novas asserções disponibilizam os resultados do SIF para a ontologia *fuzzy*, contribuindo para a realização de tarefas de inferência como a verificação de instâncias.

Para realizar a integração de inferências, o motor de inferência HyFOM segue uma arquitetura híbrida, fracamente acoplada, que utiliza implementações de motores de inferência existentes para ontologias e SIF. Essa arquitetura foi escolhida em função da agilidade de desenvolvimento com o reuso de componentes, que permite um foco maior na concepção da abordagem de integração. As próximas seções detalham a arquitetura de HyFOM, o procedimento de integração desenvolvido e o protótipo implementado.

### 5.3.1 Arquitetura e Componentes de HyFOM

A arquitetura do motor de inferência híbrido HyFOM, que combina raciocinadores de ontologias e um sistema de inferência *fuzzy*, é ilustrada na Figura 5.1. Na parte inferior da figura, a base de conhecimento é composta por uma ontologia *fuzzy* e por uma base de regras *fuzzy*. A ontologia *fuzzy* modela a semântica de um domínio específico em termos de conceitos, propriedades e relacionamentos *fuzzy*, além de instâncias e termos linguísticos associados a propriedades numéricas. A base de regras *fuzzy* contém um conjunto de regras combinando propriedades numéricas e termos linguísticos no antecedente para inferir valores de propriedades numéricas presentes nos consequentes das regras. As definições das propriedades numéricas e dos termos linguísticos devem ser as mesmas tanto na ontologia *fuzzy* quanto nas regras *fuzzy* para manter a consistência do vocabulário usado no domínio. Para que HyFOM realize a integração das inferências, o domínio em questão deve apresentar alguma situação semelhante ao contexto de integração descrito na Seção 5.2, representada na ontologia *fuzzy* e na base de regras *fuzzy*.

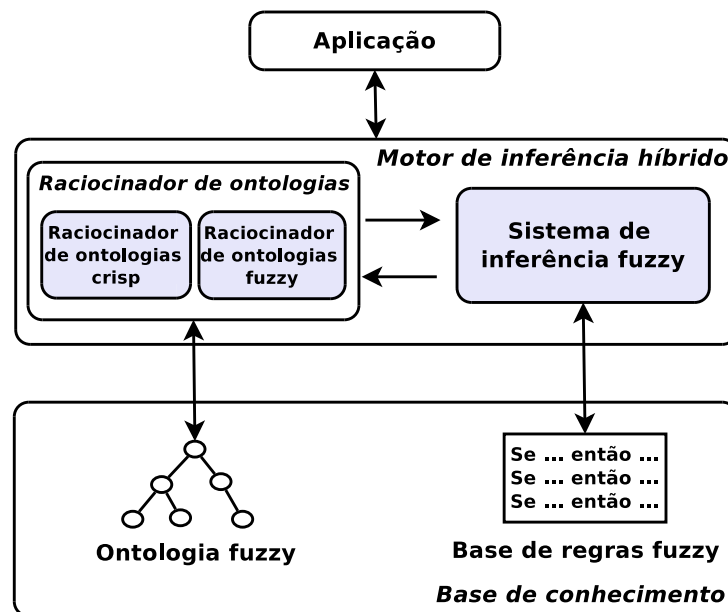


Figura 5.1: Arquitetura de HyFOM para integração de raciocínio de ontologias *fuzzy* e sistemas de inferência *fuzzy*.

A base de conhecimento é processada pelo módulo do motor de inferência híbrido, composto por dois módulos principais: um raciocinador de ontologias e um sistema de inferência *fuzzy*. Como HyFOM segue uma arquitetura híbrida, os componentes baseados em motores de inferência existentes na literatura são indicados na Figura 5.1 como módulos de cor acinzentada.

O raciocinador de ontologias é responsável por realizar tarefas de raciocínio tais como a verificação de instâncias e a verificação de satisfazibilidade e de consistência, sendo composto por

um raciocinador de ontologias *crisp* e um raciocinador de ontologias *fuzzy*. O raciocinador de ontologias *crisp* é um motor de inferência de ontologias convencionais, sem suporte a definições *fuzzy*, utilizado principalmente para prover consultas otimizadas a inferências *crisp* e asserções associadas a propriedades numéricas na ontologia. O raciocinador de ontologias *fuzzy*, por sua vez, é baseado em lógica de descrição *fuzzy*, portanto capaz de realizar tarefas de inferência como a verificação de instâncias *fuzzy*, verificação de satisfazibilidade e de consistência com base em definições e asserções *fuzzy* de conceitos e relacionamentos. O sistema de inferência *fuzzy* é responsável por efetuar inferências envolvendo propriedades numéricas e regras *fuzzy*.

Na parte superior da figura, aplicações acionam o motor de inferência híbrido, listando instâncias da ontologia *fuzzy* que devem ser verificadas para possíveis inferências envolvendo propriedades numéricas e regras *fuzzy*. As instâncias devem estar associadas ao contexto de integração apresentado na Seção 5.2, expressando, assim, a demanda pela integração das inferências de SIF para realizar a tarefa de verificação de instâncias do raciocínio de ontologia *fuzzy*.

### 5.3.2 Abordagem de Integração

Os componentes da arquitetura de HyFOM são integrados de acordo com a abordagem descrita no Algoritmo 1. O processo é iniciado pela aplicação, que invoca HyFOM para verificar se uma instância ou conjunto de instâncias da ontologia possui alguma propriedade numérica cujo valor possa ser inferido com base em um sistema de inferência *fuzzy*. No caso do exemplo da Seção 5.2, a aplicação invocaria HyFOM para verificar a instância *pedido1*, cujo valor da propriedade *precoFrete* pode ser inferido a partir de regras *fuzzy*. Essa propriedade corresponde à propriedade de saída do sistema de inferência *fuzzy*, referente ao consequente das regras *fuzzy* (linha 2 do Algoritmo 1). HyFOM trabalha com regras contendo um atributo no consequente, por isso a função *getOutputProperty* retorna somente uma propriedade de saída.

Para cada instância, HyFOM primeiramente verifica se o valor da propriedade de saída pode ser obtido por meio do módulo raciocinador de ontologias, ao invocar a função *getPropValue* (linha 3). Caso seja possível, a ontologia *fuzzy* é capaz de prover o valor da propriedade de saída sem a necessidade de envolver o raciocínio do SIF. Caso contrário, o SIF é acionado para realizar a inferência do valor de saída (linha 12), desde que os valores das propriedades de entrada estejam disponíveis. As propriedades utilizadas como entradas do SIF são obtidas pela função *getInputProperties* (linha 1) e os seus respectivos valores são obtidos da ontologia *fuzzy* (linhas 5 a 10) para cada instância listada pela aplicação.

Para obter os valores de propriedades a partir da ontologia *fuzzy*, HyFOM utiliza a função

**Algorithm 1:** Abordagem de integração de HyFOM

---

```

input   : Fuzzy ontology individual ind;
result  : Fuzzy ontology with a new property assertion;
initialize:
    fuzOnt ← Fuzzy ontology;
    fuzRuleBase ← Fuzzy rule base;
    crispOntR ← Crisp Ontology Reasoner;
    fuzOntR ← Fuzzy Ontology Reasoner;
    FIS ← Fuzzy Inference System;
1 inputProperties ← FIS.getInputProperties(fuzRuleBase);
2 outputProp ← FIS.getOutputProperty(fuzRuleBase);
3 outputValue ← getPropValue(ind,outputProp);
4 if outputValue =  $\emptyset$  then
5     for  $i \leftarrow 0$  to inputProperties.size()−1 do
6         inputProp ← inputProperties [i];
7         inputValue ← getPropValue(ind,inputProp);
8         if inputValue  $\neq \emptyset$  then inputValues [i] ← inputValue;
9         ;
10        else throw exception "missing input value for FIS";
11        ;
12    end
13    if inputValues  $\neq \emptyset$  then
14        outputValue ← FIS.getOutput(fuzRuleBase,inputValues);
15        expertAuthorization ← expertSupervision(fuzOnt,ind,outputProp, outputValue);
16        if expertAuthorization = true then
17            crispOntR.newAssertion(fuzOnt,ind,outputProp, outputValue);
18        end
19    end
20 end

```

---

*getPropValue*, apresentada no Algoritmo 2. De acordo com essa função, o raciocinador de ontologia *crisp* é primeiramente acionado para verificar se o valor da propriedade em questão está modelado explicitamente como uma asserção ou pode ser inferido a partir de definições *crisp* da ontologia. Conforme mencionado anteriormente, o raciocinador de ontologia *crisp* foi considerado principalmente em função de seu acesso otimizado a asserções de propriedades numéricas e tarefas de raciocínio de ontologias convencionais que podem inferir valores de propriedades numéricas para instâncias da ontologia. Por exemplo, a definição de um conceito  $C = \exists prop1.\{10\}$  estabelece que instâncias do conceito  $C$  possuam uma restrição no valor da propriedade *prop1* associada ao valor específico 10. Essa inferência pode ser obtida por raciocinadores de ontologias tradicionais, com base em regras de inferência envolvendo construtores de conceito e restrições de valores de propriedades por nominais, isto é, indivíduos abstratos ou concretos. Neste exemplo, dada uma asserção  $C(a)$ , o raciocinador de ontologias *crisp* deve considerar que existe pelo menos uma associação entre a instância  $a$  e o valor 10 pela



propriedade *prop1*.

---

**Algorithm 2:** Função *getPropValue*

---

```

input   : Fuzzy ontology individual ind, Property prop;
output  : prop value for ind
initialize:
    fuzOnt ← Fuzzy ontology;
    crispOntR ← Crisp Ontology Reasoner;
    fuzOntR ← Fuzzy Ontology Reasoner;
1 propValue ← crispOntR.getPropValue(fuzOnt,ind,prop);
2 if propValue = ∅ then
3   | propValue ← fuzOntR.getPropValue(fuzOnt,ind,prop);
4 end
5 return propValue

```

---

Se os valores de propriedades não puderem ser obtidos por asserções ou inferências do raciocinador de ontologias *crisp*, o raciocinador de ontologias *fuzzy* é acionado, pois algumas tarefas de raciocínio sobre definições de conceitos *fuzzy* e implicações podem prover valores de propriedades, conforme exemplificado na Seção 5.2. Portanto, mesmo que os valores das propriedades de entrada não estejam explícitos na ontologia *fuzzy*, eles podem ser inferidos com base em axiomas e definições *fuzzy* presentes na ontologia. Os valores inferidos podem ser valores numéricos específicos ou representados por conjuntos *fuzzy*, dependendo das preferências da aplicação. Assim, as inferências da ontologia *fuzzy* são utilizadas para definir as entradas do SIF, representando a integração *ontologia fuzzy* → *SIF*.

A função *getPropValue* é acionada pelo algoritmo de integração de HyFOM em duas ocasiões:

1. para verificar se a ontologia *fuzzy* já provê o valor da propriedade de saída (linha 3 do Algoritmo 1);
2. para obter valores de propriedades de entrada que serão enviados ao SIF (linha 7 do Algoritmo 1). Nesta situação, HyFOM pode ser configurado para tratar entradas numéricas ou *fuzzy*, caso sejam obtidas pelo raciocinador de ontologias *fuzzy*.

Os valores das propriedades de entrada obtidos com a função *getPropValue* são submetidos ao SIF para que o valor da propriedade de saída seja derivado com base em algum método de inferência sobre regras *fuzzy* (linha 12 do Algoritmo 1). O sistema de inferência *fuzzy* pode ser um sistema de Mamdani ou Larsen, ou outro tipo de SIF que seja adequado dependendo da semântica da aplicação. Para gerar o valor numérico da saída *fuzzy* inferida pelo processamento das regras *fuzzy*, diversos métodos de defuzzificação podem ser usados, como o centro de área,

método *Moment*, meio dos máximos, entre outros (Seção 2.5.3 do Capítulo 2). A base de regras *fuzzy* deve conter a especificação do método de inferência a ser usado e do método de defuzzificação da propriedade de saída.

Depois do processo de inferência do SIF, o valor da propriedade de saída é retornado à ontologia *fuzzy* como uma nova asserção de propriedade, sob a supervisão de especialistas do domínio (linhas 13 a 16 do Algoritmo 1). Assim, os resultados do SIF são disponibilizados para outras tarefas de raciocínio da ontologia *fuzzy*, como a verificação de instâncias ou verificação de satisfazibilidade, atendendo à integração  $SIF \rightarrow ontologia\ fuzzy$ . A supervisão dos especialistas pode ser realizada para uma instância específica como também para um conjunto de instâncias e suas respectivas saídas defuzzificadas. A intervenção humana é necessária em virtude da adição de novas asserções na ontologia *fuzzy*, que devem ser realizadas com o aval dos especialistas de domínio pois modificam o estado da base de conhecimento. As saídas do SIF incorporadas à ontologia *fuzzy* são numéricas, resultantes de defuzzificação, visto que as saídas *fuzzy* geradas pelo SIF podem ser de difícil interpretação para os especialistas.

### 5.3.3 Protótipo HyFOM

A arquitetura híbrida e o processo de integração de HyFOM foram implementados em um protótipo, utilizando-se raciocinadores e *frameworks* disponíveis para aplicações baseadas em ontologias *fuzzy* e de sistemas de inferência *fuzzy* (módulos em cor acinzentada na Figura 5.1):

- *Raciocinador de ontologias crisp*: utilizou-se a biblioteca OWL API (HORRIDGE; BECHHOFFER, 2011) para desenvolvimento de aplicações baseadas em ontologias, combinada com Hermit (MOTIK; SHEARER; HORROCKS, 2009), um motor de inferência otimizado que provê acesso eficiente a asserções e inferências envolvendo ontologias *crisp*;
- *Raciocinador de ontologias fuzzy*: fuzzyDL (BOBILLO; STRACCIA, 2008), que oferece suporte a conceitos *fuzzy*, termos linguísticos e axiomas *fuzzy*, além de uma API Java para invocar as tarefas de raciocínio;
- *Sistema de inferência fuzzy*: FuzzyJ Toolkit e Fuzzy Jess (ORCHARD, 2001), que implementam os métodos de inferência de Mamdani e Larsen e defuzzificação por centro de área, meio dos máximos e momento. Também disponibilizam uma API Java para acionar o processo de inferência e obter as saídas defuzzificadas.

Para implementar a abordagem de integração de HyFOM e a interface entre os raciocinadores de ontologias e o SIF foi utilizada a plataforma Java, uma vez que todas as implementações mencionadas disponibilizam API Java para acionar os processos de inferência.

A representação da ontologia *fuzzy* baseia-se na linguagem OWL 2 (GRAU et al., 2008) e na linguagem Fuzzy OWL 2 (BOBILLO; STRACCIA, 2011), que estende OWL 2 com comentários estruturados contendo informações sobre definições *fuzzy*. Assim, especialistas de domínio podem modelar a ontologia *fuzzy* por meio do editor de ontologias Protégé juntamente com o *plugin* de Fuzzy OWL 2. Como as extensões da linguagem Fuzzy OWL 2 correspondem a comentários na linguagem OWL 2, a ontologia gerada por Protégé pode ser processada normalmente por OWL API e Hermit. Uma versão da ontologia *fuzzy* deve ser traduzida para a sintaxe de fuzzyDL, por meio de um tradutor disponibilizado pelo *plugin* de Fuzzy OWL 2, para permitir o processamento das definições e asserções *fuzzy* da ontologia. Os resultados inferidos pelo SIF são integrados à ontologia *fuzzy* por meio dos métodos de OWL API para inclusão de novas asserções de propriedade. As novas asserções também devem ser adicionadas à ontologia *fuzzy* em sintaxe fuzzyDL para manter a consistência entre as versões.

As regras *fuzzy* são descritas na linguagem *Fuzzy Markup Language* (FML) (ACAMPORA; LOIA, 2005), que conta com uma ferramenta visual (*Visual FML Tool*<sup>1</sup>). Essa ferramenta gráfica permite modelar sistemas de Mamdani e Larsen, sendo que especialistas de domínio podem escolher qual a semântica de regra, operador de agregação e métodos de defuzzificação que mais se adequam à semântica do domínio em questão. A representação das regras *fuzzy* em FML proporciona a HyFOM uma maior interoperabilidade e independência de implementação de SIF, por se tratar de uma linguagem genérica para especificação de SIF. Neste sentido, outras implementações de SIF podem ser utilizadas, desde que haja um tradutor de FML para a linguagem específica do SIF. Como FML é baseada em XML, tradutores podem ser construídos por meio de documentos baseados em *Extensible Stylesheet Language Transformations* (XSLT), que transformam documentos XML para outros formatos de arquivo texto. No caso específico do protótipo implementado, as bases de regras *fuzzy* em FML são traduzidas para a sintaxe de Fuzzy Jess.

O protótipo de HyFOM foi utilizado em experimentos descritos com maiores detalhes no Capítulo 6, onde é apresentada uma discussão sobre os resultados obtidos em termos de semântica das regras e integração das inferências.

---

<sup>1</sup>[www.corisa.it/fml/Download\\_files/VisualFMLTool%200.1.rar](http://www.corisa.it/fml/Download_files/VisualFMLTool%200.1.rar)

## 5.4 FT-FIS: Fuzzy Tableau and Fuzzy Inference System

A segunda abordagem desenvolvida para integração de inferências de ontologias *fuzzy* e de SIF é FT-FIS, projetada como uma abordagem mais formal e fortemente acoplada que HyFOM. FT-FIS combina um algoritmo baseado em *tableau fuzzy* com SIF por meio de uma arquitetura flexível inspirada nos trabalhos de Pan (2007), Wang e Ma (2008), que foi estendida, neste trabalho de doutorado, para tratar os requisitos específicos da integração com SIF. Neste contexto, foi definido em FT-FIS um predicado específico para acionar o raciocínio de SIF, chamado *fuzzyRuleReasoning*, que pode ser utilizado em definições de conceitos abstratos da ontologia *fuzzy*. Esse predicado atende aos requisitos de modelagem referentes ao contexto de integração descrito na Seção 5.2.

Assim como HyFOM, FT-FIS considera tanto a integração *ontologia fuzzy*  $\rightarrow$  SIF quanto SIF  $\rightarrow$  *ontologia fuzzy*. Ou seja, as inferências da ontologia *fuzzy* contribuem para definir as entradas enviadas ao SIF, assim como as saídas do SIF são utilizadas para a definição de pertinência de instâncias a conceitos abstratos modelados na ontologia *fuzzy*. A principal diferença, no caso de FT-FIS, é que a integração é realizada entre um algoritmo baseado em *tableau fuzzy*, que aciona o SIF toda vez que houver referência ao predicado *fuzzyRuleReasoning*. Assim, qualquer consulta realizada à ontologia *fuzzy* pode acionar o SIF, pois o algoritmo baseado em *tableau fuzzy* e suas regras de expansão se encarregam de decompor a consulta e acionar os raciocinadores de tipos de dados *fuzzy* quando necessário. Além disso, as saídas do SIF não são consolidadas na ontologia *fuzzy* como novas asserções, apenas fazem parte do processo de inferência. Desta forma, o estado da ontologia *fuzzy* não é modificado, como ocorre em HyFOM, fato que demanda supervisão dos especialistas do domínio.

A abordagem FT-FIS é detalhada nas próximas seções, com a descrição do formalismo de representação de conhecimento considerado, da arquitetura de raciocínio desenvolvida e da abordagem de integração, fundamentada por um algoritmo baseado em *tableau fuzzy*. Por fim, alguns aspectos referentes ao protótipo implementado são apresentados.

### 5.4.1 Representação de Conhecimento

A base de conhecimento processada por FT-FIS é composta por uma ontologia *fuzzy* e por um conjunto de regras *fuzzy*. Na sequência, são descritos os respectivos formalismos adotados para representação de conhecimento, cuja explicação mais detalhada está presente nos Capítulos 3 e 2.

### Ontologia *fuzzy*

O formalismo-base para representação de ontologias *fuzzy* escolhido foi a Lógica de Descrição *fuzzy*  $F\text{-}\mathcal{ALC}(G)$ , que combina a DL  $\mathcal{ALC}$  com grupos de tipos de dado *fuzzy*  $G$  (WANG; MA, 2008). Embora existam na literatura outras *fuzzy* DLs mais expressivas que *fuzzy*  $\mathcal{ALC}$ , ela foi adotada em virtude de sua simplicidade aliada à expressividade necessária para representar os requisitos de modelagem descritos na Seção 5.2. Além disso, como o foco da abordagem FT-FIS está na integração das inferências e não somente na expressividade da ontologia *fuzzy*, optou-se por um formalismo mais simples, que pode ser estendido para outras Lógicas de Descrição *Fuzzy* mais expressivas.

Outra característica importante de  $F\text{-}\mathcal{ALC}(G)$  que vai ao encontro dos requisitos de modelagem envolvendo a integração de inferências de ontologias *fuzzy* e de SIF é o suporte a tipos de dados concretos *fuzzy*. Em especial, as *fuzzy* DLs que oferecem suporte a grupos de tipos de dados *fuzzy* (denotadas com o sufixo  $(G)$ ) são apropriadas para FT-FIS por representarem predicados concretos *fuzzy* personalizados e expressões de tipos de dados *fuzzy*. Assim,  $F\text{-}\mathcal{ALC}(G)$  permite que novos predicados concretos *fuzzy* sejam definidos e combinados com outros predicados, formando expressões de tipos de dados *fuzzy*. Como FT-FIS define um predicado concreto *fuzzy* associado ao raciocínio de SIF, é necessário que o formalismo de representação de ontologias *fuzzy* seja flexível com relação à definição de novos predicados concretos *fuzzy*.

O formalismo de  $F\text{-}\mathcal{ALC}(G)$  é descrito com detalhes na Seção 3.5 do Capítulo 3. A semântica das operações *fuzzy* segue a lógica *fuzzy* de Zadeh, conforme ilustrado na Tabela 3.5 também do Capítulo 3. Na abordagem FT-FIS, foram consideradas suposições em algumas definições e axiomas de  $F\text{-}\mathcal{ALC}(G)$  favorecendo a computabilidade e a expressividade do domínio:

- *Propriedades concretas funcionais*: para simplificar o processamento das inferências envolvendo propriedades concretas, optou-se pela funcionalidade. Assim, para uma propriedade concreta  $T$  e um indivíduo abstrato  $a$ , existe no máximo um único indivíduo concreto  $v$  em asserções da ontologia *fuzzy*;
- *TBox definitória*: conforme descrito na Seção 3.3.1, são permitidas apenas definições e subsunções de conceito não ambíguas e acíclicas. Ou seja, uma TBox  $T$  contém no máximo uma definição para um nome de conceito  $A$  e a definição de  $A$  não pode fazer referência (direta ou indireta) ao próprio conceito  $A$ . De acordo com Baader, Horrocks e Sattler (2007), uma TBox definitória é interessante do ponto de vista computacional porque possibilita técnicas de raciocínio de complexidade menor que de uma TBox geral;

- *Asserções positivas*: a exemplo de (BOBILLO; STRACCIA, 2008), são permitidas asserções positivas do tipo  $\langle a : C \geq n \rangle$  e  $\langle (a, b) : R \geq n \rangle$ . Segundo Mailis, Stoilos e Stamou (2007), asserções positivas contêm operadores relacionais  $\{\geq, >\}$  enquanto asserções negativas referem-se aos operadores  $\{\leq, <\}$ , sendo que as últimas podem implicar em indecidibilidade. Além disso, Straccia (2001) argumenta que relações estritas contendo os operadores  $\{<, >\}$  correspondem a situações geralmente incomuns em termos de semântica de asserções de domínio;
- *Expressões de tipos de dados fuzzy*: as expressões de tipos de dados fuzzy devem conter predicados de um mesmo tipo de dado base, ou seja, dada uma expressão  $E$  formada por predicados  $p_i$ ,  $1 \leq i \leq n$ ,  $dom(p_i) = d$ .

Com relação à representação de propriedades numéricas e predicados de tipos de dados fuzzy, a abordagem FT-FIS segue as definições da teoria de grupos de tipos de dados fuzzy apresentadas na Seção 3.5.4 do Capítulo 3. Assim, é possível definir conceitos abstratos a partir de restrições sobre propriedades concretas ( $T_1, \dots, T_n$ ) envolvendo predicados e expressões de tipos de dados fuzzy ( $E$ ) por meio dos construtores:

$$\begin{aligned} (\forall T_1, \dots, T_n. E)^{\mathcal{I}}(a) &= \inf_{v_1, \dots, v_n \in \Delta_D} (\bigotimes_{i=1}^n T_i^{\mathcal{I}}(a, v_i)) \Rightarrow E^D(v_1, \dots, v_n) \\ (\exists T_1, \dots, T_n. E)^{\mathcal{I}}(a) &= \sup_{v_1, \dots, v_n \in \Delta_D} (\bigotimes_{i=1}^n T_i^{\mathcal{I}}(a, v_i)) \otimes E^D(v_1, \dots, v_n), \end{aligned}$$

sendo que  $x \otimes y = \min(x, y)$  e  $x \Rightarrow y = \max(1 - x, y)$ , onde  $x, y \in [0, 1]$ , correspondendo ao conjunto de operações da semântica da lógica fuzzy de Zadeh (LUKASIEWICZ; STRACCIA, 2008).  $\mathbf{Dexp}(G)$  representa o conjunto de expressões de tipos de dados fuzzy  $E$  que podem ser construídas a partir de conjunções, disjunções, e negações aplicadas a predicados concretos atômicos  $p$ , conforme descrito na Tabela 3.8 do Capítulo 3.

Como FT-FIS deve oferecer suporte à representação e ao processamento de propriedades concretas numéricas, pelo menos o tipo de dados referente ao domínio dos números reais deve estar contemplado como um dos tipos de dados base  $d_{real} \in D_G$ , segundo a definição de grupo de tipo de dado dada pela tripla  $(\phi_G, D_G, dom)$ . Cada predicado  $p \in \phi_G$  deve ser definido sobre um tipo de dado base  $d \in D_G$  tal que  $X(p) \subseteq dom(p)$ , onde  $dom(p) = V(d)^{a(p)}$ .

FT-FIS também pressupõe predicados embutidos  $p$  definidos sobre o tipo de dado dos números reais  $d_{real}$ :  $p \in \phi_G \mid dom(p) = V(d_{real})^{a(p)}$ , onde  $a(p)$  é a aridade do predicado  $p$  e  $V(d_{real})$  é conjunto de valores de  $d_{real}$ . Os predicados embutidos básicos oferecidos por FT-FIS no domínio dos números reais podem ser categorizados em predicados *crisp* unários e predicados *fuzzy* (Tabela 5.2). Esses tipos de predicados são baseados nos predicados propostos por Pan (2007) e Wang e Ma (2008), respectivamente. No âmbito da abordagem FT-FIS, o

predicado *fuzzyRuleReasoning* é definido como um predicado *fuzzy* com o objetivo de combinar o raciocínio de ontologias *fuzzy* e de SIF.

Predicados <i>crisp</i> unários	Predicados <i>fuzzy</i>
$crispInterval_{k1,k2,a,b}$	$leftShoulder_{k1,k2,a,b}$
$=_y, \neq_y$	$rightShoulder_{k1,k2,a,b}$
$<_y, >_y$	$triangular_{k1,k2,a,b,c}$
$\leq_y, \geq_y$	$trapezoidal_{k1,k2,a,b,c,d}$
	$fuzzyRuleReasoning$

**Tabela 5.2: Predicados embutidos de FT-FIS no domínio dos números reais.**

Em geral, os predicados *crisp* unários correspondem a intervalos *crisp* definidos sobre  $[k1, k2] \subseteq V(d_{real})$  referente ao domínio da propriedade concreta restrita pelo predicado. Sejam  $a, b \in [k1, k2]$ , com  $a \leq b$ , a extensão do predicado  $crispInterval_{k1,k2,a,b}$  é definida por:

$$X(crispInterval_{k1,k2,a,b}) = \{x \in V(d_{real}) \mid$$

$$crispInterval_{k1,k2,a,b}(x) = \begin{cases} 1, & \text{se } x \in [a, b] \\ 0, & \text{caso contrário} \end{cases} \quad (5.4)$$

$$\}.$$

A extensão de  $\neg crispInterval_{k1,k2,a,b}$  é obtida pela operação de complemento padrão *fuzzy* (Equação 2.7 do Capítulo 2) sobre sua função de pertinência.

A extensão dos demais predicados *crisp* unários pode ser definida a partir do predicado  $crispInterval_{k1,k2,a,b}$ :

- $X(=_y) = \{x \in V(d_{real}) \mid =_y(x) = crispInterval_{k1,k2,y,y}(x)\};$
- $X(\neq_y) = \{x \in V(d_{real}) \mid \neq_y(x) = \neg crispInterval_{k1,k2,y,y}(x)\};$
- $X(\leq_y) = \{x \in V(d_{real}) \mid \leq_y(x) = crispInterval_{k1,k2,k1,y}(x)\};$
- $X(\geq_y) = \{x \in V(d_{real}) \mid \geq_y(x) = crispInterval_{k1,k2,y,k2}(x)\};$
- $X(<_y) = \{x \in V(d_{real}) \mid <_y(x) = \neg crispInterval_{k1,k2,y,k2}(x)\};$
- $X(>_y) = \{x \in V(d_{real}) \mid >_y(x) = \neg crispInterval_{k1,k2,k1,y}(x)\};$
- $X(\neg =_y) = X(\neq_y);$
- $X(\neg \neq_y) = X(=_y);$
- $X(\neg >_y) = X(\leq_y);$

- $X(\neg <_y) = X(\geq_y)$ ;
- $X(\neg \geq_y) = X(<_y)$ ;
- $X(\neg \leq_y) = X(>_y)$ .

Um exemplo de descrição de conceito com um predicado unário *crisp* é  $Item \sqcap \exists peso. >_5$ , que restringe o valor da propriedade concreta *peso* dos indivíduos do conceito *Item* para valores maiores que 5.

Os predicados *fuzzy* são predicados que correspondem a conjuntos *fuzzy* definidos sobre o domínio dos números reais. FT-FIS considera predicados *fuzzy* unários definidos segundo funções de pertinência parametrizadas como funções triangulares e trapezoidais (Seção 2.2 do Capítulo 2). Considerando que os predicados *fuzzy* são definidos em um intervalo  $[k1, k2] \subseteq V(d_{real})$  com parâmetros  $a, b, c, d \in [k1, k2]$  e  $a < b < c < d$ , a extensão dos predicados *fuzzy* embutidos na abordagem FT-FIS é dada por:

$$\begin{aligned}
 X(leftShoulder_{k1, k2, a, b}) &= \{x \in V(d_{real}) \mid \\
 leftShoulder_{k1, k2, a, b}(x) &= \begin{cases} 1, & \text{se } x \in [k1, a] \\ \frac{b-x}{b-a}, & \text{se } x \in (a, b) \\ 0, & \text{se } x \in [b, k2] \end{cases} \quad (5.5) \\
 & \};
 \end{aligned}$$

$$\begin{aligned}
 X(rightShoulder_{k1, k2, a, b}) &= \{x \in V(d_{real}) \mid \\
 rightShoulder_{k1, k2, a, b}(x) &= \begin{cases} 0, & \text{se } x \in [k1, a] \\ \frac{x-a}{b-a}, & \text{se } x \in (a, b) \\ 1, & \text{se } x \in [b, k2] \end{cases} \quad (5.6) \\
 & \};
 \end{aligned}$$



$$\begin{aligned}
X(\text{triangular}_{k1,k2,a,b,c}) &= \{x \in V(d_{real}) \mid \\
&\quad \text{triangular}_{k1,k2,a,b,c}(x) = \begin{cases} 0, & \text{se } x \in [k1, a] \\ \frac{x-a}{b-a}, & \text{se } x \in (a, b) \\ 1, & \text{se } x = b \\ \frac{c-x}{c-b}, & \text{se } x \in (b, c) \\ 0, & \text{se } x \in [c, k2] \end{cases} \\
&\quad \};
\end{aligned} \tag{5.7}$$

$$\begin{aligned}
X(\text{trapezoidal}_{k1,k2,a,b,c,d}) &= \{x \in V(d_{real}) \mid \\
&\quad \text{trapezoidal}_{k1,k2,a,b,c,d}(x) = \begin{cases} 0, & \text{se } x \in [k1, a] \\ \frac{x-a}{b-a}, & \text{se } x \in (a, b) \\ 1, & \text{se } x \in [b, c] \\ \frac{d-x}{d-c}, & \text{se } x \in (c, d) \\ 0, & \text{se } x \in [d, k2] \end{cases} \\
&\quad \}.
\end{aligned} \tag{5.8}$$

A extensão da negação dos predicados *fuzzy* unários também é definida pela operação de complemento *fuzzy* padrão. Como um exemplo de uso dos predicados *fuzzy* unários, pode-se definir um predicado personalizado *pesoBaixo* com extensão

$$X(\text{pesoBaixo}) = \{x \in V(d_{real}) \mid \text{pesoBaixo}(x) = \text{leftShoulder}_{0,30,5,10}(x)\},$$

onde  $[0, 30]$  é o domínio do predicado *pesoBaixo* com  $a = 5$  e  $b = 10$  como parâmetros da função de pertinência. Descrições de conceito como  $\text{Item} \sqcap \exists \text{peso.pesoBaixo}$  representam itens que tenham peso baixo, cujo grau de pertinência é obtido com base na extensão do predicado *fuzzy* embutido *leftShoulder*.

Definido no âmbito da abordagem FT-FIS, o predicado *fuzzyRuleReasoning*, contribuição deste doutorado, atua como uma interface entre o conhecimento representado na ontologia *fuzzy* e as inferências de um SIF que trata regras *fuzzy* e variáveis linguísticas. É um predicado  $n$ -ário, contendo  $n1$  variáveis de entrada  $vi_1, \dots, vi_{n1}$  e  $n2$  variáveis de saída  $vo_1, \dots, vo_{n2}$ ,  $n1 + n2 = n$ , referentes às entradas e saídas de um SIF. A aridade mínima  $a_{min}(\text{fuzzyRuleReasoning}) = 2$  para regras *fuzzy* contendo apenas uma variável de entrada e uma de saída. O domínio do predicado *fuzzyRuleReasoning* é definido sobre o conjunto de todos os conjuntos *fuzzy* que

podem ser definidos no domínio dos números reais, denotado como  $\mathfrak{S}(d_{real})$ , onde  $f : V(d_{real}) \rightarrow [0, 1], \forall f \in \mathfrak{S}(d_{real})$ . Assim,  $dom(fuzzyRuleReasoning) = \mathfrak{S}(d_{real})^n$ , de modo que as variáveis utilizadas nesse predicado assumem valores que são conjuntos *fuzzy*. Caso os valores dessas variáveis correspondam a valores numéricos, os mesmos serão tratados como conjuntos *fuzzy* do tipo *singleton*. A extensão de *fuzzyRuleReasoning* é definida como:

$$\begin{aligned} X(fuzzyRuleReasoning) = \{ \langle vi_1, \dots, vi_{n1}; vo_1, \dots, vo_{n2} \rangle \in \mathfrak{S}(d_{real})^n \mid \\ \langle vo_1, \dots, vo_{n2} \rangle = FIS(vi_1, \dots, vi_{n1}), n1 + n2 = n \} \end{aligned} \quad (5.9)$$

A função  $FIS(vi_1, \dots, vi_{n1})$  presente na extensão do predicado *fuzzyRuleReasoning* indica que um SIF deve ser acionado para as variáveis de entrada  $vi_1, \dots, vi_{n1}$  retornando como saídas do raciocínio baseado em regras *fuzzy* os valores das variáveis  $\langle vo_1, \dots, vo_{n2} \rangle$ . Diferentemente dos demais predicados embutidos de FT-FIS, o predicado *fuzzyRuleReasoning* produz os valores das variáveis  $vo_1, \dots, vo_{n2}$  em função dos valores de  $vi_1, \dots, vi_{n1}$ , representados por conjuntos *fuzzy*. Por esse motivo, não faz sentido aplicar a negação sobre esse predicado, pois seu objetivo é gerar valores de propriedades concretas numéricas. Como *fuzzyRuleReasoning* foi definido para atender aos requisitos de modelagem descritos na Seção 5.2, é apropriado para situações em que valores de propriedades concretas numéricas são derivados da combinação de valores de outras propriedades numéricas por meio de regras *fuzzy*. Para ilustrar o uso do predicado *fuzzyRuleReasoning*, remetendo ao exemplo relativo a uma loja de comércio eletrônico descrito na Seção 5.2, pode-se definir um predicado personalizado *freteAlto* de aridade 4:

$$\begin{aligned} X(freteAlto) = \{ \langle v_0, v_1, v_2, v_3 \rangle \in \mathfrak{S}(d_{real})^4 \mid \\ fuzzyRuleReasoning(v_0, v_1, v_2; v_3) \wedge (v_3 \wedge rightShoulder_{0,400,200,300}) \} \end{aligned} \quad (5.10)$$

Nesse caso, a extensão do predicado personalizado *freteAlto* é definida a partir de uma expressão de predicados de tipo de dado *fuzzy*: uma conjunção entre o predicado *fuzzyRuleReasoning* e o predicado *fuzzy* unário *rightShoulder*. Utiliza-se uma conjunção porque a definição de *freteAlto* adiciona uma restrição ao valor de frete inferido por meio das regras *fuzzy*, para que tenha interseção com o predicado *fuzzy rightShoulder*. A ligação entre os dois predicados é a variável  $v_3$ , cujo valor resulta do raciocínio de SIF, sendo as entradas representadas pelas variáveis  $v_0, v_1, v_2$  (peso, volume, distância). Uma vez definido, esse predicado personalizado pode ser utilizado em definições de conceitos na ontologia *fuzzy*, como no caso do conceito *BonusProximaCompra*:

$$\begin{aligned} BonusProximaCompra = Cliente \sqcap \\ \exists possuiPedido.(\exists peso, volume, distancia, precoFrete.freteAlto) \end{aligned} \quad (5.11)$$

Conforme ilustrado no exemplo, o predicado embutido *fuzzyRuleReasoning* pode ser aplicado em descrições de conceito para associar o conhecimento modelado na ontologia *fuzzy* ao raciocínio baseado em regras *fuzzy*, realizado por um SIF. O procedimento de integração para acionar um SIF a partir de um algoritmo de raciocínio baseado em *tableau fuzzy* é descrito posteriormente na Seção 5.4.3.

### **Regras *fuzzy***

Conforme descrito no Capítulo 2, as regras *fuzzy* são capazes de expressar conhecimento com base em variáveis linguísticas e operações *fuzzy*, capturando a imprecisão presente na linguagem humana e em alguns domínios de aplicação.

Em FT-FIS, as regras *fuzzy* possuem um papel importante para tratar a imprecisão presente na inferência de valores de propriedades concretas numéricas que são consideradas em descrições e definições de conceitos na ontologia *fuzzy*. O contexto de integração apresentado na Seção 5.2 destaca as situações em que as regras *fuzzy* complementam o conhecimento representado em ontologias *fuzzy*, contribuindo para a aumentar a expressividade da representação de domínios que envolvem imprecisão.

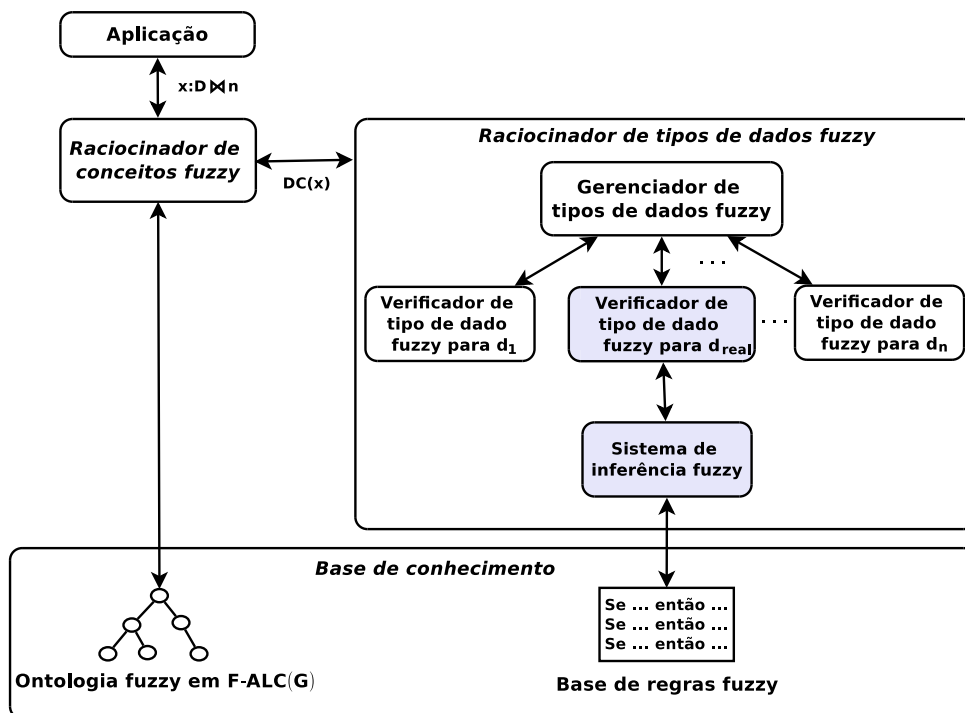
Em termos de sintaxe, as regras *fuzzy* consideradas na abordagem FT-FIS seguem o formato descrito na Equação 2.23 do Capítulo 2, em que cada regra é composta por conjunções de variáveis linguísticas no antecedente e uma variável linguística no consequente. Caso haja mais de uma variável linguística no consequente, as diferentes variáveis são tratadas de forma independente, configurando uma disjunção equivalente a um conjunto de regras com as mesmas variáveis no antecedente e uma variável linguística distinta no consequente.

Para manter a consistência do vocabulário e das definições presentes tanto na ontologia *fuzzy* quanto nas regras *fuzzy*, as variáveis linguísticas consideradas nas regras *fuzzy* e as propriedades concretas numéricas correspondentes na ontologia *fuzzy* devem ser definidas com a mesma nomenclatura e universo de discurso. Além disso, a partição *fuzzy* das variáveis linguísticas e dos predicados *fuzzy* definidos na ontologia *fuzzy* deve ser equivalente com relação à nomenclatura e às funções de pertinência utilizadas.

Com relação à semântica das regras *fuzzy*, FT-FIS pode dar suporte a qualquer método de raciocínio baseado em regras *fuzzy*, desde que seja instanciada a implementação de SIF correspondente. No contexto desse trabalho, foram explorados os métodos de Mamdani e Larsen, descritos na Seção 2.5.3 do Capítulo 2.

### 5.4.2 Arquitetura de Raciocínio de FT-FIS

Segundo Wang et al. (2008), o raciocínio de uma Lógica de Descrição *fuzzy* e de um grupo de tipos de dados *fuzzy*  $G$  podem ser tratados separadamente, pois lidam com domínios disjuntos: domínio  $\Delta^I$  de indivíduos abstratos e domínio de indivíduos concretos  $\Delta^D$  relativos a tipos de dados, respectivamente. Essa premissa possibilita o desenvolvimento de uma arquitetura de raciocínio composta por um *raciocinador de conceitos fuzzy*, responsável pelo processamento de construtores de conceito de uma *fuzzy* DL; e por um *raciocinador de tipos de dados fuzzy*, que trata expressões de tipos de dados *fuzzy*. Neste sentido, a abordagem FT-FIS estende a arquitetura de raciocínio definida por Wang et al. (2008), descrita na Seção 3.5.4 do Capítulo 3, ao adicionar um sistema de inferência *fuzzy* como componente do raciocinador de tipos de dados *fuzzy*. A Figura 5.2, que estende a figura proposta por Wang e Ma (2008), apresenta os principais componentes da arquitetura de raciocínio de FT-FIS e como eles estão interconectados.



**Figura 5.2:** Arquitetura de raciocínio de FT-FIS, no qual módulos em cinza representam o foco da extensão de FT-FIS para integrar o raciocínio de regras *fuzzy*.

O procedimento de raciocínio é iniciado pelo raciocinador de conceitos *fuzzy*, responsável por analisar uma assertão *fuzzy*  $\langle x : D \approx n \rangle$ , fornecida pela aplicação, juntamente com a ontologia *fuzzy* em  $F\text{-}\mathcal{ALC}(G)$ . O raciocinador de conceitos *fuzzy* baseia-se no método de raciocínio de uma *fuzzy* DL combinada com um grupo de tipos de dados *fuzzy*. Como a *fuzzy* DL considerada por FT-FIS é  $F\text{-}\mathcal{ALC}(G)$ , o raciocinador de conceitos *fuzzy* segue o algoritmo de raciocínio dessa *fuzzy* DL, proposto por Wang e Ma (2008). Assim, o raciocinador de conceitos *fuzzy* deve

ser capaz de tratar os construtores de conceito dessa *fuzzy* DL e realizar as devidas expansões no caso de conceitos complexos.

No entanto, o raciocinador de conceitos *fuzzy* não é capaz de interpretar e processar expressões de tipos de dados *fuzzy*, pois somente trata a parte abstrata do conhecimento representado em ontologias *fuzzy*. Assim, toda vez que houver um construtor de conceito contendo alguma expressão de tipos de dados *fuzzy*  $E$  associada a um indivíduo  $x$ ,  $E$  é armazenada em um conjunto  $DC(x)$ , que atua como interface entre o raciocinador de conceitos *fuzzy* e o raciocinador de tipos de dados *fuzzy*. Então, o raciocinador de tipos de dados *fuzzy* é acionado para checar a satisfazibilidade de  $DC(x)$ . Segundo Wang e Ma (2008), essa arquitetura possui a vantagem de ser flexível e genérica: quando novos tipos de dados *fuzzy* e novos predicados *fuzzy* são necessários, somente o raciocinador de tipos de dados *fuzzy* deve ser estendido. Essa característica é importante para a abordagem FT-FIS, pois somente o raciocinador de tipos de dados *fuzzy* deve ser estendido, sem haver necessidade de modificar o raciocinador de conceitos *fuzzy*, para a definição do predicado *fuzzyRuleReasoning*.

O papel do raciocinador de tipos de dados *fuzzy* é verificar a satisfazibilidade das expressões de tipos de dados *fuzzy* de acordo com o significado das extensões dos predicados de tipos de dados *fuzzy* embutidos, descritos na Seção 5.4.1. Essa verificação é realizada para cada  $DC(x)$  associado a um indivíduo abstrato  $x$  da ontologia *fuzzy*. Conforme ilustrado na Figura 5.2, o raciocinador de tipos de dados *fuzzy* é composto por um *gerenciador de tipos de dados fuzzy* e por *verificadores de tipos de dados fuzzy*.

O gerenciador de tipos de dados *fuzzy* decompõe as expressões de tipos de dados *fuzzy*  $E$ , presentes em  $DC(x)$ , em várias conjunções de predicados de tipos de dados *fuzzy* embutidos, por tipo de dado base  $d \in D_G$ . A decomposição é necessária porque cada verificador de tipo de dado *fuzzy* é responsável por verificar a satisfazibilidade de conjunções de predicados referentes a um tipo de dado base específico. Os predicados decompostos são submetidos aos verificadores de tipos de dados *fuzzy* correspondentes, que retornam se são satisfáveis ou não. O gerenciador de tipos de dados *fuzzy* repassa o resultado obtido pelos verificadores ao raciocinador de conceitos *fuzzy*, para determinar se o conteúdo de  $DC(x)$  é satisfável ou não.

Na Figura 5.2, são ilustrados  $n + 1$  exemplos de verificadores de tipos de dados *fuzzy*, onde  $d_1, \dots, d_n$  representam tipos de dados base distintos e  $d_{real}$  refere-se ao tipo de dado dos números reais. Para a abordagem FT-FIS, no mínimo deve ser definido o verificador de tipos de dados *fuzzy* para o domínio dos números reais, em função dos predicados básicos oferecidos por FT-FIS: predicados *crisp* unários, predicados *fuzzy* e o predicado *fuzzyRuleReasoning* (Seção 5.4.1). Cada verificador de tipo de dado *fuzzy* é capaz de interpretar e processar os

predicados embutidos definidos sobre o seu tipo de dado  $d$  correspondente, verificando se a extensão dos predicados e os valores concretos recebidos do gerenciador de tipos de dados *fuzzy* são satisfatíveis, ou seja, não são contraditórios.

Em especial, o verificador de tipo de dado *fuzzy* do domínio dos números reais aciona um sistema de inferência *fuzzy* toda vez que o predicado *fuzzyRuleReasoning* ocorrer em conjunções de predicados. Essa extensão, definida na abordagem FT-FIS, permite confiar o processamento de regras *fuzzy* a um sistema de inferência *fuzzy*, que é especializado para tratar o raciocínio baseado em regras *fuzzy*. A partir das variáveis de entrada vindas da ontologia *fuzzy*, o SIF gera as saídas correspondentes e as retorna para o verificador de tipos de dados *fuzzy* que, então, verifica a satisfazibilidade juntamente com os demais predicados da conjunção. Desta forma, FT-FIS integra um SIF à arquitetura de raciocínio definida por Wang et al. (2008), sendo acionado somente quando necessário, sem modificar as demais tarefas de raciocínio de  $F\text{-}\mathcal{ALC}(G)$ .

A próxima seção explica mais detalhadamente como os componentes da arquitetura são integrados, com foco no processamento de predicados de tipos de dados *fuzzy* e a integração do raciocínio de SIF por meio do predicado *fuzzyRuleReasoning*.

### 5.4.3 Abordagem de Integração

Conforme descrito na Seção 3.5.2 do Capítulo 3, as tarefas de raciocínio básicas de *fuzzy* DL incluem a verificação de satisfazibilidade de conceitos, consistência da base de conhecimento, verificação de instâncias e melhor limite de grau de verdade. Como demonstrado por Straccia (2001), essas tarefas de raciocínio podem ser reduzidas para a verificação de consistência da base de conhecimento, geralmente realizada por meio de algoritmos baseados em *tableau*, que tentam construir um modelo ou interpretação para uma base de conhecimento *fuzzy*.

Neste sentido, o algoritmo baseado em *tableau fuzzy* para  $F\text{-}\mathcal{ALC}(G)$ , proposto por Wang e Ma (2008), foi considerado para a abordagem de integração de FT-FIS, por possibilitar a definição de novos predicados *fuzzy* e a interface com um sistema de inferência *fuzzy* de forma modular, sem interferir nas regras de expansão que tratam o conhecimento abstrato da ontologia *fuzzy*. Assim, o foco da descrição da abordagem de integração realizada por FT-FIS é voltado para regras de expansão do *tableau fuzzy* e para os procedimentos que envolvem predicados de tipos de dados *fuzzy*. As definições básicas sobre o raciocínio baseado em *tableau fuzzy* envolvendo a parte abstrata de ontologias *fuzzy* são apresentadas na Seção 3.5.3 do Capítulo 3. A Seção 3.5.4 também contém as definições acerca da simbologia e da nomenclatura utilizada para a explicação do funcionamento do algoritmo baseado em *tableau fuzzy*.

Em geral, como ocorre nos raciocinadores de ontologias baseados em *tableau*, a consulta da aplicação é reduzida para a verificação de consistência de bases de conhecimento *fuzzy* (Seção 3.5.2 do Capítulo 3). Desta forma, o algoritmo baseado em *tableau fuzzy* trabalha com uma base de conhecimento  $\mathcal{K}' = \mathcal{K} \cup \{\langle x : D \bowtie n \rangle\}$ , onde  $\mathcal{K}$  é a base de conhecimento da ontologia *fuzzy* e  $\{\langle x : D \bowtie n \rangle\}$  é a asserção *fuzzy* resultante da redução da tarefa de raciocínio requisitada pela aplicação. A partir de  $\mathcal{K}'$ , o algoritmo baseado em *tableau fuzzy* de  $F\text{-}\mathcal{ALC}(G)$ , implementado pelo raciocinador de conceitos *fuzzy*, tenta construir um modelo que mostre que  $\mathcal{K}'$  possui pelo menos uma interpretação na qual seus componentes TBox e ABox são consistentes. Como pré-requisito, a descrição de conceito  $D$  deve estar na forma normal negativa, inclusive eventuais expressões de tipos de dados *fuzzy*, que pode ser obtida por meio da aplicação das leis de De Morgan (STRACCIA, 2001).

Conforme definido por Horrocks (2003), um *tableau* é um grafo dirigido que representa um modelo de  $\mathcal{K}'$ . No caso de um *tableau fuzzy* com suporte a grupos de tipos de dados  $G$ , nós abstratos correspondem a indivíduos abstratos  $x \in \Delta^{\mathcal{I}}$  e nós de tipos de dados representam valores concretos  $v \in \Delta_D$ . Nós de tipos de dados também são chamados de nós-folha não rotulados, pois não possuem rótulos e somente atuam como valores de propriedades concretas.

Cada nó abstrato  $x$  possui um conjunto de rótulos  $\mathcal{L}(x) = \{\langle C \bowtie k \rangle\}$ , onde  $C$  é uma descrição de conceito, possivelmente complexa. Cada aresta  $\langle x, y \rangle$  também possui um conjunto de rótulos  $\mathcal{L}(\langle x, y \rangle) = \{\langle R \bowtie k \rangle\}$ , onde cada rótulo representa uma asserção de relacionamento  $\langle R(x, y) \bowtie k \rangle$  entre dois nós abstratos  $x$  e  $y$ . Neste caso, o nó  $y$  é chamado de sucessor- $R$  de  $x$ . Cada aresta  $\langle x, v \rangle$  possui um conjunto de rótulos  $\mathcal{L}(\langle x, v \rangle) = \{\langle T \bowtie k \rangle\}$  para cada asserção  $\langle T(x, v) \bowtie k \rangle$ . Como FT-FIS considera propriedades concretas funcionais, deve existir no máximo um rótulo  $\langle T \bowtie k \rangle \in \mathcal{L}(\langle x, v \rangle)$ . Um nó de tipo de dado  $v$  é um sucessor- $T$  de um nó abstrato  $x$  se, para alguma propriedade concreta  $T$ ,  $T \in \mathcal{L}(\langle x, v \rangle)$ . Maiores detalhes sobre as definições de nós abstratos, nós de tipos de dados, sucessores- $T$  e rótulos são apresentados na Seção 3.5.4 do Capítulo 3.

Antes de inicializar o *tableau fuzzy*, o raciocinador de conceitos *fuzzy* aplica a técnica de *unfolding* (HORROCKS, 2003; STRACCIA, 2001; STOILLOS et al., 2007), conforme descrito na Seção 3.3.3 do Capítulo 3, visto que FT-FIS requer que a TBox seja definitiva. O raciocinador de conceitos *fuzzy* inicializa um *tableau fuzzy* para as asserções *fuzzy* presentes na ABox  $\mathcal{A}'$  de  $\mathcal{K}'$ , de modo que:

- Exista um nó abstrato  $x_i$  para cada indivíduo  $a_i$  de  $\mathcal{A}'$ , rotulado com um conjunto de rótulos  $\mathcal{L}(x_i) = \{\langle C \bowtie k \rangle\}$  para cada asserção  $\langle C(a_i) \bowtie k \rangle$  em  $\mathcal{A}'$ ;  $DC(x_i)$  é inicializado como um conjunto vazio;

- Exista uma aresta  $\langle x_i, x_j \rangle$ , rotulada com o conjunto  $\mathcal{L}(\langle x_i, x_j \rangle) = \{\langle R \bowtie k \rangle\}$  para cada asserção  $\langle R(a_i, a_j) \bowtie k \rangle$  in  $\mathcal{A}'$ ;
- Exista um nó de tipo de dado  $v_{ij}$  para cada indivíduo concreto  $v_{ij}$  presente em asserções  $\langle T_j(a_i, v_{ij}) \bowtie k \rangle$  de  $\mathcal{A}'$ ;
- Exista uma aresta  $\langle x_i, v_{ij} \rangle$ , rotulada com o conjunto  $\mathcal{L}(\langle x_i, v_{ij} \rangle) = \{\langle T_j \bowtie k \rangle\}$  para cada asserção  $\langle T_j(a_i, v_{ij}) \bowtie k \rangle$  de  $\mathcal{A}'$ .

Depois da inicialização, as descrições de conceito  $C$  presentes em rótulos de nós abstratos são decompostas com base em regras de expansão, para obter rótulos envolvendo apenas conceitos primitivos, respeitando a semântica dos construtores de conceito da *fuzzy* DL e as restrições sobre o grau de verdade das asserções *fuzzy*. Assim, são aplicadas as regras de expansão descritas na Tabela 3.7 do Capítulo 3, que tratam os construtores de conceito de *fuzzy*  $\mathcal{ALC}$  referentes à parte abstrata do conhecimento representado em  $\mathcal{K}'$ . As regras de expansão adicionam novos nós ou novos rótulos a nós existentes conforme a semântica dos construtores de conceito, até que nenhuma regra seja aplicável ou então tenha ocorrido alguma contradição (*clash*) nos rótulos de um nó, que impeça a definição de um modelo. A principal condição de *clash* para a parte abstrata do conhecimento envolvendo os elementos de  $\Delta^{\mathcal{I}}$  corresponde à ocorrência de pares conjugados em rótulos de um mesmo nó abstrato. As condições de *clash* de  $F\text{-}\mathcal{ALC}(G)$  são apresentadas com mais detalhes nas Seções 3.5.3 e 3.5.4.

Quando houver algum construtor de conceito contendo expressões de tipos de dados *fuzzy*  $E$ , o raciocinador de conceitos *fuzzy* aplica as regras- $G$ , descritas na Tabela 3.9 da Seção 3.5.4. Essas regras podem adicionar novos nós de tipos de dados e arestas correspondentes a propriedades concretas ou novos rótulos a arestas existentes. Em FT-FIS, a ordem de aplicação das regras de expansão para um nó abstrato  $x$ , incluindo as regras- $G$ , seguem algumas recomendações de (BAADER; NUTT, 2003; PAN, 2007):

1. Regras que adicionam rótulos ao próprio nó abstrato  $x$ : regra- $\neg$ , regra- $\Box_{\triangleright}$ , regra- $\Box_{\triangleleft}$ . Elas aumentam a probabilidade de ocorrência de um *clash* no próprio nó, sem a necessidade de expandir o restante do *tableau fuzzy*;
2. Regras não determinísticas que adicionam rótulos ao próprio nó abstrato  $x$ : regra- $\Box_{\triangleright}$  e regra- $\Box_{\triangleleft}$ . Segundo Baader e Nutt (2003), as regras são consideradas não determinísticas no sentido de que a ABox de  $\mathcal{K}'$  é finitamente transformada em diversas ABoxes de modo que a ABox original é consistente se alguma das ABoxes geradas também é. Como essas regras também adicionam rótulos ao próprio nó, também aumentam a probabilidade da ocorrência de um *clash* no próprio nó;



3. Regras que adicionam sucessores abstratos diretos: regra- $\exists_{\triangleright}$  e regra- $\forall_{\triangleleft}$ . São regras que produzem novos nós abstratos no *tableau fuzzy*, mas que não geram situações de *clash* de forma direta;
4. Regras que adicionam rótulos em sucessores abstratos diretos: regra- $\exists_{\triangleleft}$  e regra- $\forall_{\triangleright}$ . Elas adicionam novos rótulos aos sucessores abstratos, possivelmente gerando pares conjugados e, portanto, levando à ocorrência de *clash*. Como elas tratam os sucessores, devem ser aplicadas sempre que novos sucessores são adicionados ao *tableau fuzzy*;
5. Regras que adicionam sucessores de tipos de dados *fuzzy* diretos: regra- $\exists_{p\triangleright}$  e regra- $\forall_{p\triangleleft}$ . São regras que produzem novos nós de tipos de dados no *tableau fuzzy*, caso não existam previamente, e adicionam elementos a  $DC(x)$ ;
6. Regras que adicionam elementos a  $DC(x)$  referentes a sucessores de tipo de dado *fuzzy* existentes: regra- $\exists_{p\triangleleft}$  e regra- $\forall_{p\triangleright}$ . Elas adicionam elementos a  $DC(x)$ , possivelmente gerando pares conjugados envolvendo expressões de predicados de tipos de dados *fuzzy*. Como elas tratam sucessores de tipos de dados *fuzzy* existentes, devem ser aplicadas sempre que novos sucessores desse tipo são adicionados ao *tableau fuzzy*.

FT-FIS prioriza a aplicação das regras que tratam a parte abstrata do conhecimento para um nó abstrato  $x$  para que as situações de *clash* envolvendo seus rótulos sejam identificadas logo que possível. Assim, as regras- $G$  possuem prioridade menor, sendo acionadas caso não tenham ocorrido situações de *clash* nas regras aplicadas anteriormente. Consequentemente, evita-se o processamento desnecessário do conhecimento envolvendo grupos de tipos de dados *fuzzy*, no caso de *clash* nos rótulos dos nós abstratos. Os sucessores abstratos gerados são expandidos depois que todas as regras possíveis foram aplicadas ao nó abstrato  $x$  e foi verificada a satisfazibilidade de  $DC(x)$  pelo raciocinador de tipos de dados *fuzzy*. No caso das regras não determinísticas, cada possibilidade de expansão do *tableau fuzzy* é tratada por vez, até que seja obtido algum modelo sem *clash* ou até que todas as possibilidades tenham sido consideradas.

Em especial, as regras- $G$  armazenam em um conjunto  $DC(x)$  as expressões de tipos de dados *fuzzy*  $E$  que ocorrem em descrições de conceito que rotulam um indivíduo abstrato  $x$ . Assim, como o raciocinador de conceitos *fuzzy* não possui o conhecimento específico para decompor as expressões  $E$ , ele as armazena em  $DC(x)$  e invoca o raciocinador de tipos de dados *fuzzy* para verificar sua satisfazibilidade. Dentre os possíveis predicados presentes nas expressões  $E$ , o predicado *fuzzyRuleReasoning* pode ser utilizado para fazer referência ao raciocínio de SIF para inferir valores de propriedades concretas numéricas. Desta forma, conforme

mencionado anteriormente,  $DC(x)$  atua como uma interface entre o raciocinador de conceitos *fuzzy* e o raciocinador de tipos de dados *fuzzy*.

Como FT-FIS considera propriedades concretas funcionais, para cada nó abstrato  $x_i$  existe apenas um nó de tipo de dado, identificado por  $v_{ij}$ , ligado por uma propriedade concreta  $T_j$ . Os identificadores são importantes para realizar o mapeamento entre os nós de tipos de dados presentes no *fuzzy tableau* e os valores de propriedades concretas considerados pelo raciocinador de tipos de dados *fuzzy*. Em especial, para FT-FIS, os identificadores são relevantes para referenciar os valores dos nós de tipos de dados que correspondem a entradas e saídas do SIF.

Assim, as expressões de tipos de dados  $E$  referentes a sucessores- $\langle T_1, \dots, T_n \rangle$  de um nó abstrato  $x_i$  são armazenadas em  $DC(x_i)$ , que contém elementos da forma:

$$(\langle E(v_{i1}, \dots, v_{in}) \bowtie k \rangle) \quad (5.12)$$

onde  $E \in \mathbf{Dexp}(G)$ ,  $\bowtie \in \{<, \leq, >, \geq\}$ ,  $k \in [0, 1]$  e  $\langle v_{i1}, \dots, v_{in} \rangle$  são variáveis que correspondem a identificadores de nós de tipos de dados referentes a sucessores- $\langle T_1, \dots, T_n \rangle$  de um nó abstrato  $x_i$ . A presença de várias expressões  $E_j \in DC(x_i)$ , com  $1 \leq j \leq m$ ,  $m \geq 1$  equivale a uma conjunção de expressões, ou seja, todas  $E_j$  devem ser satisfatíveis para que  $DC(x_i)$  seja satisfatível. As expressões em  $DC(x_i)$  podem pertencer a tipos de dados base  $d \in D_G$  distintos, porém não devem compartilhar variáveis, uma vez que os domínios dos tipos de dados base são disjuntos conforme a definição de grupo de tipo de dado *fuzzy* (WANG; MA, 2008) (Seção 3.5.4 do Capítulo 3). O compartilhamento de variáveis significa que um mesmo nó de tipo de dado está sendo referenciado por expressões de tipos de dados distintos, o que configura uma inconsistência com relação à teoria de grupos de tipos de dados *fuzzy*  $G$ .

Depois da aplicação das regras de expansão para um nó abstrato  $x_i$  sem *clash*,  $DC(x_i)$  é enviado para o raciocinador de tipos de dados *fuzzy*, para verificar a sua satisfazibilidade. Se o raciocinador de tipos de dados *fuzzy* retornar que  $DC(x_i)$  é insatisfatível, o nó abstrato  $x_i$  contém um *clash* que, por ter sido obtido em função da verificação dos predicados de um grupo de tipos de dados *fuzzy*  $G$ , é denominado um *G-clash*.

Para verificar a satisfazibilidade de  $DC(x_i)$ , primeiramente o raciocinador de tipos de dados *fuzzy* aciona o gerenciador de tipos de dados *fuzzy*. Sua principal função é transformar as expressões presentes em  $DC(x_i)$  em disjunções de conjunções de predicados de um mesmo tipo de dado base, para que seja possível invocar os verificadores de tipos de dados *fuzzy* correspondentes. Além disso, ele verifica algumas condições de *clash* que sejam independentes da avaliação do significado dos predicados.

Assim, seja  $\psi$  uma conjunção de expressões de tipos de dados *fuzzy* representando o conteúdo de  $DC(x_i)$ , os passos realizados pelo gerenciador de tipos de dados *fuzzy* da abordagem FT-FIS, inspirados nos gerenciadores de tipos de dados propostos por Pan (2007), Wang e Ma (2008), são:

1. Decompor  $\psi$  em  $\psi_j$ , separando as expressões por tipo de dado base  $d_j \in D_G$ . Assim, cada  $\psi_j$  representa uma conjunção de expressões de tipos de dados cujos predicados são definidos sobre o domínio  $d_j$ ;
2. Transformar cada  $\psi_j$  para a forma normal disjuntiva  $\psi'_j$ ;
3. Verificar a satisfazibilidade de cada conjunção  $\mathcal{C}$  de  $\psi'_j$ :
  - (a) Dada uma restrição de predicado *fuzzy*  $\beta = \langle p(v_1, \dots, v_n) \bowtie k \rangle$ , onde  $v_1, \dots, v_n$  é uma tupla de variáveis e  $p$  um predicado de tipo de dado *fuzzy*; se ambos  $\beta$  e algum par conjugado  $\beta^C$  ocorrerem em  $\mathcal{C}$ , então  $\mathcal{C}$  é insatisfável;
  - (b) Caso contrário, o gerenciador de tipos de dados *fuzzy* aciona o verificador de tipos de dados *fuzzy* correspondente ao tipo de dado base  $d_j$  para verificar a satisfazibilidade de cada conjunção  $\mathcal{C}$  de  $\psi'_j$ . Se pelo menos alguma conjunção  $\mathcal{C}$  for satisfável, então  $\psi'_j$  é satisfável. Por fim,  $\psi$  é satisfável se todos  $\psi'_j$  forem satisfáveis. Neste caso, o gerenciador de tipos de dados *fuzzy* retorna que  $DC(x_i)$  é satisfável. Caso contrário, retorna que  $DC(x_i)$  é insatisfável.

Diferentemente da proposta de Wang e Ma (2008), na abordagem FT-FIS os predicados negados não são processados de forma genérica no gerenciador de tipos de dados *fuzzy*. Não é necessário decompor a negação sobre predicados de tipos de dados *fuzzy* porque FT-FIS determina que a negação seja aplicada somente a predicados atômicos, conforme descrito na Seção 5.4.1. Os predicados atômicos negados são, portanto, encaminhados para os verificadores de tipos de dados *fuzzy*, pois as negações combinadas a restrições de grau de verdade  $\bowtie k$  podem assumir significados específicos dependendo do tipo de dado base.

Ao ser acionado pelo gerenciador de tipos de dados *fuzzy*, o verificador de tipo de dado *fuzzy*  $DC_j$  é responsável por decidir a satisfazibilidade da conjunção  $\mathcal{C}$  de predicados *fuzzy* definidos sobre um tipo de dado base  $d_j \in D_G$ . A conjunção  $\mathcal{C}$  possui o seguinte formato (WANG; MA, 2008):

$$\mathcal{C} = \bigwedge_{t=1}^m \langle P_t(v_1^t, \dots, v_{n_t}^t) \bowtie k \rangle \quad (5.13)$$

onde  $v_1^t, \dots, v_{n_t}^t$  são variáveis correspondendo a identificadores de nós de tipos de dados e  $P_t$  representa um predicado atômico  $p$  ou um predicado atômico negado  $\bar{p}$ , definido sobre o tipo de dado base  $d_j$  com aridade  $n_t$ .

Para que os verificadores de tipos de dados *fuzzy* sejam corretamente acionados pelo gerenciador de tipos de dados *fuzzy*, os domínios dos tipos de dados base devem ser disjuntos entre si, conforme a teoria de grupos de tipos de dados *fuzzy*. Além disso, eles devem prover algumas informações ao gerenciador (PAN, 2007):

- sua função de verificação de satisfazibilidade;
- o nome do tipo de dado base  $d_j$  correspondente;
- o conjunto de nomes de predicados de tipos de dados *fuzzy* embutidos definidos para o seu tipo de dado base.

Conforme ilustrado na Figura 5.2, a arquitetura de raciocínio de FT-FIS pressupõe a definição do verificador de tipo de dado *fuzzy* do domínio dos números reais, denominado  $DC_{d_{real}}$ , pois os predicados embutidos na abordagem FT-FIS, incluindo o predicado *fuzzyRuleReasoning*, pertencem a esse domínio. A Tabela 5.2 mostra os predicados embutidos básicos para os quais  $DC_{d_{real}}$  deve oferecer suporte, no contexto da abordagem FT-FIS. A Seção 5.4.1 também descreve a extensão de cada predicado embutido.

Ao analisar cada elemento  $P_t(v_1^t, \dots, v_{n_t}^t) \bowtie k$  da conjunção  $\mathcal{C}$ ,  $DC_{d_{real}}$  atribui uma variável para cada  $v_q^t, q \in [1, n_t]$  para manter a correspondência correta com os identificadores de nós de tipos de dados *fuzzy* do *tableau fuzzy*. Se  $v_q^t$  é o identificador de um nó de tipo de dado *fuzzy* que é sucessor- $T$  de um nó abstrato  $x$ , então o valor de  $v_q^t$  é representado como um conjunto *fuzzy* definido no domínio  $[k1, k2]$  da propriedade concreta  $T$ .

Primeiramente,  $DC_{d_{real}}$  avalia os elementos contendo predicados unários  $P_t(v_1^t)$ , pois eles atribuem valores a  $v_1^t$  em função das restrições de grau do tipo  $\bowtie k$ . Como  $v_1^t$  pode, eventualmente, ser uma variável de entrada do predicado *fuzzyRuleReasoning*, seu valor deve ser definido antes que o SIF seja acionado. Para realizar a restrição de grau de verdade  $\bowtie k$  a um conjunto *fuzzy*  $FS$  definido em  $[k1, k2]$ , a abordagem FT-FIS define a função  $DR(FS, \bowtie, k)$ , descrita na Tabela 5.3, que retorna um conjunto *fuzzy*  $FS'$ . Assim, o valor de  $v_1^t$  é definido pela aplicação da função  $DR(X(P_t), \bowtie, k)$ .

Caso uma variável  $v_q^t$  já tenha sido inicializada por algum predicado  $P_{t-1}$  analisado anteriormente, seu valor é combinado por conjunção *fuzzy* com o resultado de  $DR(X(P_t), \bowtie, k)$ .

**Tabela 5.3: Restrições de grau aplicadas sobre um conjunto *fuzzy*  $FS$  em FT-FIS.**

$\bowtie$	$k$	$FS' = DR(FS, \bowtie, k)$
$\geq$	0	$X(\text{crispInterval}_{k1,k2,k1,k2})$
$\geq$	(0,1]	$x \in V(d_{real}) \mid FS'(x) = \begin{cases} FS(x), & \text{se } FS(x) \geq k \\ 0, & \text{caso contrário} \end{cases}$
$>$	[0,1)	$x \in V(d_{real}) \mid FS'(x) = \begin{cases} FS(x), & \text{se } FS(x) > k \\ 0, & \text{caso contrário} \end{cases}$
$\leq$	[0,1]	$DR(\neg FS, \geq, 1 - k)$
$<$	(0,1]	$DR(\neg FS, >, 1 - k)$

Seguindo a semântica da conjunção de  $F\text{-}\mathcal{ALC}(G)$  (Seção 5.4.1), a operação usada é a interseção *fuzzy* padrão definida pelo mínimo.

Depois que os predicados embutidos unários são processados,  $DC_{d_{real}}$  analisa os elementos da conjunção  $\mathcal{C}$  que contêm o predicado  $fuzzyRuleReasoning_t(vi_1^t, \dots, vi_{n_1}^t; vo_1^t, \dots, vo_{n_2}^t)$ . As entradas a serem enviadas ao SIF são obtidas em função dos identificadores das variáveis de entrada  $vi_1^t, \dots, vi_{n_1}^t$ , que identificam os nós de tipos de dados *fuzzy* do *tableau fuzzy*. Assim, os valores de entrada são obtidos das variáveis definidas por  $DC_{d_{real}}$  durante o processamento dos predicados embutidos unários, desde que tenham os mesmos identificadores de nós de tipos de dados *fuzzy*. Caso o valor de alguma variável de entrada não tenha sido definido previamente por  $DC_{d_{real}}$ , seu valor é considerado como  $X(\text{crispInterval}_{k1,k2,k1,k2})$ , sendo que  $[k1, k2]$  se refere ao domínio da propriedade concreta  $T$  referente à variável de entrada em questão. Essa estratégia é adotada em FT-FIS em virtude da suposição de mundo aberto considerada por ontologias: se o valor de uma propriedade concreta é necessário para realizar alguma tarefa de inferência, mas não foi definido, considera-se todos os valores possíveis, ou seja, o próprio universo de discurso  $[k1, k2]$ .

Uma vez definidos os valores de todas as variáveis de entrada  $vi_1^t, \dots, vi_{n_1}^t$ ,  $DC_{d_{real}}$  delega o processamento do predicado  $fuzzyRuleReasoning$  para o componente da arquitetura de raciocínio especializado para essa função: o sistema de inferência *fuzzy*. Assim, o SIF é acionado para inferir os valores das saídas por meio de raciocínio aproximado baseado em regras *fuzzy*. O método de inferência utilizado no SIF é especificado juntamente com a base de regras, conforme descrito na Seção 5.4.1. Dentre os métodos geralmente utilizados nas aplicações, estão o método de Mamdani e Larsen, porém outros métodos podem ser considerados, desde que a implementação do SIF ofereça o suporte apropriado. Os pré-requisitos para o SIF, no contexto da abordagem FT-FIS, são:

- Suporte para entradas *fuzzy* e saídas *fuzzy*, pois os valores das variáveis em  $DC_{d_{real}}$  correspondem a conjuntos *fuzzy*;

- Dados os valores de entrada  $vi_1^t, \dots, vi_{n1}^t$ , o SIF deve sempre retornar valores de saída  $o_1, \dots, o_{n2}$ , representados como conjuntos *fuzzy*. Caso nenhuma regra tenha sido disparada, deve-se especificar na base de regras *fuzzy* valores-padrão das saídas, possivelmente representados como conjuntos *fuzzy* do tipo *singleton*. Assim, garante-se que o processamento das regras sempre retorna alguma resposta, em tempo finito;
- Os métodos de defuzificação podem ser acionados por  $DC_{d_{real}}$  mesmo depois da geração das saídas do SIF, uma vez que as saídas *fuzzy* retornadas pelo SIF podem ser combinadas com valores de variáveis previamente definidos por outros predicados embutidos tratados por  $DC_{d_{real}}$ . Desta forma, a defuzificação pode ser acionada por  $DC_{d_{real}}$  depois que todos os predicados da conjunção  $\mathcal{C}$  foram avaliados.

Após o processamento das regras *fuzzy* e entradas  $vi_1^t, \dots, vi_{n1}^t$ , o SIF retorna conjuntos *fuzzy* que representam as saídas inferidas  $o_1, \dots, o_{n2}$  ao  $DC_{d_{real}}$ . Para realizar a correspondência com as variáveis  $vo_1^t, \dots, vo_{n2}^t$  do predicado *fuzzyRuleReasoning*,  $DC_{d_{real}}$  deve conhecer a sequência das saídas e as variáveis linguísticas processadas pelo SIF. Para realizar o mapeamento correto, os nomes e os universos de discurso das variáveis linguísticas utilizadas nas regras e das propriedades concretas modeladas na ontologia *fuzzy* devem ser os mesmos, como apresentado na Seção 5.4.1. Para cada saída  $o_h, h \in [1, n2]$ , representada como um conjunto *fuzzy*  $FS o_h$ ,  $DC_{d_{real}}$  aplica a função  $DR(FS o_h, \bowtie, k)$  para realizar a restrição de grau presente no predicado *fuzzyRuleReasoning* $_t(vi_1^t, \dots, vi_{n1}^t; vo_1^t, \dots, vo_{n2}^t) \bowtie k$ . Por fim,  $DC_{d_{real}}$  atribui o conjunto *fuzzy* resultante  $FS o'_h$  à variável  $vo_h^t$  correspondente, realizando uma conjunção *fuzzy* caso algum valor tenha sido atribuído anteriormente.

Durante o processamento dos predicados de tipos de dados *fuzzy*, podem ocorrer situações que configuram contradições, chamadas de *G-clash*. As situações de *G-clash* consideradas por  $DC_{d_{real}}$ , no contexto da abordagem FT-FIS, são:

1.  $\langle P_t(v_1^t, \dots, v_{n_t}^t) \bowtie k \rangle$ , onde  $k \notin [0, 1]$ ;
2.  $\langle P_t(v_1^t, \dots, v_{n_t}^t) > 1 \rangle$ ;
3.  $\langle P_t(v_1^t, \dots, v_{n_t}^t) < 0 \rangle$ ;
4.  $altura(FS(x)) \leq k$ , quando  $DR(FS, >, k)$ ;
5.  $altura(FS(x)) < k$ , quando  $DR(FS, \geq, k)$ ;
6.  $min(FS_1, FS_2)(x) = 0, \forall x \in V(d_{real})$ .

As situações de *G-clash* de 1 a 3 referem-se a restrições de grau fora do intervalo  $[0, 1]$ . As situações 4 e 5 correspondem à aplicação de restrição de grau do tipo  $\triangleright k$ , onde  $\triangleright = \{>, \geq\}$ , sobre um conjunto *fuzzy*  $FS$  cuja altura é menor que  $k$ . Por fim, um *G-clash* ocorre quando a conjunção de dois conjuntos *fuzzy*  $FS_1$  e  $FS_2$  é vazia (situação 6), onde  $FS_1$  corresponde ao valor de uma variável  $v$  proveniente do processamento de um predicado  $P_{t-1}$  e  $FS_2$  o valor resultante da avaliação de um predicado  $P_t$ . Note que as situações 4 a 6 podem ocorrer depois do acionamento do SIF, quando são aplicadas restrições de graus sobre as saídas do SIF (situações 4 e 5) e quando as saídas são atribuídas às respectivas variáveis em  $DC_{d_{real}}$  (situação 6).

A ocorrência de *G-clash* ao processar algum predicado  $P_t \in \mathcal{C}$  implica que  $\mathcal{C}$  é insatisfável. Caso contrário, quando todos predicados  $P_t \in \mathcal{C}$  foram processados por  $DC_{d_{real}}$  sem *G-clash*,  $\mathcal{C}$  é satisfável. Neste caso,  $DC_{d_{real}}$  também retorna para o gerenciador de tipos de dados *fuzzy* os valores de cada variável correspondente aos nós de tipos de dados *fuzzy* do *tableau fuzzy* sobre o domínio dos números reais. Assim, esses valores são disponibilizados para o raciocinador de conceitos *fuzzy* e podem ser defuzificados, caso necessário. A defuzificação é realizada pelo SIF, sendo opcional quando a aplicação invoca o raciocinador de conceitos *fuzzy* para verificação de consistência de  $\mathcal{K}'$ . Por esta razão, o SIF deve ser capaz de realizar a defuzificação separadamente do processo de inferência baseado em regras *fuzzy*, conforme mencionado anteriormente como um dos requisitos para o SIF.

Por fim, o processo de raciocínio finaliza quando o *tableau fuzzy* é completo, ou seja, nenhuma das regras de expansão, sejam regras de  $F\text{-}\mathcal{ALC}(G)$  ou regras- $G$ , podem ser aplicadas; ou quando existe algum nó abstrato  $x$  contendo um *clash*, incluindo as situações de *G-clash*. Se as regras de expansão podem ser aplicadas de modo que seja possível obter um *tableau fuzzy* completo e livre de *clash*, o raciocinador de conceitos *fuzzy* retorna que  $\mathcal{K}'$  é consistente. Caso contrário, retorna que  $\mathcal{K}'$  é inconsistente. O *tableau fuzzy* completo e livre de *clash* resultante, incluindo os nós de tipos de dados *fuzzy*, representam um modelo ou interpretação  $\mathcal{I}$  que demonstra que  $\mathcal{K}'$  é consistente.

#### 5.4.4 Protótipo FT-FIS

A abordagem FT-FIS foi implementada por um protótipo que instancia a arquitetura de raciocínio apresentada na Seção 5.4.2. Para o desenvolvimento do protótipo, foram consideradas linguagens de representação de ontologias *fuzzy* e de regras *fuzzy*, além de *frameworks* disponíveis para aplicações baseadas em ontologias *fuzzy* e de sistemas de inferência *fuzzy*.

A linguagem de representação de conhecimento utilizada para descrever a ontologia *fuzzy* em  $F\text{-}\mathcal{ALC}(G)$  baseia-se na sintaxe da linguagem definida por fuzzyDL (BOBILLO; STRAC-

CIA, 2008), por ser uma linguagem simplificada que contempla a expressividade de *fuzzy*  $\mathcal{ALC}(D)$ , subconjunto de  $F\text{-}\mathcal{ALC}(G)$ . Além disso, Bobillo e Straccia (2011) disponibilizam um tradutor de Fuzzy OWL 2 para a sintaxe utilizada por fuzzyDL que, futuramente, pode ser estendido para tratar a sintaxe de ontologias *fuzzy* utilizada em FT-FIS.

No contexto da abordagem FT-FIS, a linguagem definida por fuzzyDL foi estendida, neste projeto de doutorado, para representar as expressões de tipos de dados *fuzzy*  $E$ , a definição e o uso de predicados de tipos de dados *fuzzy* sobre o tipo de dados dos números reais, segundo a teoria de grupos de tipos de dados  $G$ . Os elementos adicionais, que tratam predicados *fuzzy* definidos a partir dos predicados embutidos de FT-FIS, são descritos na Tabela 5.4, segundo o formalismo de Backus-Naur estendido (EBNF) (SCOWEN, 1993).

**Tabela 5.4: Sintaxe para definir e referenciar predicados *fuzzy* em FT-FIS.**

```

defFuzPred ::= "(" "define-fuzzy-predicate" predName "(" vars ")" "*real*" E ")";
predName ::= string;
vars ::= varId { "," varId };
varId ::= string;
E ::= atomicPred | "(" "not" atomicPred ")" | "(" "and" E E ")" | "(" "or" E E ")";
atomicPred ::= "crisp" "(" varId "," k1 "," k2 "," a "," b ")" |
  "left-shoulder" "(" varId "," k1 "," k2 "," a "," b ")" |
  "right-shoulder" "(" varId "," k1 "," k2 "," a "," b ")" |
  "triangular" "(" varId "," k1 "," k2 "," a "," b "," c ")" |
  "trapezoidal" "(" varId "," k1 "," k2 "," a "," b "," c "," d ")" |
  "(" "<=" varId y ")" |
  "(" ">=" varId y ")" |
  "(" "<=" varId y ")" |
  "(" ">=" varId y ")" |
  "(" "<" varId y ")" |
  "(" ">" varId y ")" |
  "fuzzyRuleReasoning" "(" ruleBaseName " " inputVars " " outputVars ")";
k1 ::= realNumber;
k2 ::= realNumber;
a ::= realNumber;
b ::= realNumber;
c ::= realNumber;
d ::= realNumber;
y ::= realNumber;
ruleBaseName ::= string;
inputVars ::= vars;
outputVars ::= vars;
realNumber ::= digit { digit } [ "." digit { digit } ];
string ::= letter { letter | digit | "_" };
letter = "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H" | "I" | "J" | "K" | "L" | "M" | "N" | "O" |
  "P" | "Q" | "R" | "S" | "T" | "U" | "V" | "W" | "X" | "Y" | "Z";
digit = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9";

someConcreteConcept ::= "(" "some" concreteProperties "." predName ")";
allConcreteConcept ::= "(" "all" concreteProperties "." predName ")";
concreteProperties ::= concretePropertyName { "," concretePropertyName };
concretePropertyName ::= string;

```

É possível notar que os elementos definidos por `atomicPred` correspondem aos predicados embutidos providos por FT-FIS (Tabela 5.2). No caso do predicado *fuzzyRuleReasoning*, o parâmetro `ruleBaseName` se refere ao nome da base de regras *fuzzy* a ser considerada, pois a linguagem de representação de regras *fuzzy* utilizada no protótipo permite a definição de diversas



bases de regras identificadas por um nome. Esse recurso é interessante para alguns domínios de aplicação, por exemplo sistemas de recomendação ou suporte a decisão, em que cada base de regras *fuzzy* pode representar as preferências de uma categoria de indivíduos.

Além da definição dos predicados *fuzzy*, a Tabela 5.4 descreve os construtores de conceito existencial (`someConcreteConcept`) e universal (`allConcreteConcept`) que consideram os predicados definidos a partir das expressões de predicados de tipos de dados *fuzzy*  $E$ . O elemento `concreteProperties` faz referência a propriedades concretas  $T_1, \dots, T_n$  definidas sobre o domínio dos números reais. A aridade e a ordem das variáveis dos predicados *fuzzy* identificados por `predName` devem ser respeitadas ao utilizar os construtores de conceito `someConcreteConcept` e `allConcreteConcept` em definições ou descrições de conceitos na ontologia *fuzzy*. A partir dessas extensões, a linguagem de representação de ontologias *fuzzy* utilizada em FT-FIS atende aos requisitos de modelagem apresentados na Seção 5.2.

Assim como em HyFOM, as regras *fuzzy* são descritas na linguagem *Fuzzy Markup Language* (FML) (ACAMPORA; LOIA, 2005). FML provê interoperabilidade e independência de implementação de SIF, por ser uma linguagem que modela os requisitos gerais para o raciocínio baseado em regras *fuzzy*. Com a ferramenta gráfica *Visual FML Tool*, especialistas do domínio podem modelar variáveis linguísticas, partições *fuzzy* e diversas bases de regras *fuzzy*, cada uma com seu respectivo método de inferência. Também é possível especificar o método de defuzzificação para cada variável de saída. Na ontologia *fuzzy*, os identificadores das bases de regras *fuzzy* são referenciados pelo elemento `ruleBaseName` da Tabela 5.4. Conforme mencionado anteriormente, as variáveis linguísticas especificadas em FML devem estar associadas às respectivas propriedades concretas na ontologia *fuzzy*, com mesmo nome e universo de discurso.

A implementação da arquitetura de raciocínio e da abordagem de integração de FT-FIS foi realizada com base na plataforma Java. Tanto o raciocinador de conceitos *fuzzy* quanto o gerenciador de tipos de dados *fuzzy* foram desenvolvidos com o suporte da API de `fuzzyDL`. Vale ressaltar que ela foi utilizada, principalmente, como referência para a análise sintática da linguagem de ontologias *fuzzy* e para a definição dos pacotes e classes que representam os elementos da ontologia *fuzzy*. Assim, os serviços de inferência de `fuzzyDL` não foram utilizados para a implementação do algoritmo baseado em *fuzzy tableau* de  $F\text{-}\mathcal{ALC}(G)$ , que foi desenvolvido diretamente a partir das especificações definidas por Pan (2007), Wang e Ma (2008).

Os serviços de raciocínio disponibilizados pelo raciocinador de conceitos *fuzzy* para aplicações baseadas em ontologias *fuzzy* são:

- *Verificação de consistência da base de conhecimento*: tenta construir um *tableau fuzzy* para a base de conhecimento da ontologia *fuzzy*  $\mathcal{K}$ , acionando o raciocínio baseado em regras *fuzzy* quando necessário. Se for possível obter um *tableau fuzzy* para  $\mathcal{K}$ , então  $\mathcal{K}$  é consistente; caso contrário, é inconsistente;
- *Verificação de consistência da base de conhecimento com defuzzificação*: além de tentar construir um *tableau fuzzy*, também retorna os valores defuzzificados de cada nó de tipo de dado *fuzzy* processado por  $DC_{d_{real}}$  caso seja possível obter um *tableau fuzzy* para  $\mathcal{K}$ ;
- *Verificação de satisfazibilidade para uma asserção de conceito fuzzy*  $\psi = \langle x : D \bowtie n \rangle$ : verifica a consistência de  $\mathcal{K}' = \mathcal{K} \cup \{\langle x : D \bowtie n \rangle\}$ ;
- *Verificação de satisfazibilidade para uma asserção de conceito fuzzy*  $\psi = \langle x : D \bowtie n \rangle$  com defuzzificação: além de tentar construir um *tableau fuzzy* para  $\mathcal{K}'$ , também retorna os valores defuzzificados de cada nó de tipo de dado *fuzzy* processado por  $DC_{d_{real}}$  caso seja possível obter um *tableau fuzzy* para  $\mathcal{K}'$ ;
- *Verificação de instâncias*: verifica se uma asserção de conceito *fuzzy*  $\psi = \langle x : D \bowtie n \rangle$  é uma implicação lógica de  $\mathcal{K}$ . Conforme descrito na Seção 3.5.2 do Capítulo 3, essa tarefa é reduzida para a verificação de consistência da base de conhecimento  $\mathcal{K}' = \mathcal{K} \cup \{\langle x : D \bowtie' n \rangle\}$ , onde  $\bowtie' n$  corresponde a uma restrição de grau conjugada de  $\bowtie n$ . Se for possível construir um *tableau fuzzy* para  $\mathcal{K}'$ , ele representa um contra-exemplo, ou seja, uma interpretação em que a asserção  $\psi$  não é verdadeira. Caso contrário, o raciocinador de conceitos *fuzzy* retorna que  $\psi$  é uma implicação lógica de  $\mathcal{K}$ .

Para representar conjuntos *fuzzy* e realizar as operações de conjunção *fuzzy* e de restrição de grau, o verificador de tipos de dados *fuzzy*  $DC_{d_{real}}$  e o SIF foram implementados com base no FuzzyJ Toolkit (ORCHARD, 2001). Com relação ao SIF, FuzzyJ Toolkit disponibiliza métodos de inferência de Mamdani e Larsen com suporte a entradas e saídas *fuzzy*, além da defuzzificação por centro de área, meio dos máximos e *moment defuzzification*. Também provê uma API Java para acionar o processo de inferência e realizar a defuzzificação para um determinado conjunto *fuzzy*. Portanto, FuzzyJ Toolkit atende aos requisitos do SIF, no contexto da abordagem FT-FIS, descritos na Seção 5.4.3. O SIF implementado conta com um tradutor de bases de regras *fuzzy* especificadas em FML para os objetos correspondentes das classes de FuzzyJ Toolkit, por meio da API Java para processamento de XML (JAXP - *Java API for XML Processing*). No entanto, assim como em HyFOM, outras implementações de SIF podem ser utilizadas, desde que haja um tradutor de FML para a linguagem específica do SIF.

O protótipo desenvolvido segundo a abordagem FT-FIS foi utilizado em experimentos descritos no Capítulo 6, que discute os resultados obtidos em comparação com HyFOM e alguns trabalhos correlatos da literatura.

## 5.5 Considerações Finais

Este capítulo apresentou as duas abordagens para integração de raciocínio de ontologias *fuzzy* e sistemas de inferência *fuzzy* definidas e desenvolvidas no âmbito deste projeto de doutorado: HyFOM e FT-FIS. Primeiramente, foram identificados os requisitos de modelagem relacionados à integração e, posteriormente, como os mesmos são atendidos pelas abordagens definidas. Foram descritos detalhes dos componentes das arquiteturas e como é realizada a integração de raciocínio de ontologias *fuzzy* e SIF, além de aspectos associados à implementação de protótipos.

No próximo capítulo são discutidos os experimentos realizados, os resultados obtidos e a comparação com os resultados de trabalhos correlatos.

# Capítulo 6

## AVALIAÇÃO DAS ABORDAGENS DESENVOLVIDAS

---

---

### 6.1 Considerações Iniciais

Para analisar as contribuições e limitações de HyFOM e FT-FIS, foram realizados experimentos envolvendo dois domínios distintos: recomendação de atrações turísticas e análise de risco químico em alimentos. Em ambos os domínios, foram identificados conceitos, propriedades numéricas e regras *fuzzy* que correspondem aos requisitos de modelagem apresentados na Seção 5.2. Desta forma, os protótipos de HyFOM e FT-FIS foram aplicados para proporcionar inferências combinando ontologias *fuzzy* e sistemas de inferência *fuzzy*.

Além de demonstrar a aplicação das abordagens desenvolvidas, os testes exploraram tanto a integração de inferências de ontologias *fuzzy* para SIF quanto dos resultados do SIF para a ontologia *fuzzy*. Neste sentido, os experimentos foram organizados da seguinte forma:

1. *Entradas numéricas*: os valores das propriedades numéricas que correspondem às entradas para o SIF são modeladas de forma explícita na ontologia *fuzzy* como valores numéricos. Neste caso, o foco dos experimentos está no uso das inferências do SIF na ontologia *fuzzy* ( $SIF \rightarrow ontologia\ fuzzy$ ) para definir valores de propriedades numéricas associadas às saídas do SIF ou verificar a pertinência para conceitos abstratos;
2. *Entradas fuzzy*: os valores das propriedades numéricas que correspondem às entradas para o SIF são obtidas a partir de inferências da ontologia *fuzzy*, sendo que seus valores são representados por conjuntos *fuzzy*. As inferências do SIF também são utilizadas tanto para definir valores de propriedades numéricas associadas às saídas do SIF quanto para verificar a pertinência para conceitos abstratos. Desta forma, ambas formas de integração de inferências  $ontologia\ fuzzy \rightarrow SIF$  e  $SIF \rightarrow ontologia\ fuzzy$  são consideradas.

Para tratar a integração  $SIF \rightarrow \text{ontologia fuzzy}$ , cada experimento considerou o uso direto das saídas defuzificadas do SIF na aplicação, com o objetivo de avaliar o resultado numérico inferido pelo SIF. Além disso, também foram tratadas situações em que restrições são aplicadas às saídas do SIF, baseadas em predicados *fuzzy*, para a definição de conceitos abstratos na ontologia *fuzzy*. Com isso, foi possível explorar como inferências de FIS podem influenciar nas tarefas de raciocínio da ontologia *fuzzy*.

Dentre os trabalhos relacionados, o motor de inferência fuzzyDL (BOBILLO; STRACCIA, 2008) foi selecionado para comparação com HyFOM e FT-FIS nos dois domínios de aplicação. FuzzyDL foi escolhido por ser um dos principais motores de inferência de lógica de descrição *fuzzy*, que oferece suporte a propriedades numéricas associadas a predicados *fuzzy*. Além disso, fuzzyDL é capaz de representar regras *fuzzy* baseadas em implicações e também segundo a semântica de Mamdani, oferecendo três métodos de defuzificação para obter valores numéricos: Menor dos Máximos (*Smallest of Maxima* - SOM), Meio dos Máximos (*Middle of Maxima* - MOM) e Maior dos Máximos (*Largest of Maxima* - LOM). Em função dessas características, além de outras apresentadas na Tabela 4.1 do Capítulo 4, fuzzyDL é o trabalho correlato que mais se aproxima de HyFOM e FT-FIS. Assim, foram consideradas três abordagens de fuzzyDL para tratar ontologias *fuzzy* e regras *fuzzy*:

- Implicações *fuzzy*: as regras *fuzzy* correspondem a regras implicativas, em que a semântica das regras é dada por operações de implicação *fuzzy* e as regras disparadas são combinadas por meio de conjunção. Nos experimentos, foi considerada a semântica de Zadeh, mais comum nas aplicações, que define a operação de mínimo para conjunções, e escolhida a inclusão de conjuntos de Zadeh para implicações *fuzzy*. Assim, como os experimentos consideraram todas as regras *fuzzy* com grau de verdade igual a 1.0, a semântica da regra é dada por  $\beta \geq \alpha$  para implicações  $\alpha \rightarrow \beta$  segundo a inclusão de conjuntos de Zadeh. A implicação de Kleene-Dienes não foi considerada porque o grau do consequente  $\beta$  é sempre igual a 1 quando  $0 < \alpha \leq 1$ , para implicações com grau de verdade igual a 1.0. Tal semântica não é apropriada para um contexto de sistemas de recomendação, uma vez que o grau de consequente  $\beta$  seria sempre inferido com o grau máximo para qualquer  $\alpha \neq 0$ ;
- Regras de Mamdani e defuzificação: as regras *fuzzy* seguem o modelo de Mamdani e são utilizadas em conjunto com os métodos de defuzificação SOM, MOM e LOM;
- Regras de Mamdani e consulta por satisfazibilidade: as regras *fuzzy* são modeladas como conceitos da ontologia seguindo o modelo de Mamdani e os valores das propriedades de saída são obtidos por consultas de satisfazibilidade máxima e mínima providas por

fuzzyDL. Este caso corresponde ao uso de fuzzyDL somente com recursos de modelagem e de consulta disponíveis para raciocinadores de lógica de descrição *fuzzy*.

Para cada abordagem de fuzzyDL, o objetivo é comparar dois aspectos principais: semântica das inferências providas pelas regras *fuzzy* e a integração com a ontologia *fuzzy*. A semântica será analisada em função dos valores inferidos a partir do raciocínio baseado em regras *fuzzy*. A integração será verificada ao utilizar as inferências obtidas por meio das regras *fuzzy* em conceitos da ontologia e ao utilizar o raciocínio da ontologia para definir os valores de entrada para as regras *fuzzy*.

Vale ressaltar que HyFOM usa o motor de inferência fuzzyDL como um de seus componentes, porém somente para lidar com inferências relacionadas à ontologia *fuzzy*, pois as regras *fuzzy* são processadas pelo SIF. Nas comparações com HyFOM e FT-FIS, fuzzyDL foi executado de forma independente de HyFOM, sendo que a ontologia *fuzzy* e as regras *fuzzy* consideradas nos experimentos foram adaptadas de acordo com as abordagens de fuzzyDL citadas anteriormente.

Este capítulo descreve os experimentos realizados da seguinte forma. A Seção 6.2 apresenta o contexto da aplicação de recomendação de atrações turísticas, incluindo a ontologia *fuzzy* e as regras *fuzzy* consideradas nos testes. Na sequência, a Seção 6.3 descreve os experimentos realizados no domínio de análise de risco químico em alimentos, baseado em dados reais disponibilizados pela Embrapa Gado de Leite de Juiz de Fora, MG. Ambas seções abordam a aplicação de HyFOM e FT-FIS, assim como a comparação de seus resultados com fuzzyDL, discutindo contribuições e limitações de cada abordagem, contextualizadas nos dois domínios considerados.

## 6.2 Recomendação de Atrações Turísticas

A primeira aplicação de HyFOM e FT-FIS, baseada em dados sintéticos, refere-se a um sistema para recomendar atrações turísticas que podem ser visitadas em um período curto de tempo, geralmente durante meio período ou no máximo um dia inteiro, de uma determinada cidade. Neste contexto, a aplicação pode auxiliar turistas na decisão sobre quais as melhores atrações dependendo de fatores subjetivos e imprecisos como tempo estimado de visitaç o, relev ncia da atraç o, entre outros.

Para descrever os experimentos realizados, primeiramente, a ontologia *fuzzy* e as regras *fuzzy* s o apresentadas; depois s o discutidos os experimentos com entradas num ricas e, por

fim, os experimentos envolvendo entradas *fuzzy*.

### 6.2.1 Ontologia *Fuzzy* e Regras *Fuzzy*

No caso específico da aplicação considerada nos experimentos, os fatores subjetivos associados as atrações turísticas são representados por três características principais: *satisfação*, *relevância* e *duração da visita*. A satisfação geral associada a uma atração turística pode ser definida com base na sua relevância, que descreve a sua importância referente ao quanto uma atração é “imperdível”, e o tempo estimado de visita, sendo que essas características envolvem termos subjetivos e imprecisos como satisfação *alta*, relevância *média* ou tempo de visita *baixo*, baseados na opinião de turistas experientes. Dependendo do perfil do turista, a satisfação pode refletir a preferência pela relevância ou pelo tempo de visita, ou ainda por uma relação entre as duas características. Por exemplo, para turistas com restrições de tempo e que desejem visitar muitas atrações, a prioridade deve ser maior para atrações relevantes que levam menor tempo para visita.

As atrações turísticas podem ser classificadas em categorias como museus, parques, locais religiosos, estabelecimentos comerciais, entre outros. A partir dessas categorias, turistas experientes podem opinar qual a relevância e o tempo de visita, dependendo de suas preferências subjetivas. Por exemplo, turistas que gostam de atividades ao ar livre podem sugerir que parques são atrações com relevância alta e duração de visita médio. Desta forma, valores de relevância e de tempo de visita podem ser obtidos em função das categorias de atrações turísticas. Além disso, outros conceitos podem ser modelados em função da satisfação das atrações para especificar atrações interessantes, atrações acima da média, entre outras. Esses conceitos podem ser diretamente utilizados pela aplicação para recomendar as atrações turísticas aos usuários. Ainda, os valores de satisfação podem ser considerados para priorizar as atrações e sugerir uma classificação para orientar a decisão dos turistas.

Neste contexto, os principais conceitos, propriedades numéricas e predicados *fuzzy* associados às atrações turísticas foram modelados em uma ontologia *fuzzy*. Uma versão foi representada segundo a sintaxe de fuzzyDL para ser utilizada em HyFOM e nos testes com fuzzyDL (Tabela 6.1), enquanto outra versão foi modelada de acordo com a sintaxe de fuzzyDL estendida para FT-FIS (Tabela 6.2).

A ontologia *fuzzy* baseia-se na semântica da lógica *fuzzy* de Zadeh, portanto as operações para conjunção, disjunção e negação são definidas pelo mínimo, máximo e complemento *fuzzy* padrão, respectivamente. A satisfação (*satisfaction*), relevância (*relevance*) e tempo de visita (*timing*) são representados como propriedades funcionais numéricas, cujos valores estão asso-

**Tabela 6.1: Ontologia fuzzy sobre atrações turísticas utilizada em HyFOM e fuzzyDL.**

```

(define-fuzzy-logic zadeh)

(functional relevance)
(range relevance *real* 0.0 10.0)
(define-fuzzy-concept lowRelevance left-shoulder(0.0, 10.0, 1.0, 5.0))
(define-fuzzy-concept mediumRelevance triangular(0.0, 10.0, 1.0, 5.0, 9.0))
(define-fuzzy-concept highRelevance right-shoulder(0.0, 10.0, 5.0, 9.0))

(functional timing)
(range timing *real* 0.0 8.0)
(define-fuzzy-concept lowTiming left-shoulder(0.0, 8.0, 1.0, 4.0))
(define-fuzzy-concept mediumTiming triangular(0.0, 8.0, 1.0, 4.0, 7.0))
(define-fuzzy-concept highTiming right-shoulder(0.0, 8.0, 4.0, 7.0))

(functional satisfaction)
(range satisfaction *real* 0.0 10.0)
(define-fuzzy-concept veryLowSatisfaction left-shoulder(0.0, 10.0, 1.0, 3.0) )
(define-fuzzy-concept lowSatisfaction triangular(0.0, 10.0, 1.0, 3.0, 5.0) )
(define-fuzzy-concept mediumSatisfaction triangular(0.0, 10.0, 3.0, 5.0, 7.0) )
(define-fuzzy-concept highSatisfaction triangular(0.0, 10.0, 5.0, 7.0, 9.0) )
(define-fuzzy-concept veryHighSatisfaction right-shoulder(0.0, 10.0, 7.0, 9.0) )
(define-fuzzy-concept aboveAverageSatisfaction right-shoulder(0.0, 10.0, 3.0, 5.0) )

(define-concept InterestingAttraction
  (and TouristicAttraction (some satisfaction veryHighSatisfaction)))
(define-concept AboveAverageAttraction
  (and TouristicAttraction (some satisfaction aboveAverageSatisfaction)))

(define-concept Museum
  (and TouristicAttraction (some relevance mediumRelevance)(some timing highTiming) ))
(define-concept Park
  (and TouristicAttraction (some relevance highRelevance) (some timing mediumTiming) ))
(define-concept Shopping
  (and TouristicAttraction (some relevance lowRelevance) (some timing lowTiming) ))
(define-concept ReligiousSite
  (and TouristicAttraction (some relevance mediumRelevance) (some timing mediumTiming) ))
(define-concept Sports
  (and TouristicAttraction (some relevance mediumRelevance) (some timing lowTiming) ))

(instance attraction1 TouristicAttraction 1.0)
(instance attraction1 (= relevance 4.0) 1.0)
(instance attraction1 (= timing 1.5) 1.0)

(instance attraction2 TouristicAttraction 1.0)
(instance attraction2 (= relevance 7.5) 1.0)
(instance attraction2 (= timing 2.0) 1.0)

(instance attraction3 Museum 1.0)
(instance attraction4 Park 0.8)
(instance attraction5 ReligiousSite 0.5)
(instance attraction6 Shopping 0.2)

```

ciados a predicados *fuzzy* como *baixo*, *médio*, *alto*, entre outros, definidos por conjuntos *fuzzy*. Também são definidos conceitos abstratos em função dos valores dessas propriedades, como o conceito de atração interessante (*InterestingAttraction*), atração acima da média (*AboveAverageAttraction*), além de categorias de atrações como Museu (*Museum*), Parque (*Park*), entre outros. Nesta ontologia *fuzzy* simplificada, o relacionamento entre as categorias de atrações e as propriedades de relevância e de tempo de visitaç o exemplificam um perfil específico de prefer ncias de turistas, sendo que   poss vel estend -la para contemplar outros perfis que sejam de interesse para a recomenda o de atrações. Por fim, s o ilustradas inst ncias de atrações



**Tabela 6.2: Ontologia fuzzy sobre atrações turísticas utilizada em FT-FIS.**

```

(define-fuzzy-logic zadeh)

(functional relevance)
(range relevance *real* 0.0 10.0)

(functional timing)
(range timing *real* 0.0 8.0)

(functional satisfaction)
(range satisfaction *real* 0.0 10.0)

(define-fuzzy-predicate lowRelevance(x) *real* left-shoulder(x, 0.0, 10.0, 1.0, 5.0) )
(define-fuzzy-predicate mediumRelevance(x) *real* triangular(x, 0.0, 10.0, 1.0, 5.0, 9.0) )
(define-fuzzy-predicate highRelevance(x) *real* right-shoulder(x, 0.0, 10.0, 5.0, 9.0) )

(define-fuzzy-predicate lowTiming(x) *real* left-shoulder(x, 0.0, 8.0, 1.0, 4.0) )
(define-fuzzy-predicate mediumTiming(x) *real* triangular(x, 0.0, 8.0, 1.0, 4.0, 7.0) )
(define-fuzzy-predicate highTiming(x) *real* right-shoulder(x, 0.0, 8.0, 4.0, 7.0) )

(define-fuzzy-predicate verylowSatisfaction(x) *real* left-shoulder(x, 0.0, 10, 1.0, 3.0) )
(define-fuzzy-predicate lowSatisfaction(x) *real* triangular(x, 0.0, 10.0, 1.0, 3.0, 5.0) )
(define-fuzzy-predicate mediumSatisfaction(x) *real* triangular(x, 0.0, 10.0, 3.0, 5.0, 7.0) )
(define-fuzzy-predicate highSatisfaction(x) *real* triangular(x, 0.0, 10.0, 5.0, 7.0, 9.0) )
(define-fuzzy-predicate veryHighSatisfaction(x) *real* right-shoulder(x, 0.0, 10.0, 7.0, 9.0) )
(define-fuzzy-predicate aboveAverageSatisfaction(x) *real* right-shoulder(x, 0.0, 10.0, 3.0, 5.0) )

(define-fuzzy-predicate high(v0,v1,v2) *real*
(and fuzzyRuleReasoning("RuleBase1";v0,v1,v2) veryHighSatisfaction(v2)))
(define-fuzzy-predicate aboveAverage(v0,v1,v2) *real*
(and fuzzyRuleReasoning("RuleBase1";v0,v1,v2) aboveAverageSatisfaction(v2)))

(define-concept InterestingAttraction
(and TouristicAttraction (some relevance,timing,satisfaction.high)))
(define-concept AboveAverageAttraction
(and TouristicAttraction (some relevance,timing,satisfaction.aboveAverage)))

(define-primitive-concept Museum
(and (some relevance.mediumRelevance) (some timing.highTiming) ))
(define-primitive-concept Park
(and (some relevance.highRelevance) (some timing.mediumTiming) ))
(define-primitive-concept Shopping
(and (some relevance.lowRelevance) (some timing.lowTiming) ))
(define-primitive-concept ReligiousSite
(and (some relevance.mediumRelevance) (some timing.mediumTiming) ))
(define-primitive-concept Sports
(and (some relevance.mediumRelevance) (some timing.lowTiming) ))

(instance attraction1 TouristicAttraction 1.0)
(instance attraction1 (= relevance 4.0) 1.0)
(instance attraction1 (= timing 1.5) 1.0)

(instance attraction2 TouristicAttraction 1.0)
(instance attraction2 (= relevance 7.5) 1.0)
(instance attraction2 (= timing 2.0) 1.0)

(instance attraction3 Museum 1.0)
(instance attraction4 Park 0.8)
(instance attraction5 ReligiousSite 0.5)
(instance attraction6 Shopping 0.2)

```

turísticas, algumas contendo valores numéricos explícitos de relevância e tempo de visitação (*attraction1*, *attraction2*), enquanto outras são instâncias das categorias de atrações com graus de pertinência específicos (*attraction3* a *attraction6*). Por se tratar de uma ontologia *fuzzy*, é possível expressar a compatibilidade entre uma instância e um conceito por meio do grau de

pertinência, sendo que quanto mais próximo do grau 1.0, mais compatível é a instância com relação à definição do conceito.

Para inferir a satisfação de uma atração a partir da combinação de relevância e tempo de visita, um conjunto de regras *fuzzy* foi modelado com base na experiência de turistas. Nos experimentos realizados, o conjunto de regras *fuzzy* foi modelado de modo que a satisfação de uma atração seja maior conforme a relevância for maior e o tempo de visita menor. A intenção é realizar recomendações de modo que um turista visite várias atrações relevantes em um curto período de tempo. Entretanto, outros conjuntos de regras poderiam ter sido utilizados, novamente dependendo do perfil de turista mais apropriado para a aplicação. A Figura 6.1 mostra o conjunto de regras *fuzzy* considerado nos experimentos, modelado com a ferramenta Visual FML Tool que automaticamente gera um arquivo FML correspondente.

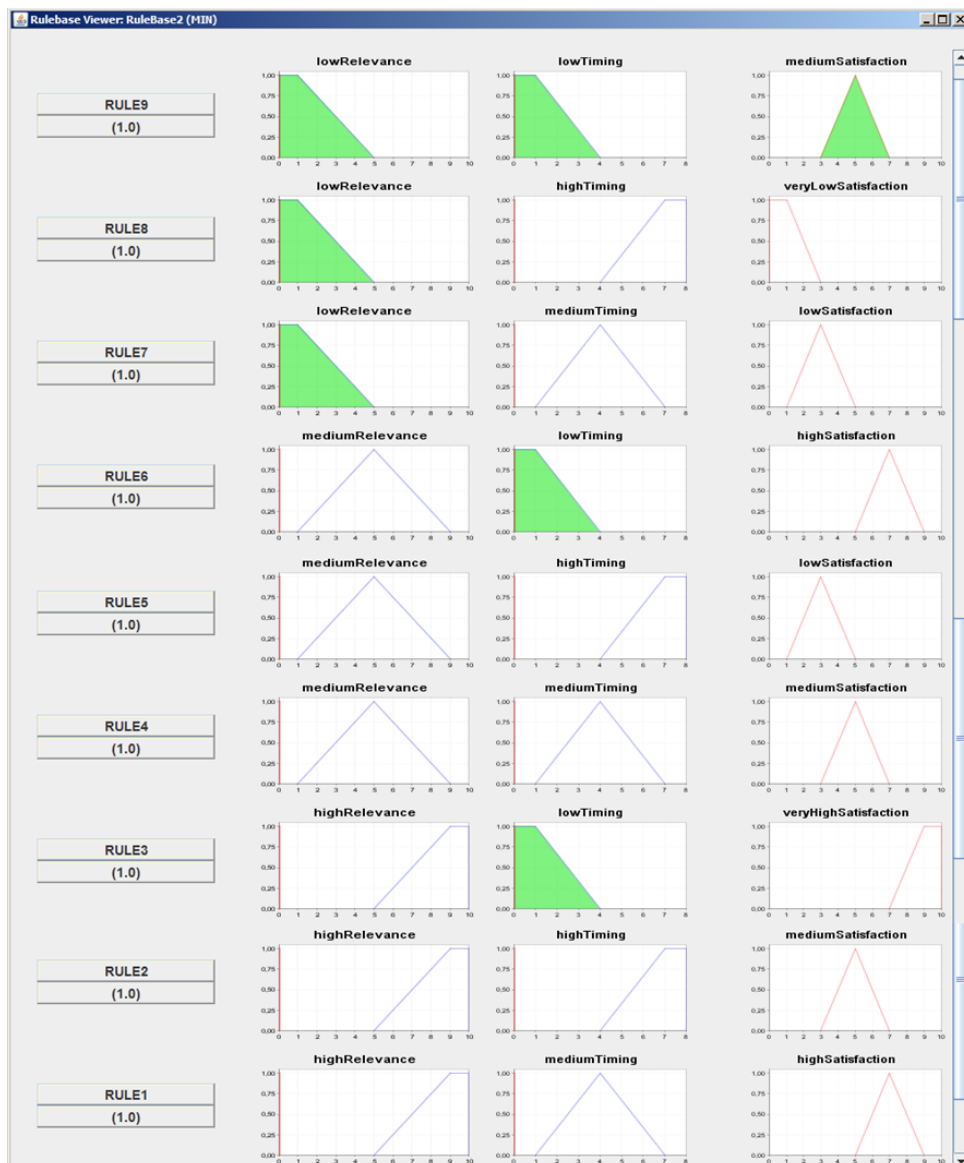


Figura 6.1: Regras *fuzzy* utilizadas nos experimentos para recomendação de atrações turísticas.

Com relação aos experimentos comparativos com fuzzyDL, utilizou-se como base a versão da ontologia *fuzzy* da Tabela 6.1 e então adicionadas regras *fuzzy* dependendo do tipo de abordagem de representação em fuzzyDL. As regras *fuzzy* da Figura 6.1 foram representadas de duas formas em fuzzyDL: como implicações *fuzzy* (Tabela 6.3) e por meio de definições de conceito e operadores *fuzzy* (Tabela 6.4) conforme proposto por Bobillo e Straccia (2008) para modelar regras segundo a semântica de Mamdani.

**Tabela 6.3: Regras implicativas *fuzzy* em fuzzyDL.**

(implies (and touristicAttraction (some relevance lowRelevance) (some timing lowTiming)) (some satisfaction mediumSatisfaction))
(implies (and touristicAttraction (some relevance lowRelevance) (some timing mediumTiming)) (some satisfaction lowSatisfaction))
(implies (and touristicAttraction (some relevance lowRelevance) (some timing highTiming)) (some satisfaction veryLowSatisfaction))
(implies (and touristicAttraction (some relevance mediumRelevance) (some timing lowTiming)) (some satisfaction highSatisfaction))
(implies (and touristicAttraction (some relevance mediumRelevance) (some timing mediumTiming)) (some satisfaction mediumSatisfaction))
(implies (and touristicAttraction (some relevance mediumRelevance) (some timing highTiming)) (some satisfaction lowSatisfaction))
(implies (and touristicAttraction (some relevance highRelevance) (some timing lowTiming)) (some satisfaction veryHighSatisfaction))
(implies (and touristicAttraction (some relevance highRelevance) (some timing mediumTiming)) (some satisfaction highSatisfaction))
(implies (and touristicAttraction (some relevance highRelevance) (some timing highTiming)) (some satisfaction mediumSatisfaction))

Tendo em vista a semântica deste domínio, é possível identificar os requisitos de modelagem descritos na Seção 5.2 do Capítulo 5, que indicam que HyFOM e FT-FIS podem ser aplicados para tratar a integração de raciocínio de ontologias *fuzzy* e regras *fuzzy*:

1. Valores de propriedades concretas definidos a partir de regras *fuzzy* que combinam outras propriedades concretas numéricas: é o caso da satisfação de uma atração turística, cujo valor pode ser definido a partir de regras *fuzzy* que combinam valores imprecisos de relevância e de tempo estimado de visitaç o (Figura 6.1);
2. Valores de propriedades concretas obtidos por meio de infer ncias da ontologia *fuzzy*: é o caso das categorias de atrações turísticas (museus, parques, locais religiosos), que podem definir os valores de relevância e de tempo de visitaç o, dependendo do perfil do turista;
3. Definição de conceitos abstratos em função de valores de propriedades concretas numéricas: é o caso dos conceitos modelados em função da satisfação das atrações, por exemplo, atrações interessantes e atrações acima da média.

**Tabela 6.4: Regras *fuzzy* sob a semântica de Mamdani em fuzzyDL.**

```

(define-concept Rule1
  (and (some relevance highRelevance) (some timing lowTiming) (some satisfaction veryHighSatisfaction)))

(define-concept Rule2
  (and (some relevance highRelevance) (some timing mediumTiming) (some satisfaction highSatisfaction)))

(define-concept Rule3
  (and (some relevance highRelevance) (some timing highTiming) (some satisfaction mediumSatisfaction)))

(define-concept Rule4
  (and (some relevance mediumRelevance) (some timing lowTiming) (some satisfaction highSatisfaction)))

(define-concept Rule5
  (and (some relevance mediumRelevance) (some timing mediumTiming) (some satisfaction mediumSatisfaction)))

(define-concept Rule6
  (and (some relevance mediumRelevance) (some timing highTiming) (some satisfaction lowSatisfaction)))

(define-concept Rule7
  (and (some relevance lowRelevance) (some timing lowTiming) (some satisfaction mediumSatisfaction)))

(define-concept Rule8
  (and (some relevance lowRelevance) (some timing mediumTiming) (some satisfaction lowSatisfaction)))

(define-concept Rule9
  (and (some relevance lowRelevance) (some timing highTiming) (some satisfaction veryLowSatisfaction)))

(define-concept RuleSet (or Rule1 Rule2 Rule3 Rule4 Rule5 Rule6 Rule7 Rule8 Rule9) )

```

Assim, a ontologia *fuzzy* e as regras *fuzzy* descritas nesta seção foram aplicadas em HyFOM, FT-FIS e fuzzyDL para recomendar atrações turísticas a partir da combinação das inferências de ontologia *fuzzy* e SIF. As próximas subseções descrevem os resultados obtidos neste domínio e as comparações feitas com fuzzyDL.

### 6.2.2 Experimentos com Entradas Numéricas

Os experimentos com entradas numéricas consideram que a ontologia *fuzzy* contém asserções explícitas envolvendo instâncias e propriedades concretas numéricas correspondentes às entradas do SIF. Portanto, o objetivo é analisar o uso das saídas do SIF na ontologia *fuzzy* ( $SIF \rightarrow ontologia\ fuzzy$ ), já que os valores de entradas são modelados explicitamente.

Resultados preliminares dos experimentos com entradas numéricas comparando HyFOM e fuzzyDL foram publicados por Yaguinuma et al. (2013). Foram realizados testes mais abrangentes, analisando não somente HyFOM mas também FT-FIS frente às abordagens disponíveis em fuzzyDL. Com relação aos valores de entrada para o SIF, os testes consideraram valores de relevância em  $[1, 9]$  e de tempo de visitaç o em  $[1, 7]$ , com um incremento de 1, de modo a cobrir diferentes combinações de entradas para as regras *fuzzy* utilizadas (Figura 6.1). Assim, foram simuladas 63 instâncias de atrações turísticas, cada uma com os valores de relevância e tempo de visitaç o representados explicitamente na ontologia *fuzzy*, conforme exemplificado pelas

instâncias *attraction1* e *attraction2* das Tabelas 6.1 e 6.2. As 63 instâncias simuladas foram adicionadas às versões da ontologia *fuzzy* das Tabelas 6.1 e 6.2 para realizar os experimentos com entradas numéricas utilizando HyFOM e FT-FIS.

Neste contexto, foram realizadas duas consultas envolvendo a recomendação de atrações turísticas com base nos valores de satisfação inferidos por SIF.

### Consulta 1: satisfação inferida a partir de regras *fuzzy*

A primeira consulta refere-se ao uso direto dos valores inferidos de satisfação para classificação das atrações a fim de comparar os resultados do raciocínio baseado em regras *fuzzy* providos por HyFOM, FT-FIS e fuzzyDL.

Segundo a abordagem de integração de HyFOM, os valores de satisfação inferidos pelo SIF são modelados como asserções explícitas na ontologia *fuzzy*, portanto basta consultar diretamente pelo valor da propriedade de satisfação para cada instância de atração turística. No caso de FT-FIS, definiu-se um predicado *fuzzy* a partir do predicado *fuzzyRuleReasoning*, que foi utilizado para a definição do conceito abstrato *InterestingAttraction*:

```
(define-fuzzy-predicate high(v0,v1,v2) *real* fuzzyRuleReasoning("RuleBase1";v0,v1,v2))
(define-concept InterestingAttraction (some relevance,timing,satisfaction.high))
```

A consulta submetida a FT-FIS foi a verificação de satisfazibilidade para

$$\psi = \langle x : InterestingAttraction > 0.0 \rangle$$

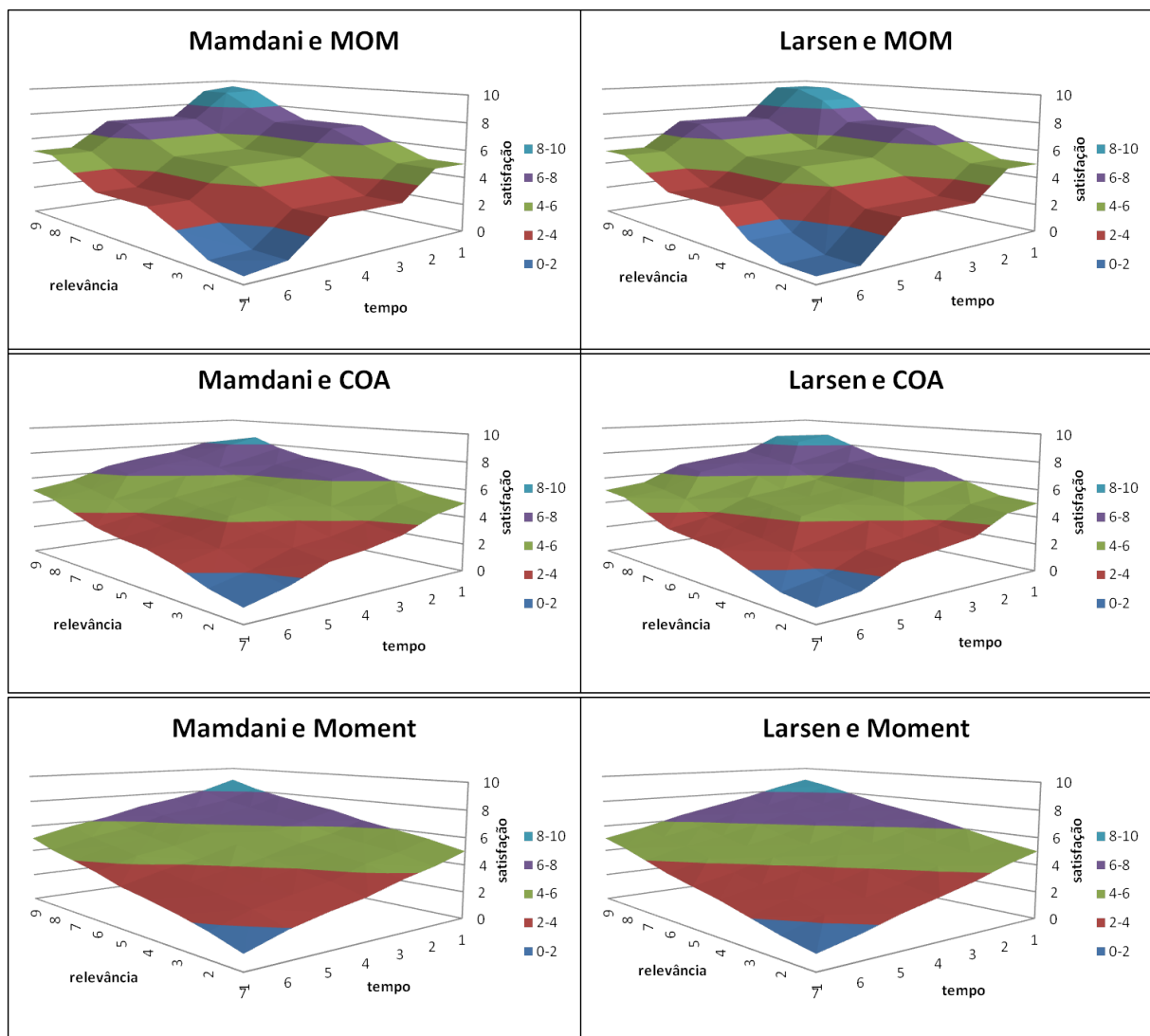
com defuzzificação dos valores de satisfação inferidos pelo SIF para todas instâncias *x* de atrações turísticas. O grau de verdade considerado na consulta ( $> 0.0$ ) foi utilizado para que a saída *fuzzy* do SIF não fosse restrita por um grau específico, ou seja, as saídas do SIF são consideradas de forma direta, sem restrições de grau. Os métodos de defuzzificação considerados foram meio dos máximos (MOM), centro de área (COA) e defuzzificação *Moment*. Os valores de satisfação obtidos por HyFOM e FT-FIS são descritos na Tabela 6.5 e graficamente na Figura 6.2. Na tabela, os valores de relevância são indicados pela letra R e de tempo de visitação pela letra T.

Os resultados de satisfação inferidos com HyFOM e FT-FIS são idênticos, pois se baseiam em uma mesma implementação de SIF. Considerando os métodos de raciocínio de Mamdani e Larsen, é possível notar que existe uma diferença sutil entre os resultados. De forma geral, é mantida a semântica esperada para a recomendação de atrações segundo as regras *fuzzy* modeladas: quanto maior a relevância e menor o tempo de visitação, maior é a satisfação, tendo em vista turistas que desejem visitar o maior número de atrações relevantes em um

**Tabela 6.5: Resultados de HyFOM/FT-FIS para consulta 1 e entradas numéricas.**

Mamdani e MOM									
T \ R	1	2	3	4	5	6	7	8	9
1	5	5	6	7	7	7	8	9,25	9,5
2	5	5	6	7	7	7	8	9,167	9,167
3	3	3	4	5	5	5	6	7	7
4	3	3	4	5	5	5	6	7	7
5	3	3	4	5	5	5	6	7	7
6	0,833	0,833	2	3	3	3	4	5	5
7	0,5	0,75	2	3	3	3	4	5	5
Mamdani e COA									
T \ R	1	2	3	4	5	6	7	8	9
1	5	5,333	6	6,667	7	7,208	7,75	8,542	8
2	4,5	4,875	5,333	6,106	6,5	6,734	7,083	7,882	8
3	3,5	3,894	4,667	5,125	5,5	5,736	6,444	6,958	7,333
4	3	3,333	4	4,667	5	5,333	6	6,667	7
5	2,667	3,042	3,556	4,264	4,5	4,875	5,333	6,106	6,5
6	1,667	2,118	2,917	3,266	3,5	3,894	4,667	5,125	5,5
7	1	1,458	2,25	2,792	3	3,333	4	4,667	5
Mamdani e Moment									
T \ R	1	2	3	4	5	6	7	8	9
1	5	5,579	6	6,421	7	7,338	7,741	8,243	8,917
2	4,273	4,863	5,228	5,59	6,273	6,623	6,962	7,361	8,061
3	3,727	4,41	4,772	5,137	5,727	6,14	6,501	6,874	7,464
4	3	3,579	4	4,421	5	5,579	6	6,421	7
5	2,536	3,126	3,499	3,86	4,273	4,863	5,228	5,59	6,273
6	1,939	2,639	3,038	3,377	3,727	4,41	4,772	5,137	5,727
7	1,083	1,757	2,259	2,662	3	3,579	4	4,421	5
Larsen e MOM									
T \ R	1	2	3	4	5	6	7	8	9
1	5	5	6	7	7	7	8,667	9,5	9,5
2	5	5	6	7	7	7	8,667	9,5	9,5
3	3	3	4	5	5	5	6	7	7
4	3	3	4	5	5	5	6	7	7
5	3	3	4	5	5	5	6	7	7
6	0,5	0,5	1,333	3	3	3	4	5	5
7	0,5	0,5	1,333	3	3	3	4	5	5
Larsen e COA									
T \ R	1	2	3	4	5	6	7	8	9
1	5	5,22	6	6,78	7	7,22	8	8,78	8
2	4,633	4,903	5,307	6,263	6,633	6,903	7,307	8,263	8,633
3	3,367	3,737	4,693	5,097	5,367	5,737	6,693	7,097	7,367
4	3	3,22	4	4,78	5	5,22	6	6,78	7
5	2,633	2,903	3,307	4,263	4,633	4,903	5,307	6,263	6,633
6	1,367	1,737	2,693	3,097	3,367	3,737	4,693	5,097	5,367
7	1	1,22	2	2,78	3	3,22	4	4,78	5
Larsen e Moment									
T \ R	1	2	3	4	5	6	7	8	9
1	5	5,431	6	6,569	7	7,408	7,952	8,5	8,917
2	4,389	4,859	5,284	5,758	6,389	6,839	7,247	7,706	8,326
3	3,611	4,242	4,716	5,141	5,611	6,222	6,691	7,115	7,58
4	3	3,431	4	4,569	5	5,431	6	6,569	7
5	2,42	2,885	3,309	3,778	4,389	4,859	5,284	5,758	6,389
6	1,674	2,294	2,753	3,161	3,611	4,242	4,716	5,141	5,611
7	1,083	1,5	2,048	2,592	3	3,431	4	4,569	5

tempo limitado. Tanto para atrações com relevância alta e tempo de visitaçoão alto quanto para relevância baixa e tempo de visitaçoão baixo, os valores de satisfaçoão devem ser medianos, porque somente uma das característicás é satisfaíta.



**Figura 6.2:** Gráficos referentes aos resultados de HyFOM/FT-FIS.

A maior diferença entre os resultados de HyFOM e FT-FIS ocorre em virtude do método de defuzzificação adotado. MOM apresenta resultados menos precisos, em que algumas atrações com mesmos valores de relevância, porém com valores de tempo de visita diferentes, possuem os mesmos valores de satisfação, quando deveriam ser distintos. O mesmo ocorre quando se fixa o tempo de visita e a relevância é variada. Graficamente, esses resultados correspondem aos platôs presentes nos gráficos da Figura 6.2 para a defuzzificação MOM. A perda de informação ocorre porque MOM considera somente o intervalo de grau máximo do conjunto *fuzzy* resultante do raciocínio das regras *fuzzy* para gerar um valor numérico.

Por outro lado, a defuzzificação por COA e *Moment* conseguem expressar de modo mais preciso a relação definida pelas regras *fuzzy*, ilustrada graficamente como um plano com menos platôs na Figura 6.2. Como esses métodos de defuzzificação baseiam-se no formato do conjunto *fuzzy* resultante da inferência das regras *fuzzy*, há menor perda de informação ao gerar a saída

numérica. Larsen e Mamdani combinados com defuzificação *Moment* proporcionaram os resultados mais precisos, representados graficamente por um plano mais uniforme que as demais combinações de método de inferência e defuzificação disponibilizados pelas implementações de HyFOM e FT-FIS.

Para fuzzyDL com regras implicativas, foram submetidas consultas pela satisfazibilidade mínima e máxima com relação ao conceito *InterestingAttraction*. Esse conceito foi considerado por ser definido em função de valores de satisfação, um dos requisitos para que a consulta *show-concrete-fillers-for* retorne os valores de satisfação:

```
(define-concept InterestingAttraction
  (and touristicAttraction (some satisfaction veryHighSatisfaction)))

(min-sat? InterestingAttraction attraction1 )
(max-sat? InterestingAttraction attraction1 )
(show-concrete-fillers-for attraction1 satisfaction)
```

Para os testes com fuzzyDL e regras *fuzzy* segundo a semântica de Mamdani, representadas pelo conceito *RuleSet*, foram realizadas consultas para cada método de defuzificação (SOM, MOM, LOM) e para satisfazibilidade mínima e máxima:

```
(defuzzify-som? RuleSet attraction1 satisfaction )
(defuzzify-mom? RuleSet attraction1 satisfaction )
(defuzzify-lom? RuleSet attraction1 satisfaction )

(min-sat? RuleSet attraction1 )
(max-sat? RuleSet attraction1 )
(show-concrete-fillers-for attraction1 satisfaction)
```

Os resultados de fuzzyDL são apresentados na Tabela 6.6 e na Figura 6.3. Com relação aos resultados de fuzzyDL com implicações *fuzzy*, as consultas *max-sat* e *min-sat* retornam os mesmos valores de satisfação, por isso somente um resultado é exibido na Tabela 6.6. Os valores inferidos de satisfação seguem a tendência esperada para a recomendação das atrações, porém existem diversas situações em que as consultas não retornam resposta porque a base de conhecimento da ontologia *fuzzy* encontra-se inconsistente ao verificar a satisfazibilidade.

A inconsistência deve-se ao fato de que, por definição, as regras implicativas são baseadas em operadores de implicação e combinadas por conjunção, de modo que cada regra define uma restrição *fuzzy* sobre o conjunto dos valores possíveis (GUILLAUME; CHARNOMORDIC, 2012). Assim, quando os consequentes das implicações *fuzzy* ativadas pelas entradas não possuem uma interseção, a base de conhecimento encontra-se em um estado inconsistente, pois não existe uma interpretação que satisfaça às implicações *fuzzy* e aos valores de relevância e tempo de visitaç o considerados. No caso das regras *fuzzy* da Tabela 6.3, qualquer atraç o que tenha valores de relevância no intervalo (1,5) ou (5,9) e de tempo de visitaç o no intervalo (1,4)



**Tabela 6.6: Resultados de fuzzyDL para consulta 1 e entradas numéricas.**

fuzzyDL com implicações fuzzy									
T \ R	1	2	3	4	5	6	7	8	9
1	5	5,5	6	6,5	7	7,5	8	8,5	9
2	4,3333	inconsist.	inconsist.	inconsist.	6,3333	inconsist.	inconsist.	inconsist.	8,3333
3	3,6667	inconsist.	inconsist.	inconsist.	5,6667	inconsist.	inconsist.	inconsist.	7,6667
4	3	3,5	4	4,5	5	5,5	6	6,5	7
5	2,3333	inconsist.	inconsist.	inconsist.	4,3333	inconsist.	inconsist.	inconsist.	6,3333
6	1,6667	inconsist.	inconsist.	inconsist.	3,6667	inconsist.	inconsist.	inconsist.	5,6667
7	1	1,5	2	2,5	3	3,5	4	4,5	5
fuzzyDL e SOM									
T \ R	1	2	3	4	5	6	7	8	9
1	5	4,5	4	6,5	7	6,5	6	8,5	9
2	inconsist.	inconsist.	4	inconsist.	inconsist.	inconsist.	6	inconsist.	inconsist.
3	inconsist.	inconsist.	2	inconsist.	inconsist.	inconsist.	4	inconsist.	inconsist.
4	3	2,5	2	4,5	5	4,5	4	6,5	7
5	inconsist.	inconsist.	2	inconsist.	inconsist.	inconsist.	4	inconsist.	inconsist.
6	inconsist.	inconsist.	0	inconsist.	inconsist.	inconsist.	2	inconsist.	inconsist.
7	0	0	0	2,5	3	2,5	2	4,5	5
fuzzyDL e MOM									
T \ R	1	2	3	4	5	6	7	8	9
1	5	5	6	7	7	7	8	9,25	9,5
2	inconsist.	inconsist.	6	inconsist.	inconsist.	inconsist.	8	inconsist.	inconsist.
3	inconsist.	inconsist.	4	inconsist.	inconsist.	inconsist.	6	inconsist.	inconsist.
4	3	3	4	5	5	5	6	7	7
5	inconsist.	inconsist.	4	inconsist.	inconsist.	inconsist.	6	inconsist.	inconsist.
6	inconsist.	inconsist.	2	inconsist.	inconsist.	inconsist.	4	inconsist.	inconsist.
7	0,5	0,75	2	3	3	3	4	5	5
fuzzyDL e LOM									
T \ R	1	2	3	4	5	6	7	8	9
1	5	5,5	8	7,5	7	7,5	10	10	10
2	inconsist.	inconsist.	8	inconsist.	inconsist.	inconsist.	10	inconsist.	inconsist.
3	inconsist.	inconsist.	6	inconsist.	inconsist.	inconsist.	8	inconsist.	inconsist.
4	3	3,5	6	5,5	5	5,5	8	7,5	7
5	inconsist.	inconsist.	6	inconsist.	inconsist.	inconsist.	8	inconsist.	inconsist.
6	inconsist.	inconsist.	4	inconsist.	inconsist.	inconsist.	6	inconsist.	inconsist.
7	1	1,5	4	3,5	3	3,5	6	5,5	5
fuzzyDL e max-sat(RuleSet)									
T \ R	1	2	3	4	5	6	7	8	9
1	5 (1)	5 (0,75)	7 (0,5)	7 (0,75)	7 (1)	7 (0,75)	9 (0,5)	9 (0,75)	9 (1)
2	5 (0,667)	5 (0,667)	5 (0,5)	7 (0,667)	7 (0,667)	7 (0,667)	9 (0,5)	9 (0,667)	9 (0,667)
3	3 (0,667)	3 (0,667)	3 (0,5)	5 (0,667)	5 (0,667)	5 (0,667)	7 (0,5)	7 (0,667)	7 (0,667)
4	3 (1)	3 (0,75)	3 (0,5)	5 (0,75)	5 (1)	5 (0,75)	5 (0,5)	7 (0,75)	7 (1)
5	3 (0,667)	3 (0,667)	3 (0,5)	5 (0,667)	5 (0,667)	5 (0,667)	7 (0,5)	7 (0,667)	7 (0,667)
6	1 (0,667)	1 (0,667)	3 (0,5)	3 (0,667)	3 (0,667)	3 (0,667)	5 (0,5)	5 (0,667)	5 (0,667)
7	1 (1)	1 (0,75)	3 (0,5)	3 (0,75)	3 (1)	2,5 (0,75)	5 (0,5)	5 (0,75)	5 (1)

ou (4,7) resulta em inconsistência porque a interseção entre os consequentes das implicações é vazia.

Note que os valores de relevância e tempo de visita são perfeitamente admissíveis e o conjunto de regras *fuzzy* é apropriado, uma vez que os modelos de Mamdani e Larsen produzem resultados pertinentes, conforme discutido anteriormente. Portanto, neste contexto, a semântica mais restritiva das regras implicativas conduziu às situações de inconsistência. A ausência de resultados, em função da inconsistência, não é apropriada para uma aplicação de recomendação, que geralmente demanda soluções agregadas que equilibrem as saídas das regras individuais

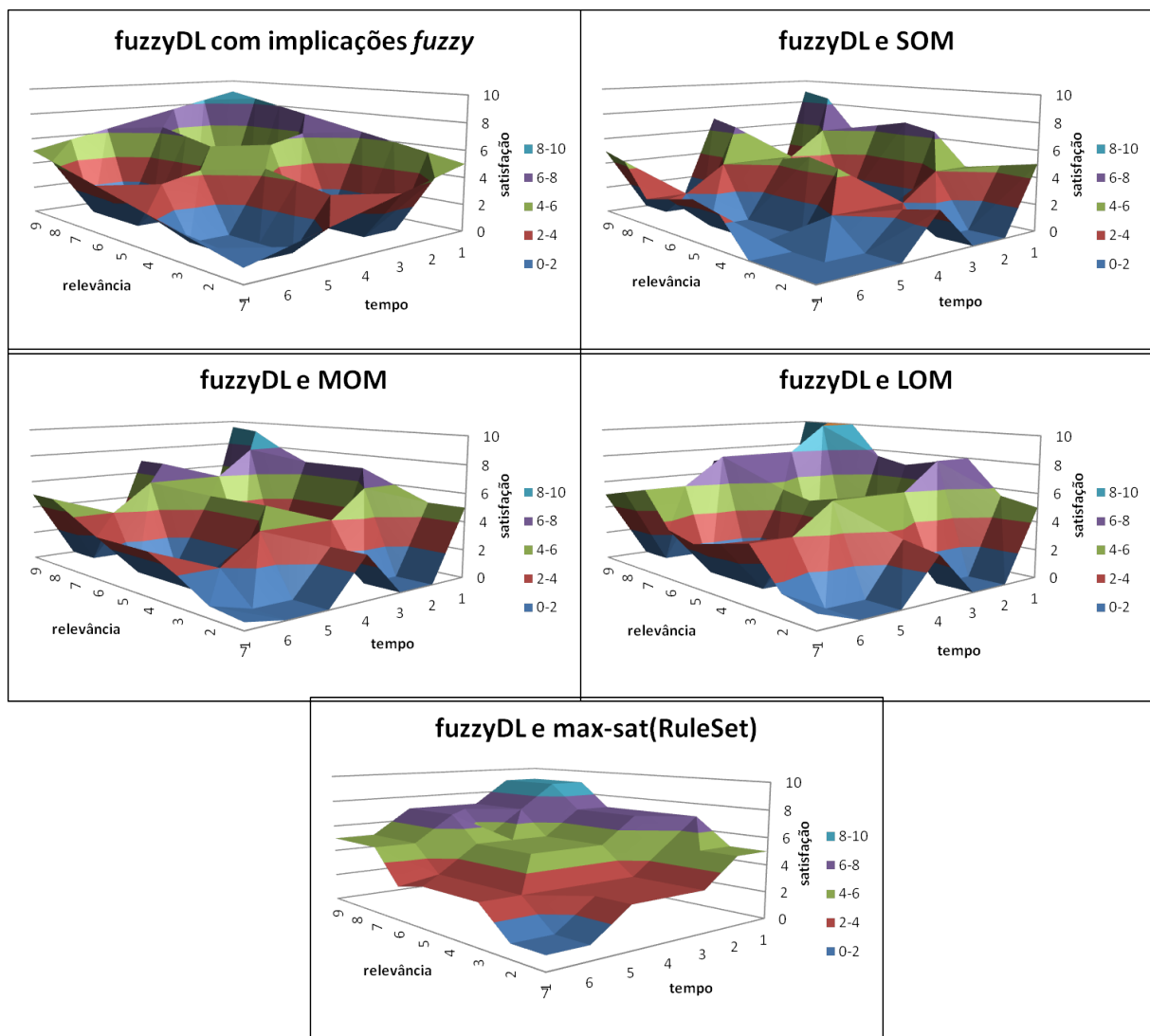


Figura 6.3: Gráficos com os resultados de fuzzyDL.

para apoiar a tomada de decisão.

Os resultados de fuzzyDL com regras de Mamdani e defuzificação SOM, MOM e LOM possuem as mesmas limitações que os resultados de HyFOM/FT-FIS e MOM: como a defuzificação considera somente o intervalo de máximo grau, perde-se informação relevante sobre o formato do conjunto *fuzzy* ao gerar o valor de satisfação.

Além disso, alguns resultados de fuzzyDL com regras de Mamdani e defuzificação SOM, MOM e LOM não foram obtidos em função de inconsistência da base de conhecimento. Ao contrário das regras implicativas, neste caso as situações de inconsistência não são esperadas, tanto que HyFOM e FT-FIS são capazes de inferir a satisfação para todas as combinações de relevância e tempo de visitaç o. Alguns experimentos adicionais foram realizados, com uma granularidade mais fina de valores de entrada, por m as situa es de inconsist ncia persistiram, possivelmente vinculadas a algum problema de precis o do software MILP utilizado

por fuzzyDL. Essa limitação não ocorre com HyFOM e FT-FIS, pois o raciocínio baseado em regras *fuzzy* sempre retorna resultados, um dos requisitos do SIF a ser utilizado em HyFOM e FT-FIS. Em geral, nos gráficos da Figura 6.3, as regiões que correspondem a depressões ilustram justamente essas situações de inconsistência, que também não são apropriadas para o contexto de sistemas de recomendação, conforme discutido anteriormente.

Para o teste com fuzzyDL, regras de Mamdani e consulta por satisfazibilidade mínima, não foi possível obter os resultados, pois a consulta *show-concrete-fillers-for* não retornou os valores de satisfação. A Tabela 6.6 exibe os resultados da consulta por satisfazibilidade máxima (*max-sat*) e *show-concrete-fillers-for*, com o grau da satisfazibilidade máxima descrito entre parênteses. O gráfico dos valores de satisfação obtidos (Figura 6.3) é similar ao gráfico de HyFOM/FT-FIS com a defuzzificação MOM, portanto possui as mesmas limitações. No entanto, a informação adicional sobre os graus de verdade pode ser considerada para tratar a ambiguidade dos valores de satisfação inferidos, embora alguns resultados permaneçam ambíguos como, por exemplo, as atrações com relevância igual a 3 e tempo de visita variando no intervalo [3, 7].

De modo geral, as respostas mais apropriadas para a consulta 1, que leva em consideração os valores inferidos de satisfação para classificar as atrações turísticas, foram obtidas por HyFOM e FT-FIS em comparação com os resultados providos por fuzzyDL. Especificamente, a combinação de métodos de Mamdani e Larsen com a defuzzificação *Moment* proporcionaram uma aproximação mais expressiva da relação estabelecida pelas regras *fuzzy* consideradas nesta aplicação.

### **Consulta 2: saída de SIF restrita por predicado *fuzzy***

Outro exemplo de consulta trata do uso da saída do SIF em definições de conceitos na ontologia *fuzzy*, ilustrando a integração  $SIF \rightarrow \text{ontologia fuzzy}$ . Especificamente, a consulta refere-se ao conceito abstrato *AboveAverageAttraction*, definido em função dos valores de satisfação restritos pelo predicado *fuzzy aboveAverageSatisfaction*, representado por um conjunto *fuzzy right-shoulder(3,5)*. Esse conceito pode ser considerado para recomendar atrações que estejam acima da média em termos de satisfação, sendo que os valores de satisfação inferidos podem ser usados para classificar as respostas.

Em HyFOM, os valores defuzzificados de satisfação provenientes do SIF são representados na ontologia *fuzzy* como asserções de propriedades concretas, sob a supervisão de especialistas. Desta forma, a ontologia *fuzzy* contém os valores numéricos de satisfação exibidos na Tabela 6.5, dependendo do método de raciocínio e defuzzificação considerados no SIF. Neste caso,

para obter as atrações acima da média, é realizada uma consulta pela satisfazibilidade máxima do conceito abstrato *AboveAverageAttraction* para cada atração turística. O grau retornado pela consulta representa a pertinência da atração ao conceito *AboveAverageAttraction*, obtida por meio da verificação do valor de satisfação inferido pelo SIF no conjunto *fuzzy* referente ao predicado *fuzzy aboveAverageSatisfaction*. Assim, o grau de pertinência pode ser usado para classificar as respostas. Os resultados apresentados na Tabela 6.7, em termos de grau de pertinência a *AboveAverageAttraction*, mostram que as combinações de métodos de raciocínio e defuzificação envolvendo Mamdani, Larsen, COA e *Moment* proveem os resultados mais expressivos, com uma transição gradual de graus de pertinência para a recomendação das atrações acima da média.

Em FT-FIS, definiu-se o predicado *fuzzy aboveAverage* a partir da conjunção de *fuzzyRuleReasoning* e *aboveAverageSatisfaction* para modelar o conceito *AboveAverageAttraction*:

```
(define-fuzzy-predicate aboveAverageSatisfaction(x) *real* right-shoulder(x, 0.0, 10.0, 3.0, 5.0) )
(define-fuzzy-predicate aboveAverage(v0,v1,v2) *real*
  (and fuzzyRuleReasoning("RuleBase1";v0,v1;v2) aboveAverageSatisfaction(v2)))

(define-concept AboveAverageAttraction
  (and TouristicAttraction (some relevance,timing,satisfaction.aboveAverage)))
```

A consulta submetida a FT-FIS foi a verificação de satisfazibilidade para

$$\psi = \langle x : AboveAverageAttraction > 0.0 \rangle$$

com defuzificação dos valores de satisfação inferidos pelo SIF para todas instâncias  $x$  de atrações turísticas. Assim como na consulta 1, o grau de verdade considerado na consulta ( $> 0.0$ ) foi utilizado para que a saída *fuzzy* do SIF não fosse restrita por um grau específico, ou seja, as saídas do SIF são consideradas de forma direta, sem restrições de grau. Também foram utilizados os métodos de defuzificação MOM, COA e *Moment*.

Note que, diferentemente de HyFOM, o resultado do predicado *fuzzyRuleReasoning* é um conjunto *fuzzy* obtido por meio do raciocínio do SIF, que deve ser restrito pela conjunção com *aboveAverageSatisfaction*. Desta forma, a conjunção é realizada entre conjuntos *fuzzy* e não entre um valor defuzificado e um conjunto *fuzzy*. Com isso, evita-se perda de informação ao considerar o resultado do SIF como um valor defuzificado antes de verificar o predicado *aboveAverageSatisfaction*. Em FT-FIS, a defuzificação é realizada somente ao final do processo de inferência do *tableau fuzzy*, quando todas as restrições foram aplicadas. Os valores de satisfação obtidos por FT-FIS são descritos na Tabela 6.8 e graficamente na Figura 6.4.

Dentre os resultados de FT-FIS, o único que gerou inconsistência é a atração com relevância igual a 1 e tempo de visitaç o igual a 7, porque o conjunto *fuzzy* resultante do racioc nio do

Tabela 6.7: Graus de pertinência obtidos a partir da satisfação inferida por HyFOM para consulta 2 e entradas numéricas.

Mamdani e MOM									
T \ R	1	2	3	4	5	6	7	8	9
1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1
3	0	0	0,5	1	1	1	1	1	1
4	0	0	0,5	1	1	1	1	1	1
5	0	0	0,5	1	1	1	1	1	1
6	0	0	0	0	0	0	0,5	1	1
7	0	0	0	0	0	0	0,5	1	1
Mamdani e COA									
T \ R	1	2	3	4	5	6	7	8	9
1	1	1	1	1	1	1	1	1	1
2	0,75	0,938	1	1	1	1	1	1	1
3	0,25	0,447	0,834	1	1	1	1	1	1
4	0	0,166	0,5	0,834	1	1	1	1	1
5	0	0,021	0,278	0,632	0,75	0,938	1	1	1
6	0	0	0	0,133	0,25	0,447	0,834	1	1
7	0	0	0	0	0	0,166	0,5	0,834	1
Mamdani e Moment									
T \ R	1	2	3	4	5	6	7	8	9
1	1	1	1	1	1	1	1	1	1
2	0,636	0,932	1	1	1	1	1	1	1
3	0,364	0,705	0,886	1	1	1	1	1	1
4	0	0,29	0,5	0,71	1	1	1	1	1
5	0	0,063	0,25	0,43	0,636	0,932	1	1	1
6	0	0	0,019	0,188	0,364	0,705	0,886	1	1
7	0	0	0	0	0	0,29	0,5	0,71	1
Larsen e MOM									
T \ R	1	2	3	4	5	6	7	8	9
1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1
3	0	0	0,5	1	1	1	1	1	1
4	0	0	0,5	1	1	1	1	1	1
5	0	0	0,5	1	1	1	1	1	1
6	0	0	0	0	0	0	0,5	1	1
7	0	0	0	0	0	0	0,5	1	1
Larsen e COA									
T \ R	1	2	3	4	5	6	7	8	9
1	1	1	1	1	1	1	1	1	1
2	0,816	0,952	1	1	1	1	1	1	1
3	0,184	0,368	0,846	1	1	1	1	1	1
4	0	0,11	0,5	0,89	1	1	1	1	1
5	0	0	0,154	0,632	0,816	0,952	1	1	1
6	0	0	0	0,048	0,184	0,368	0,846	1	1
7	0	0	0	0	0	0,11	0,5	0,89	1
Larsen e Moment									
T \ R	1	2	3	4	5	6	7	8	9
1	1	1	1	1	1	1	1	1	1
2	0,694	0,93	1	1	1	1	1	1	1
3	0,306	0,621	0,858	1	1	1	1	1	1
4	0	0,216	0,5	0,784	1	1	1	1	1
5	0	0	0,154	0,389	0,694	0,93	1	1	1
6	0	0	0	0,08	0,306	0,621	0,858	1	1
7	0	0	0	0	0	0,216	0,5	0,784	1

SIF não possui uma interseção com o predicado *fuzzy aboveAverageSatisfaction*. Desta forma, essa atração não deve ser considerada como uma atração acima da média, porque de fato ela

Tabela 6.8: Resultados de FT-FIS para consulta 2 e entradas numéricas.

Mamdani e MOM									
T \ R	1	2	3	4	5	6	7	8	9
1	5	5	6	7	7	7	8	9,25	9,5
2	5	5	6	7	7	7	8	9,167	9,167
3	4	4	5	5	5	5	6	7	7
4	4	4	5	5	5	5	6	7	7
5	4	4	5	5	5	5	6	7	7
6	4	4	4	4	4	4	5	5	5
7	inconsist.	4	4	4	4	4	5	5	5
Mamdani e COA									
T \ R	1	2	3	4	5	6	7	8	9
1	5	5,333	6	6,667	7	7,208	7,75	8,542	8
2	5	5,375	6	6,5	6,5	6,734	7,083	7,882	8
3	4,917	5,667	5,667	5,5	5,5	5,736	6,444	6,958	7,333
4	4	4,75	5	5	5	5,333	6	6,667	7
5	4	4,75	5	5	5	5,375	6	6,5	6,5
6	4	4,861	4,917	4,917	4,917	5,667	5,667	5,5	5,5
7	inconsist.	4	4	4	4	4,75	5	5	5
Mamdani e Moment									
T \ R	1	2	3	4	5	6	7	8	9
1	5	5,579	6	6,421	7	7,338	7,741	8,243	8,917
2	5	5,604	6	6,273	6,273	6,623	6,962	7,361	8,061
3	4,952	5,792	5,821	5,727	5,727	6,14	6,501	6,874	7,464
4	4	4,875	5	5	5	5,579	6	6,421	7
5	4	4,875	5	5	5	5,604	6	6,273	6,273
6	4	4,926	4,952	4,952	4,952	5,792	5,821	5,727	5,727
7	inconsist.	4	4	4	4	4,875	5	5	5
Larsen e MOM									
T \ R	1	2	3	4	5	6	7	8	9
1	5	5	6	7	7	7	8,667	9,5	9,5
2	5	5	6	7	7	7	8,667	9,5	9,5
3	3,8	3,8	5	5	5	5	6	7	7
4	4	3,857	5	5	5	5	6	7	7
5	3,8	3,8	5	5	5	5	6	7	7
6	3,5	3,5	5	3,8	3,8	3,8	5	5	5
7	inconsist.	3,4	3,667	3,857	4	3,857	5	5	5
Larsen e COA									
T \ R	1	2	3	4	5	6	7	8	9
1	5	5,22	6	6,78	7	7,22	8	8,78	8
2	4,979	5,231	5,905	6,58	6,633	6,903	7,307	8,263	8,633
3	4,713	5,29	5,429	5,35	5,367	5,737	6,693	7,097	7,367
4	4	4,444	4,915	4,992	5	5,22	6	6,78	7
5	3,905	4,502	4,915	4,979	4,979	5,231	5,905	6,58	6,633
6	3,775	4,773	4,789	4,713	4,713	5,29	5,429	5,35	5,367
7	inconsist.	3,735	3,845	3,931	4	4,444	4,915	4,992	5
Larsen e Moment									
T \ R	1	2	3	4	5	6	7	8	9
1	5	5,431	6	6,569	7	7,408	7,952	8,5	8,917
2	4,967	5,444	5,929	6,296	6,389	6,839	7,247	7,706	8,326
3	4,729	5,517	5,645	5,588	5,611	6,222	6,691	7,115	7,58
4	4	4,605	4,889	4,986	5	5,431	6	6,569	7
5	3,933	4,621	4,889	4,967	4,967	5,444	5,929	6,296	6,389
6	3,833	4,744	4,77	4,729	4,729	5,517	5,645	5,588	5,611
7	inconsist.	3,8	3,889	3,952	4	4,605	4,889	4,986	5

representa o pior caso dentre as combinações de relevância e tempo de visitaç o.

Para o restante das atra es, os valores de satisfa o inferidos em FT-FIS correspondem   defuzifica o aplicada sobre o conjunto *fuzzy* resultante da conjun o da sa da *fuzzy* do SIF com

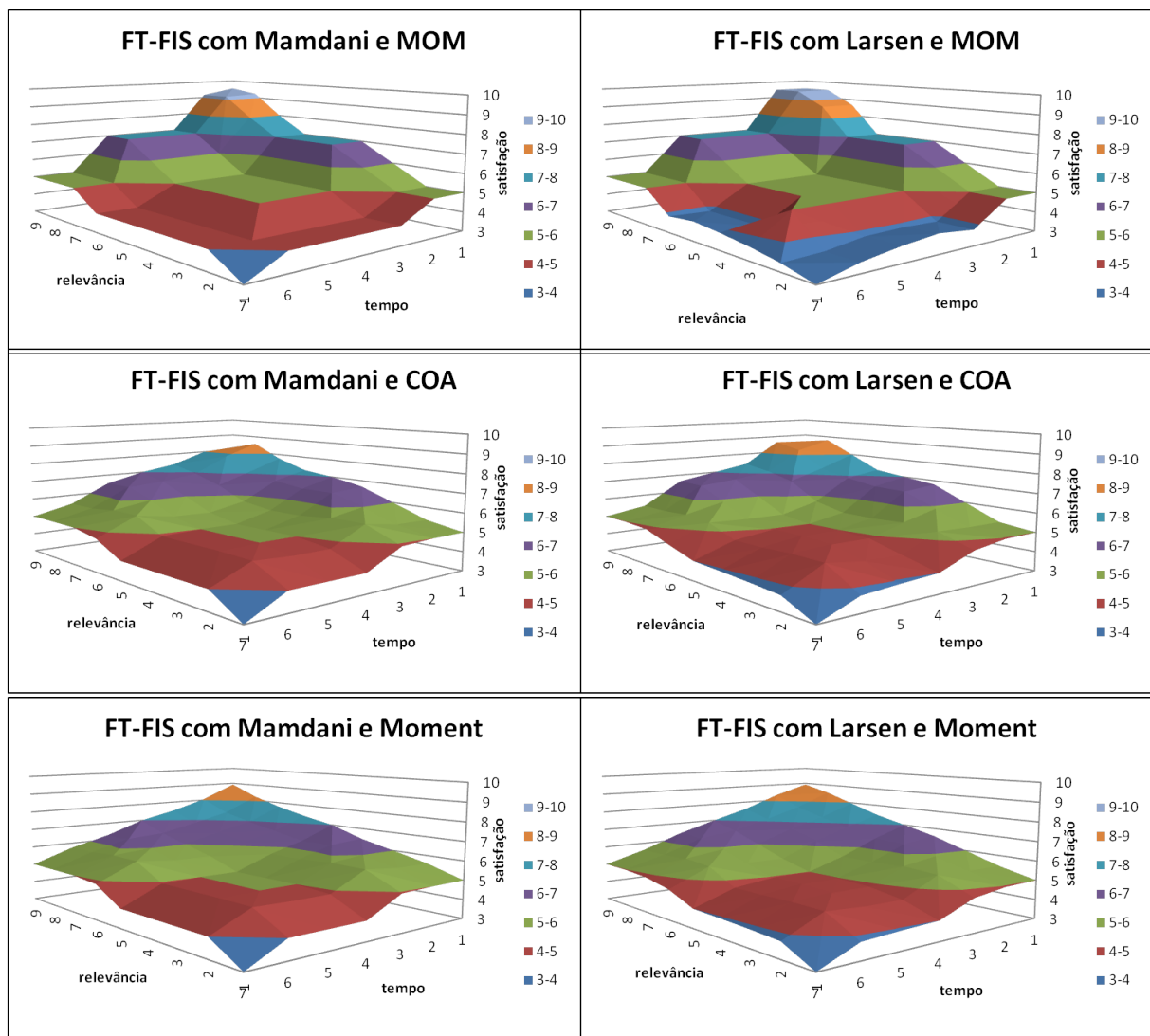


Figura 6.4: Gráficos referentes aos resultados de FT-FIS.

o predicado *aboveAverageSatisfaction*, definido na ontologia *fuzzy*. Portanto, nesta consulta, o resultado do SIF é restrito por um predicado presente na ontologia, que deve ser considerado para definir o valor de satisfação de uma atração. Com relação aos valores inferidos, pode-se notar que os resultados seguem a mesma tendência da consulta 1, em que Larsen apresenta resultados um mais precisos que Mamdani, por expressar a relação definida pelas regras *fuzzy*: quanto maior a relevância e menor o tempo de visitaç o, maior a satisfa o. O m todo de Larsen associado   defuzifica o *Moment* proporciona os resultados mais aproximados com rela o   sem ntica estabelecida pelas regras, com o gr fico mais pr ximo de um plano uniforme.

Com rela o aos testes com *fuzzyDL* e regras *fuzzy* segundo a sem ntica de Mamdani, o conceito *AboveAverageAttraction* foi definido a partir da conjun o entre o conceito *RuleSet*, que representa as regras *fuzzy*, e o conceito referente   restri o dos valores de satisfa o pelo predicado *fuzzy aboveAverageSatisfaction*. Foram realizadas consultas para cada m todo de

defuzzificação (SOM, MOM, LOM) e para satisfazibilidade mínima e máxima:

```
(define-concept AboveAverageAttraction
  (and RuleSet (some satisfaction aboveAverageSatisfaction)))

(defuzzify-som? AboveAverageAttraction attraction1 satisfaction )
(defuzzify-mom? AboveAverageAttraction attraction1 satisfaction )
(defuzzify-lom? AboveAverageAttraction attraction1 satisfaction )

(min-sat? AboveAverageAttraction attraction1 )
(max-sat? AboveAverageAttraction attraction1 )
(show-concrete-fillers-for attraction1 satisfaction)
```

Os resultados de fuzzyDL com defuzzificação SOM, MOM e LOM, apresentados na Tabela 6.9 e na Figura 6.5, são menos precisos em função da defuzzificação baseada no intervalo de máximo grau, com perda de informação com relação aos valores de satisfação inferidos. Além disso, permanecem as situações de inconsistência, mesmo quando FT-FIS é capaz de gerar resultados pertinentes.

Quanto às consultas por satisfazibilidade, a consulta por satisfazibilidade mínima e *show-concrete-fillers-for* não retornou os valores de satisfação, portanto não foi possível obter os resultados, assim como na consulta 1. Para a satisfazibilidade máxima, os valores de satisfação e os graus retornados são apresentados na Tabela 6.9. De forma geral, os resultados possuem características similares aos resultados obtidos por FT-FIS com defuzzificação MOM, que é possível notar pela semelhança dos gráficos correspondentes. Mesmo que o grau de satisfazibilidade máxima seja utilizado para tratar a ambiguidade das respostas, ainda assim algumas permanecem ambíguas, como no caso das atrações com relevância igual a 3 e tempo de visita variando no intervalo [1,5].

Para fuzzyDL com regras implicativas, foram submetidas consultas pela satisfazibilidade mínima e máxima com relação ao conceito *AboveAverageAttraction*:

```
(define-fuzzy-concept aboveAverageSatisfaction right-shoulder(0.0, 10.0, 3.0, 5.0) )
(define-concept AboveAverageAttraction
  (and touristicAttraction (some satisfaction aboveAverageSatisfaction)))

(min-sat? AboveAverageAttraction attraction1 )
(max-sat? AboveAverageAttraction attraction1 )
(show-concrete-fillers-for attraction1 satisfaction)
```

Entretanto, os testes com fuzzyDL e regras implicativas retornam os mesmos resultados da consulta 1, ou seja, a conjunção com o predicado *fuzzy aboveAverageSatisfaction* não é considerada para inferir os valores de satisfação e determinar o grau de pertinência a *AboveAverageAttraction*. Por esta razão, não é possível compará-los com os resultados de FT-FIS, que foram gerados com base na conjunção com o predicado *fuzzy* definido na ontologia.



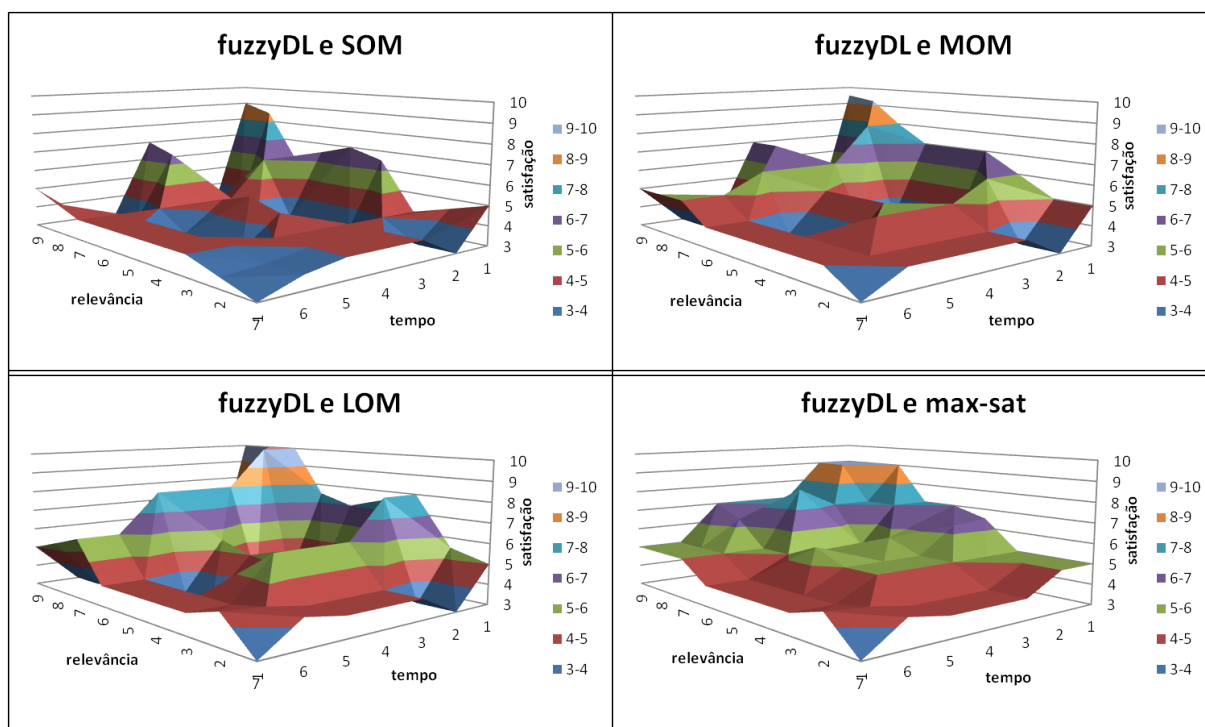
**Tabela 6.9: Resultados de fuzzyDL para consulta 2 e entradas numéricas.**

fuzzyDL e SOM									
R \ T	1	2	3	4	5	6	7	8	9
1	5	4,5	4	6,5	7	6,5	6	8,5	9
2	inconsist.	inconsist.	4	inconsist.	inconsist.	inconsist.	6	inconsist.	inconsist.
3	4	4	4	inconsist.	inconsist.	inconsist.	4	inconsist.	inconsist.
4	4	4	4	4,5	5	4,5	4	6,5	7
5	4	4	4	inconsist.	inconsist.	inconsist.	4	inconsist.	inconsist.
6	3,667	3,667	4	4	4	4	4	inconsist.	inconsist.
7	inconsist.	3,5	4	4	4	4	4	4,5	5
fuzzyDL e MOM									
R \ T	1	2	3	4	5	6	7	8	9
1	5	5	6	7	7	7	8	9,25	9,5
2	inconsist.	inconsist.	6	inconsist.	inconsist.	inconsist.	8	inconsist.	inconsist.
3	4	4	5	inconsist.	inconsist.	inconsist.	6	inconsist.	inconsist.
4	4	4	5	5	5	5	6	7	7
5	4	4	5	inconsist.	inconsist.	inconsist.	6	inconsist.	inconsist.
6	4	4	4	4	4	4	5	inconsist.	inconsist.
7	inconsist.	4	4	4	4	4	5	5	5
fuzzyDL e LOM									
R \ T	1	2	3	4	5	6	7	8	9
1	5	5,5	8	7,5	7	7,5	10	10	10
2	inconsist.	inconsist.	8	inconsist.	inconsist.	inconsist.	10	inconsist.	inconsist.
3	4	4	6	inconsist.	inconsist.	inconsist.	8	inconsist.	inconsist.
4	4	4	6	5,5	5	5,5	8	7,5	7
5	4	4	6	inconsist.	inconsist.	inconsist.	8	inconsist.	inconsist.
6	4,333	4,333	4	4	4	4	6	inconsist.	inconsist.
7	inconsist.	4,5	4	4	4	4	6	5,5	5
fuzzyDL e max-sat(AboveAverageAttraction)									
R \ T	1	2	3	4	5	6	7	8	9
1	5 (1)	5 (0,75)	5 (0,5)	6,5 (0,75)	7 (1)	7 (0,75)	9 (0,5)	9 (0,75)	9 (1)
2	5 (0,667)	5 (0,667)	5 (0,5)	6,33 (0,667)	6,33 (0,667)	7 (0,667)	7 (0,5)	9 (0,667)	9 (0,667)
3	4 (0,5)	4 (0,5)	5 (0,5)	5 (0,667)	5,667 (0,667)	5 (0,667)	7 (0,5)	7 (0,667)	7 (0,667)
4	4 (0,5)	4 (0,5)	5 (0,5)	5 (0,75)	5 (1)	5 (0,75)	8 (0,5)	7 (0,75)	7 (1)
5	4 (0,5)	4 (0,5)	5 (0,5)	5 (0,667)	5,667 (0,667)	5 (0,667)	5 (0,5)	6,33 (0,667)	7 (0,667)
6	4,33 (0,33)	4,33 (0,33)	4 (0,5)	4 (0,5)	4 (0,5)	4 (0,5)	5 (0,5)	5 (0,667)	5 (0,667)
7	inconsist.	4,5 (0,25)	4 (0,5)	4 (0,5)	4 (0,5)	4 (0,5)	5 (0,5)	5 (0,75)	5 (1)

Portanto, de forma geral, comparando os resultados de FT-FIS e das abordagens de fuzzyDL na consulta 2, os resultados mais pertinentes à semântica das regras *fuzzy* e à restrição realizada na definição do conceito *AboveAverageAttraction* foram obtidos pela combinação do método de Larsen com a defuzificação *Moment*, disponibilizados pela implementação de FT-FIS.

### 6.2.3 Experimentos com Entradas *Fuzzy*

O objetivo dos experimentos com entradas *fuzzy* é mostrar que inferências da ontologia *fuzzy* podem definir as entradas para o SIF, considerando a integração *ontologia fuzzy*  $\rightarrow$  *SIF*. Além disso, os testes demonstram que HyFOM e FT-FIS são capazes de tratar não somente valores numéricos mas também conjuntos *fuzzy* como entradas para o SIF. Este recurso é interessante para modelar conceitos que representam percepções subjetivas, em que não é possível definir com exatidão os limites de valores associados a uma propriedade concreta numérica.



**Figura 6.5:** Gráficos com os resultados de fuzzyDL para consulta 2 e entradas numéricas.

Para os testes com entradas *fuzzy*, foram representadas na ontologia *fuzzy* (Tabelas 6.1 e 6.2) categorias de atrações como Museu (*Museum*), Parque (*Park*), local religioso (*ReligiousSite*), entre outros. Como o foco é a recomendação das atrações, essas categorias podem ser modeladas para expressar preferências com relação à relevância e ao tempo de visitaç o, dependendo do perfil dos turistas. Por exemplo, o perfil de turista exemplificado na ontologia *fuzzy* representa turistas que preferem visitar parques e locais esportivos, tendo em vista as características associadas a essas categorias. Vale ressaltar que representar as prefer ncias de forma imprecisa, por meio de termos lingu sticos (alto, m dio, baixo),   mais natural e pr ximo da subjetividade humana que utilizar intervalos num ricos r gidos que podem resultar em perda de informa o por n o tratarem transi es graduais entre diferentes estados poss veis.

Neste contexto, as inst ncias de atra es tur sticas s o representadas como indiv duos das categorias de atra es, conforme exemplificam atra es *attraction3* a *attraction6* nas Tabelas 6.1 e 6.2. Foram definidas inst ncias de modo a cobrir todas combina es de valores imprecisos de relev ncia e tempo de visita o representados na ontologia (alto, m dio e baixo). Para cada combina o, foram simulados diferentes valores de grau de pertin ncia m nimo  $n \in \{1.0, 0.8, 0.5, 0.2\}$ , uma vez que a sintaxe de fuzzyDL permite modelar asser es do tipo  $\langle x : C \geq n \rangle$ . Desta forma, foram consideradas atra es com diferentes graus de pertin ncia  s categorias de atra es, com o intuito de expressar diferentes compatibilidades com rela o   defini o dos conceitos. Assim, atra es com graus mais pr ximos do grau 1.0, representam

maior compatibilidade, enquanto atrações com grau menor expressam uma menor precisão na instanciação, conseqüentemente uma maior flexibilidade na definição dos valores das propriedades de relevância e tempo de visitaç o.

Para inferir os valores de satisfa o para as inst ncias de atra oes tur sticas, os valores de relev ncia e tempo de visita o s o obtidos a partir do processo de racioc nio de ontologias *fuzzy*. No caso de racioc nio baseado em *tableau fuzzy* considerado por FT-FIS, aplicam-se o *unfolding* e regras de expans o para decompor as express es de conceito e ent o definir os valores das propriedades concretas. Por exemplo, ao expandir a asser o  $\langle attraction4 : Park \geq 0.8 \rangle$ , primeiramente   aplicado *unfolding* para expandir o conceito *Park* pela express o de conceito do lado direito da subsun o:

$$Park \sqsubseteq TouristicAttraction \sqcap \exists relevance.highRelevance \sqcap \exists timing.mediumTiming$$

Na sequ ncia, aplicam-se as regras de expans o regra- $\sqcap_{p \triangleright}$  e regra- $\exists_{p \triangleright}$  para ent o adicionar em  $DC(attraction4)$  as restri oes:

$$DC(attraction4) = \{ \langle highRelevance(v_1) \geq 0.8 \rangle, \langle mediumTiming(v_2) \geq 0.8 \rangle \}$$

onde  $v_1$  e  $v_2$  correspondem aos identificadores de n s de tipos de dados referentes aos valores das propriedades concretas *relevance* e *timing* para *attraction4*, respectivamente. A partir do cont do de  $DC(attraction4)$ , s o aplicadas as restri oes de grau ( $\geq 0.8$ ) e ent o definidas as entradas *fuzzy* a serem consideradas pelo SIF para inferir a satisfa o para *attraction4*.

Para os experimentos comparativos com fuzzyDL, utilizou-se a vers o da ontologia *fuzzy* da Tabela 6.1, as regras *fuzzy* da Tabela 6.3 para os testes com implica oes *fuzzy* e as regras da Tabela 6.4 para os testes com regras de Mamdani, defuzifica o e consultas por satisfazibilidade.

Assim como nos experimentos com entradas num ricas, foram realizadas duas consultas dentro do contexto de recomenda o de atra oes tur sticas com base nos valores de satisfa o inferidos por SIF. O objetivo   analisar os resultados das infer ncias de HyFOM, FT-FIS e fuzzyDL, considerando a sem ntica das regras *fuzzy* e a integra o *ontologia fuzzy*  $\rightarrow$  *SIF* e *SIF*  $\rightarrow$  *ontologia fuzzy*.

### Consulta 1: satisfa o inferida a partir de regras *fuzzy*

Os aspectos a serem analisados na consulta 1 s o: (1) se as entradas *fuzzy* s o consideradas para gerar os resultados de satisfa o e (2) se os valores de satisfa o s o coerentes com a

relação estabelecida pelas regras *fuzzy*. Nesta consulta, os valores inferidos a partir das regras *fuzzy* são considerados de forma direta para a recomendação, ou seja, não são restritos por algum predicado *fuzzy* presente na ontologia *fuzzy*.

As consultas submetidas a HyFOM, FT-FIS e fuzzyDL são as mesmas da consulta 1 com entradas numéricas, com a diferença de que as atrações turísticas analisadas são instâncias de categorias de atrações (museu, parque, entre outros) e portanto não possuem os valores de relevância e de tempo de visita representados explicitamente. Eles devem ser obtidos por meio das inferências da ontologia *fuzzy*, conforme discutido anteriormente. Os resultados de satisfação inferidos por HyFOM e FT-FIS são apresentados na Tabela 6.10 e os resultados de fuzzyDL na Tabela 6.11, onde os valores de relevância e tempo de visita são descritos como conjuntos *fuzzy* com restrições de grau que resultam das inferências da ontologia *fuzzy*.

Os resultados de HyFOM e FT-FIS são exibidos de forma unificada porque ambos obtêm as entradas *fuzzy* da ontologia *fuzzy* de forma correta: as restrições de grau presentes na instanciamento são corretamente aplicadas aos valores imprecisos associados às categorias de atrações. Além disso, como a implementação do SIF é a mesma para HyFOM e FT-FIS, os valores de satisfação inferidos não mudam. Na Tabela 6.10, é possível notar que todas combinações de método de raciocínio e defuzzificação são capazes de expressar resultados de satisfação pertinentes com relação à semântica das regras *fuzzy*: quanto maior a relevância e menor o tempo, maior a satisfação.

Entretanto, ao considerar a variação de graus de verdade em uma mesma combinação de entradas *fuzzy*, a defuzzificação MOM não consegue refletir a imprecisão que existe na instanciamento das categorias de atrações. Ao verificar os resultados da inferência do SIF, os conjuntos *fuzzy* possuem formatos distintos para cada restrição de grau, com exceção das restrições  $\geq 0,5$  e  $\geq 0,2$ , que resultam em uma mesma medida de possibilidade  $Poss(A, B) = 0,5$  entre o conjunto *fuzzy*  $A$  de entrada e o conjunto *fuzzy*  $B$  do antecedente das regras. Portanto, essas restrições geram o mesmo conjunto *fuzzy* no consequente, razão pela qual conduzem aos mesmos resultados defuzzificados.

Para as demais restrições de graus, os métodos de defuzzificação COA e *Moment* conseguem expressar com maior precisão os diferentes valores de satisfação, por considerarem o formato do conjunto *fuzzy* e não somente o intervalo de grau máximo como ocorre com MOM. Vale esclarecer que, conforme o grau considerado diminui, mais flexível é a instanciamento da atração com relação a um conceito. Os resultados das inferências do SIF também refletem essa flexibilidade: para entradas cujos valores de satisfação tendem a ser altos, a variação nos graus resulta em valores de satisfação mais flexíveis e, portanto, tendem a diminuir; para entradas com tendência

**Tabela 6.10: Resultados de HyFOM/FT-FIS para consulta 1 e entradas *fuzzy*.**

Entradas <i>fuzzy</i>		Mamdani			Larsen		
relevância	tempo	MOM	COA	Moment	MOM	COA	Moment
alta $\geq 1,0$	baixo $\geq 1,0$	9,5	8	8,917	9,5	8	8,917
alta $\geq 0,8$	baixo $\geq 0,8$	9,5	8,549	7,855	9,5	8,713	8,186
alta $\geq 0,5$	baixo $\geq 0,5$	9,5	7,5	7,208	9,5	8,08	7,523
alta $\geq 0,2$	baixo $\geq 0,2$	9,5	7,5	7,208	9,5	8,08	7,523
alta $\geq 1,0$	médio $\geq 1,0$	7	7	7	7	7	7
alta $\geq 0,8$	médio $\geq 0,8$	7	6,697	6,189	7	6,844	6,518
alta $\geq 0,5$	médio $\geq 0,5$	7	6,225	5,877	7	6,581	6,184
alta $\geq 0,2$	médio $\geq 0,2$	7	6,225	5,877	7	6,581	6,184
alta $\geq 1,0$	alto $\geq 1,0$	5	5	5	5	5	5
alta $\geq 0,8$	alto $\geq 0,8$	5	5	5	5	5	5
alta $\geq 0,5$	alto $\geq 0,5$	5	5	5	5	5	5
alta $\geq 0,2$	alto $\geq 0,2$	5	5	5	5	5	5
média $\geq 1,0$	baixo $\geq 1,0$	7	7	7	7	7	7
média $\geq 0,8$	baixo $\geq 0,8$	7	6,697	6,189	7	6,755	6,476
média $\geq 0,5$	baixo $\geq 0,5$	7	6,225	5,877	7	6,581	6,184
média $\geq 0,2$	baixo $\geq 0,2$	7	6,225	5,877	7	6,581	6,184
média $\geq 1,0$	médio $\geq 1,0$	5	5	5	5	5	5
média $\geq 0,8$	médio $\geq 0,8$	5	5	5	5	4,966	5,029
média $\geq 0,5$	médio $\geq 0,5$	5	5	5	5	5	5
média $\geq 0,2$	médio $\geq 0,2$	5	5	5	5	4,83	4,928
média $\geq 1$	alto $\geq 1$	3	3	3	3	3	3
média $\geq 0,8$	alto $\geq 0,8$	3	3,303	3,811	3	3,211	3,57
média $\geq 0,5$	alto $\geq 0,5$	3	3,775	4,123	3	3,419	3,816
média $\geq 0,2$	alto $\geq 0,2$	3	3,775	4,123	3	3,419	3,816
baixa $\geq 1,0$	baixo $\geq 1,0$	5	5	5	5	5	5
baixa $\geq 0,8$	baixo $\geq 0,8$	5	5	5	5	5	5
baixa $\geq 0,5$	baixo $\geq 0,5$	5	5	5	5	5	5
baixa $\geq 0,2$	baixo $\geq 0,2$	5	5	5	5	5	5
baixa $\geq 1,0$	médio $\geq 1,0$	3	3	3	3	3	3
baixa $\geq 0,8$	médio $\geq 0,8$	3	3,303	3,811	3	3,211	3,57
baixa $\geq 0,5$	médio $\geq 0,5$	3	3,775	4,123	3	3,419	3,816
baixa $\geq 0,2$	médio $\geq 0,2$	3	3,775	4,123	3	3,419	3,816
baixa $\geq 1,0$	alto $\geq 1,0$	0,5	1	1,083	0,5	1	1,083
baixa $\geq 0,8$	alto $\geq 0,8$	0,5	1,451	2,145	0,5	1,347	1,897
baixa $\geq 0,5$	alto $\geq 0,5$	0,5	2,5	2,792	0,5	1,92	2,477
baixa $\geq 0,2$	alto $\geq 0,2$	0,5	2,5	2,792	0,5	1,92	2,477

a satisfação baixa, os graus distintos conduzem a um pequeno aumento da satisfação. Como há perda de informação com MOM, a recomendação de atrações pode ser menos apropriada, pois os valores de satisfação devem expressar a imprecisão que existe na instanciação dos conceitos da ontologia *fuzzy*.

Com relação aos resultados providos por fuzzyDL (Tabela 6.11), existem diversas limitações a serem discutidas. Primeiramente, os valores de satisfação inferidos por meio das implicações *fuzzy* não correspondem à relação estabelecida pelas regras *fuzzy*. Por exemplo, os maiores valores de satisfação, que deveriam ser atribuídos às entradas *fuzzy* de relevância alta e tempo de visitaçao baixo, são inferidos para as atrações com relevância média. Além disso, os valores mais baixos de satisfação não são inferidos para as atrações com relevância baixa e tempo de visitaçao alto. Possivelmente os resultados incoerentes ocorrem porque as entradas *fuzzy* não são consideradas pelas regras implicativas para gerar os valores de satisfação. Trabalhos futuros devem investigar de forma mais detalhada os motivos pelos quais alguns resultados de fuzzyDL não são coerentes com os resultados esperados.

**Tabela 6.11: Resultados de fuzzyDL para consulta 1 e entradas *fuzzy*.**

Entradas <i>fuzzy</i>		fuzzyDL com implicações <i>fuzzy</i>		fuzzyDL com consulta defuzzify			fuzzyDL com max-sat(RuleSet)	
relevância	tempo	satisfação	grau	SOM	MOM	LOM	grau	satisfação
alta $\geq 1,0$	baixo $\geq 1,0$	7	1	9	9,5	10	1	9
alta $\geq 0,8$	baixo $\geq 0,8$	7	1	9	9,5	10	1	9
alta $\geq 0,5$	baixo $\geq 0,5$	7	1	9	9,5	10	1	9
alta $\geq 0,2$	baixo $\geq 0,2$	7	1	9	9,5	10	1	9
alta $\geq 1,0$	médio $\geq 1,0$	3	1	7	7	7	1	7
alta $\geq 0,8$	médio $\geq 0,8$	3	1	7	7	7	1	7
alta $\geq 0,5$	médio $\geq 0,5$	7	1	7	7	7	1	7
alta $\geq 0,2$	médio $\geq 0,2$	7	1	7	7	7	1	7
alta $\geq 1,0$	alto $\geq 1,0$	3	1	5	5	5	1	5
alta $\geq 0,8$	alto $\geq 0,8$	9	1	5	5	5	1	5
alta $\geq 0,5$	alto $\geq 0,5$	9	1	5	5	5	1	5
alta $\geq 0,2$	alto $\geq 0,2$	9	1	5	5	5	1	5
média $\geq 1,0$	baixo $\geq 1,0$	9	1	7	7	7	1	7
média $\geq 0,8$	baixo $\geq 0,8$	9	1	7	7	7	1	7
média $\geq 0,5$	baixo $\geq 0,5$	9	1	7	7	7	1	7
média $\geq 0,2$	baixo $\geq 0,2$	9	1	7	7	7	1	7
média $\geq 1,0$	médio $\geq 1,0$	9	1	5	5	5	1	5
média $\geq 0,8$	médio $\geq 0,8$	9	1	5	5	5	1	5
média $\geq 0,5$	médio $\geq 0,5$	9	1	5	5	5	1	5
média $\geq 0,2$	médio $\geq 0,2$	9	1	5	5	5	1	5
média $\geq 1$	alto $\geq 1$	9	1	3	3	3	1	3
média $\geq 0,8$	alto $\geq 0,8$	9	1	3	3	3	1	3
média $\geq 0,5$	alto $\geq 0,5$	9	1	3	3	3	1	3
média $\geq 0,2$	alto $\geq 0,2$	3	1	3	3	3	1	3
baixa $\geq 1,0$	baixo $\geq 1,0$	7	1	5	5	5	1	5
baixa $\geq 0,8$	baixo $\geq 0,8$	3	1	5	5	5	1	5
baixa $\geq 0,5$	baixo $\geq 0,5$	3	1	5	5	5	1	5
baixa $\geq 0,2$	baixo $\geq 0,2$	3	1	5	5	5	1	5
baixa $\geq 1,0$	médio $\geq 1,0$	7	1	3	3	3	1	3
baixa $\geq 0,8$	médio $\geq 0,8$	7	1	3	3	3	1	3
baixa $\geq 0,5$	médio $\geq 0,5$	7	1	3	3	3	1	3
baixa $\geq 0,2$	médio $\geq 0,2$	7	1	3	3	3	1	3
baixa $\geq 1,0$	alto $\geq 1,0$	7	1	0	0,5	1	1	1
baixa $\geq 0,8$	alto $\geq 0,8$	7	1	0	0,5	1	1	1
baixa $\geq 0,5$	alto $\geq 0,5$	3	1	0	0,5	1	1	1
baixa $\geq 0,2$	alto $\geq 0,2$	7	1	0	0,5	1	1	1

Embora sejam capazes de expressar a satisfação de modo coerente com a definição das regras *fuzzy*, os resultados de fuzzyDL e os métodos de defuzzificação SOM, MOM e LOM não conseguem expressar a imprecisão em função dos diferentes graus de pertinência das atrações nas categorias. Novamente, o problema está na geração do valor de satisfação, que perde informação relevante sobre o formato do conjunto *fuzzy* ao considerar o intervalo de grau máximo.

A mesma limitação ocorre para os resultados de fuzzyDL e consultas por satisfazibilidade máxima, cujos resultados são bastante similares aos da defuzzificação SOM. Como os graus de satisfazibilidade máxima retornados são os mesmos para todas as atrações, também não podem ser aplicados para tratar a ambiguidade dos valores de satisfação. Assim como nas consultas com entradas numéricas, as consultas por satisfazibilidade mínima não retornaram os valores de satisfação para realizar a comparação.

De modo geral, tendo em vista os resultados de HyFOM/FT-FIS e fuzzyDL, pode-se con-

cluir que os resultados mais coerentes para a consulta 1 com entradas *fuzzy* foram obtidos por HyFOM/FT-FIS com o método de raciocínio de Mamdani e defuzificação *Moment*, por expressarem adequadamente tanto a semântica das regras *fuzzy* quanto a imprecisão existente na instanciação das atrações nas categorias. Com exceção de MOM, os demais resultados de HyFOM/FT-FIS foram pertinentes, mesmo com algumas pequenas distorções dos valores inferidos em função de problemas de precisão da implementação de SIF utilizada.

### Consulta 2: saída de SIF restrita por predicado *fuzzy*

Os experimentos também exploraram entradas *fuzzy* em conjunto com a restrição da saída do SIF na ontologia *fuzzy*. Neste caso, a sintaxe da consulta é a mesma da consulta 2 com entradas numéricas, envolvendo o conceito abstrato *AboveAverageAttraction*, com a diferença de que os valores das características das atrações são obtidos como entradas *fuzzy*. Assim, o objetivo é verificar tanto a integração *ontologia fuzzy*  $\rightarrow$  *SIF*, ilustrada pela inferência das entradas *fuzzy*, quanto *SIF*  $\rightarrow$  *ontologia fuzzy*, representada pela definição de conceitos a partir de valores inferidos pelo SIF.

Em HyFOM, como os valores de satisfação inferidos pelo SIF são modelados explicitamente na ontologia *fuzzy* com o aval de especialistas, basta consultar pelo grau de pertinência das atrações com relação ao conceito *AboveAverageAttraction*. A partir dos valores de satisfação, apresentados na Tabela 6.10, obtém-se os graus de pertinência exibidos na Tabela 6.12, que podem ser utilizados para recomendar atrações turísticas acima da média. Note que, com o uso de MOM, o conceito *AboveAverageAttraction* corresponde a um conjunto *crisp*, pois a pertinência se resume aos graus  $\{0, 1\}$ . Com as demais abordagens, é possível obter uma pertinência gradual, relevante para classificar as respostas. Os graus de pertinência das instâncias envolvendo as entradas (média  $\geq 0,8$ ; médio  $\geq 0,8$ ) e (média  $\geq 0,2$ ; médio  $\geq 0,2$ ) devem ter valor igual a 1 em todas abordagens, porém as aproximações incorretas obtidas com Larsen combinado com COA e *Moment* são ocasionadas por problemas de precisão nos resultados da implementação do SIF utilizada em HyFOM.

Os resultados obtidos com FT-FIS são apresentados na Tabela 6.13. Os valores defuzificados de satisfação são gerados a partir da conjunção entre a saída do SIF e o predicado *fuzzy* *aboveAverageSatisfaction*, por isso os valores absolutos são distintos dos valores considerados por HyFOM, que defuzifica diretamente a saída do SIF. Os resultados mostram que a semântica das regras *fuzzy* é respeitada com relação à variação de entradas *fuzzy* de relevância e satisfação. A única situação de inconsistência ocorre para a entrada {baixa  $\geq 1,0$ ; alto  $\geq 1,0$ }, porque o conjunto *fuzzy* resultante do raciocínio do SIF não possui interseção com o conjunto *fuzzy* do

**Tabela 6.12: Graus de pertinência obtidos a partir da satisfação inferida por HyFOM para consulta 2 e entradas *fuzzy*.**

Entradas <i>fuzzy</i>		Mamdani			Larsen		
relevância	tempo	MOM	COA	Moment	MOM	COA	Moment
alta $\geq 1,0$	baixo $\geq 1,0$	1	1	1	1	1	1
alta $\geq 0,8$	baixo $\geq 0,8$	1	1	1	1	1	1
alta $\geq 0,5$	baixo $\geq 0,5$	1	1	1	1	1	1
alta $\geq 0,2$	baixo $\geq 0,2$	1	1	1	1	1	1
alta $\geq 1,0$	médio $\geq 1,0$	1	1	1	1	1	1
alta $\geq 0,8$	médio $\geq 0,8$	1	1	1	1	1	1
alta $\geq 0,5$	médio $\geq 0,5$	1	1	1	1	1	1
alta $\geq 0,2$	médio $\geq 0,2$	1	1	1	1	1	1
alta $\geq 1,0$	alto $\geq 1,0$	1	1	1	1	1	1
alta $\geq 0,8$	alto $\geq 0,8$	1	1	1	1	1	1
alta $\geq 0,5$	alto $\geq 0,5$	1	1	1	1	1	1
alta $\geq 0,2$	alto $\geq 0,2$	1	1	1	1	1	1
média $\geq 1,0$	baixo $\geq 1,0$	1	1	1	1	1	1
média $\geq 0,8$	baixo $\geq 0,8$	1	1	1	1	1	1
média $\geq 0,5$	baixo $\geq 0,5$	1	1	1	1	1	1
média $\geq 0,2$	baixo $\geq 0,2$	1	1	1	1	1	1
média $\geq 1,0$	médio $\geq 1,0$	1	1	1	1	1	1
média $\geq 0,8$	médio $\geq 0,8$	1	1	1	1	0,983	1
média $\geq 0,5$	médio $\geq 0,5$	1	1	1	1	1	1
média $\geq 0,2$	médio $\geq 0,2$	1	1	1	1	0,915	0,964
média $\geq 1$	alto $\geq 1$	0	0	0	0	0	0
média $\geq 0,8$	alto $\geq 0,8$	0	0,152	0,406	0	0,106	0,285
média $\geq 0,5$	alto $\geq 0,5$	0	0,388	0,562	0	0,21	0,408
média $\geq 0,2$	alto $\geq 0,2$	0	0,388	0,562	0	0,21	0,408
baixa $\geq 1,0$	baixo $\geq 1,0$	1	1	1	1	1	1
baixa $\geq 0,8$	baixo $\geq 0,8$	1	1	1	1	1	1
baixa $\geq 0,5$	baixo $\geq 0,5$	1	1	1	1	1	1
baixa $\geq 0,2$	baixo $\geq 0,2$	1	1	1	1	1	1
baixa $\geq 1,0$	médio $\geq 1,0$	0	0	0	0	0	0
baixa $\geq 0,8$	médio $\geq 0,8$	0	0,152	0,406	0	0,106	0,285
baixa $\geq 0,5$	médio $\geq 0,5$	0	0,388	0,562	0	0,21	0,408
baixa $\geq 0,2$	médio $\geq 0,2$	0	0,388	0,562	0	0,21	0,408
baixa $\geq 1,0$	alto $\geq 1,0$	0	0	0	0	0	0
baixa $\geq 0,8$	alto $\geq 0,8$	0	0	0	0	0	0
baixa $\geq 0,5$	alto $\geq 0,5$	0	0	0	0	0	0
baixa $\geq 0,2$	alto $\geq 0,2$	0	0	0	0	0	0

predicado *aboveAverageSatisfaction*. Essa inconsistência é esperada, uma vez que essa entrada *fuzzy* corresponde ao pior caso em termos de satisfação e, portanto, não deve ser considerada uma atração acima da média.

Considerando os resultados de FT-FIS, quando se analisa a variação dos graus de pertinência em uma mesma combinação de entradas *fuzzy*, existem três situações a considerar:

- Para entradas *fuzzy* envolvendo as combinações de relevância e tempo {alta, baixo}, {alta, médio} e {média, baixo}: a tendência observada nos resultados da consulta 1 é mantida, ou seja, os valores de satisfação diminuem conforme o grau diminui em função da maior flexibilidade proporcionada pelos graus de pertinência menores;
- Para as entradas *fuzzy* do tipo {alta, alto}, {média, médio} e {baixa, baixo}: em geral, a tendência esperada é de manutenção de satisfação com a diminuição dos graus, conforme observado na consulta 1. No entanto, os valores inferidos aumentam por causa da conjun-



Tabela 6.13: Resultados de FT-FIS para consulta 2 e entradas fuzzy.

Entradas fuzzy		Mamdani			Larsen		
relevância	tempo	MOM	COA	Moment	MOM	COA	Moment
alta $\geq 1,0$	baixo $\geq 1,0$	9,5	8	8,917	9,5	8	8,917
alta $\geq 0,8$	baixo $\geq 0,8$	9,5	8,549	7,855	9,5	8,713	8,186
alta $\geq 0,5$	baixo $\geq 0,5$	9,5	7,5	7,208	9,5	8,08	7,523
alta $\geq 0,2$	baixo $\geq 0,2$	9,5	7,5	7,208	9,5	8,08	7,523
alta $\geq 1,0$	médio $\geq 1,0$	7	7	7	7	7	7
alta $\geq 0,8$	médio $\geq 0,8$	7	6,918	6,793	7	6,966	6,893
alta $\geq 0,5$	médio $\geq 0,5$	7	6,871	6,778	7	6,958	6,892
alta $\geq 0,2$	médio $\geq 0,2$	7	6,871	6,778	7	6,958	6,892
alta $\geq 1,0$	alto $\geq 1,0$	5	5	5	5	5	5
alta $\geq 0,8$	alto $\geq 0,8$	5	5,211	5,467	5	5,12	5,267
alta $\geq 0,5$	alto $\geq 0,5$	5	5,586	5,833	5	5,367	5,611
alta $\geq 0,2$	alto $\geq 0,2$	5	5,586	5,833	5	5,367	5,611
média $\geq 1,0$	baixo $\geq 1,0$	7	7	7	7	7	7
média $\geq 0,8$	baixo $\geq 0,8$	7	6,918	6,793	7	6,88	6,824
média $\geq 0,5$	baixo $\geq 0,5$	7	6,871	6,778	7	6,958	6,892
média $\geq 0,2$	baixo $\geq 0,2$	7	6,871	6,778	7	6,958	6,892
média $\geq 1,0$	médio $\geq 1,0$	5	5	5	5	5	5
média $\geq 0,8$	médio $\geq 0,8$	5	5,351	5,824	5	5,347	5,821
média $\geq 0,5$	médio $\geq 0,5$	5	6,25	6,511	5	5,92	6,404
média $\geq 0,2$	médio $\geq 0,2$	5	6,25	6,511	5	5,92	6,404
média $\geq 1$	alto $\geq 1$	4	4	4	4	4	4
média $\geq 0,8$	alto $\geq 0,8$	4	5,55	5,723	4	5,5	5,379
média $\geq 0,5$	alto $\geq 0,5$	6	6	6	6	5,709	5,807
média $\geq 0,2$	alto $\geq 0,2$	6	6	6	6	5,709	5,807
baixa $\geq 1,0$	baixo $\geq 1,0$	5	5	5	5	5	5
baixa $\geq 0,8$	baixo $\geq 0,8$	5	5,211	5,467	5	5,12	5,267
baixa $\geq 0,5$	baixo $\geq 0,5$	5	5,586	5,833	5	5,367	5,611
baixa $\geq 0,2$	baixo $\geq 0,2$	5	5,586	5,833	5	5,367	5,611
baixa $\geq 1,0$	médio $\geq 1,0$	4	4	4	4	4	4
baixa $\geq 0,8$	médio $\geq 0,8$	4	5,55	5,723	4	5,5	5,379
baixa $\geq 0,5$	médio $\geq 0,5$	6	6	6	6	5,709	5,807
baixa $\geq 0,2$	médio $\geq 0,2$	6	6	6	6	5,709	5,807
baixa $\geq 1,0$	alto $\geq 1,0$	inconsist.	inconsist.	inconsist.	inconsist.	inconsist.	inconsist.
baixa $\geq 0,8$	alto $\geq 0,8$	5	5	5	3,4	4,6	4,657
baixa $\geq 0,5$	alto $\geq 0,5$	5	5	5	5	4,915	4,889
baixa $\geq 0,2$	alto $\geq 0,2$	5	5	5	5	4,915	4,889

ção da saída fuzzy do SIF com o conjunto fuzzy de *aboveAverageSatisfaction*. O conjunto fuzzy resultante da conjunção, quando defuzificado, tende a aumentar o valor absoluto de satisfação, porque os consequentes das regras que não possuem interseção com *aboveAverageSatisfaction* são desconsiderados. Os resultados são pertinentes porque seguem a conjunção modelada na ontologia fuzzy na definição do conceito *AboveAverageAttraction*;

- Para as demais entradas fuzzy: existe a tendência de aumento da satisfação com a diminuição do grau, assim como na consulta 1, em função da maior flexibilidade proporcionada pelos graus de pertinência menores.

De forma geral, as combinações de métodos de inferência de Mamdani ou Larsen com a defuzificação *Moment*, disponíveis em FT-FIS, são capazes de expressar com maior precisão os resultados de satisfação considerando as diferentes entradas fuzzy da consulta 2.

Os resultados obtidos por fuzzyDL são exibidos na Tabela 6.14. No caso de fuzzyDL com implicações fuzzy, os resultados são exatamente iguais aos valores de satisfação obtidos na

Tabela 6.14: Resultados de fuzzyDL para consulta 2 e entradas fuzzy.

Entradas fuzzy		fuzzyDL com implicações fuzzy		fuzzyDL com consulta defuzzify			fuzzyDL com max-sat(RuleSet)	
relevância	tempo	satisfação	grau	SOM	MOM	LOM	grau	satisfação
alta $\geq$ 1,0	baixo $\geq$ 1,0	7	1	9	9,5	10	1	9
alta $\geq$ 0,8	baixo $\geq$ 0,8	7	1	9	9,5	10	1	9
alta $\geq$ 0,5	baixo $\geq$ 0,5	7	1	9	9,5	10	1	9
alta $\geq$ 0,2	baixo $\geq$ 0,2	7	1	9	9,5	10	1	9
alta $\geq$ 1,0	médio $\geq$ 1,0	3	1	7	7	7	1	7
alta $\geq$ 0,8	médio $\geq$ 0,8	3	1	7	7	7	1	7
alta $\geq$ 0,5	médio $\geq$ 0,5	7	1	7	7	7	1	7
alta $\geq$ 0,2	médio $\geq$ 0,2	7	1	7	7	7	1	7
alta $\geq$ 1,0	alto $\geq$ 1,0	3	1	5	5	5	1	5
alta $\geq$ 0,8	alto $\geq$ 0,8	9	1	5	5	5	1	5
alta $\geq$ 0,5	alto $\geq$ 0,5	9	1	5	5	5	1	5
alta $\geq$ 0,2	alto $\geq$ 0,2	9	1	5	5	5	1	5
média $\geq$ 1,0	baixo $\geq$ 1,0	9	1	7	7	7	1	7
média $\geq$ 0,8	baixo $\geq$ 0,8	9	1	7	7	7	1	7
média $\geq$ 0,5	baixo $\geq$ 0,5	9	1	7	7	7	1	7
média $\geq$ 0,2	baixo $\geq$ 0,2	9	1	7	7	7	1	7
média $\geq$ 1,0	médio $\geq$ 1,0	9	1	5	5	5	1	5
média $\geq$ 0,8	médio $\geq$ 0,8	9	1	5	5	5	1	5
média $\geq$ 0,5	médio $\geq$ 0,5	9	1	5	5	5	1	5
média $\geq$ 0,2	médio $\geq$ 0,2	9	1	5	5	5	1	5
média $\geq$ 1,0	alto $\geq$ 1,0	9	1	4	4	4	0,5	4
média $\geq$ 0,8	alto $\geq$ 0,8	9	1	4	4	4	0,5	4
média $\geq$ 0,5	alto $\geq$ 0,5	9	1	4	6	8	0,5	4
média $\geq$ 0,2	alto $\geq$ 0,2	3	1	4,6	6	7,4	0,8	5
baixa $\geq$ 1,0	baixo $\geq$ 1,0	7	1	5	5	5	1	5
baixa $\geq$ 0,8	baixo $\geq$ 0,8	3	1	5	5	5	1	5
baixa $\geq$ 0,5	baixo $\geq$ 0,5	3	1	5	5	5	1	5
baixa $\geq$ 0,2	baixo $\geq$ 0,2	3	1	5	5	5	1	5
baixa $\geq$ 1,0	médio $\geq$ 1,0	7	1	4	4	4	0,5	4
baixa $\geq$ 0,8	médio $\geq$ 0,8	7	1	4	4	4	0,5	4
baixa $\geq$ 0,5	médio $\geq$ 0,5	7	1	4	6	8	0,5	4
baixa $\geq$ 0,2	médio $\geq$ 0,2	7	1	4,6	6	7,4	0,8	5
baixa $\geq$ 1,0	alto $\geq$ 1,0	inconsist.	inconsist.	inconsist.	inconsist.	inconsist.	inconsist.	inconsist.
baixa $\geq$ 0,8	alto $\geq$ 0,8	7	1	3,4	5	6,6	0,2	5
baixa $\geq$ 0,5	alto $\geq$ 0,5	3	1	4	5	6	0,5	4
baixa $\geq$ 0,2	alto $\geq$ 0,2	7	1	4,6	5	5,4	0,8	5

consulta 1. Portanto, a conjunção com o predicado *aboveAverageSatisfaction* não influenciou na geração da satisfação e os graus retornados não correspondem à pertinência esperada para *AboveAverageAttraction*.

Com relação a fuzzyDL e defuzzificação SOM, MOM e LOM, os resultados não expressam os diferentes graus considerados nas entradas fuzzy, assim como na consulta 1. A partir das entradas fuzzy {médio, alto}, é possível observar uma maior precisão na satisfação, em conformidade com a tendência verificada nos resultados de FT-FIS. De forma geral, os resultados da consulta max-sat são similares aos resultados de SOM, sendo que os diferentes graus de pertinência obtidos a partir das entradas fuzzy {médio, alto} podem ser considerados para classificar as atrações turísticas.

Por fim, os resultados da consulta 2 com entradas fuzzy demonstram que HyFOM e FT-FIS consideram a integração *ontologia fuzzy*  $\rightarrow$  *SIF* assim como *SIF*  $\rightarrow$  *ontologia fuzzy*. De modo geral, em termos da semântica esperada para as regras fuzzy, as combinações de método

de raciocínio e defuzificação providos por HyFOM e FT-FIS são capazes de inferir valores de satisfação mais precisos e, portanto, mais apropriados para a recomendação de atrações turísticas.

## 6.3 Risco Químico em Alimentos

HyFOM e FT-FIS também foram aplicados no domínio de segurança alimentar para apoiar especialistas na avaliação de risco químico de analitos (resíduos e contaminantes) detectados em amostras de alimentos. Os experimentos basearam-se em dados disponibilizados pela Embrapa Gado de Leite, vinculada ao Ministério da Agricultura, Pecuária e Abastecimento (MAPA), com foco no Plano Nacional para Controle de Resíduos e Contaminantes (PNCRC). O PNCRC é responsável por monitorar a presença de resíduos de pesticidas e drogas veterinárias assim como contaminantes ambientais em produtos alimentícios. Como o objetivo é demonstrar a aplicação de HyFOM e FT-FIS em um contexto real, foi considerada uma parte dos dados disponíveis, em função da complexidade dos procedimentos adotados pelo MAPA e PNCRC, descritos com maiores detalhes por Magalhães Júnior (2011), Magalhães Júnior et al. (2012). A ontologia *fuzzy* e as regras *fuzzy*, descritas na próxima seção, exploram os principais conceitos e processos deste domínio com foco na integração das inferências e sua contribuição para a análise de risco químico em alimentos.

### 6.3.1 Ontologia *Fuzzy* e Regras *Fuzzy*

Conforme descrito por Magalhães Júnior (2011), Magalhães Júnior et al. (2012), análises laboratoriais obtêm a concentração de diferentes analitos em amostras de alimentos. Para cada analito, existe um nível máximo estabelecido pela Comissão *Codex Alimentarius*, que é a concentração máxima oficialmente permitida daquela substância em um tipo específico de alimento. Cada concentração de analito é confrontada com seu respectivo limite máximo, em porcentagem, para obter o risco de concentração (*Concentration Risk*) do analito detectado em uma amostra de alimento. Se a concentração detectada de um analito estiver abaixo ou igual ao seu limite máximo, o analito é denominado conforme com relação à amostra analisada; caso contrário é denominado não-conforme. Além do risco de concentração, existem outros fatores de risco considerados pelos especialistas do PNCRC (MAGALHÃES JÚNIOR, 2011):

- Tendência da concentração do analito: indica se a concentração de um analito possui uma tendência para diminuir, estabilizar ou agravar em curto, médio ou longo prazo, de

acordo com o seu perfil toxicológico. O tipo do analito pode influenciar na tendência, por exemplo, uma determinada categoria de analito pode ter a tendência de diminuir, estabilizar ou aumentar a concentração detectada;

- Período de ajustamento: período estimado para que o provedor da amostra realize as ações necessárias de modo a ajustá-la aos requisitos de conformidade do MAPA com relação a um determinado analito;
- Custo de ajustamento: custo estimado para que o provedor ajuste a amostra de acordo com os requisitos de conformidade do MAPA com relação a um determinado analito.

Os fatores de risco são combinados para a obtenção do risco agregado do analito associado a uma amostra de alimento. Para tanto, Magalhães Júnior (2011) desenvolveu a abordagem Chem-risk, que se baseia em um modelo matemático para calcular o risco agregado do analito a partir dos fatores de risco. Embora Chem-risk proporcione resultados pertinentes ao domínio, o uso de um modelo matemático pode comprometer a interpretabilidade dos especialistas com relação ao contexto e às causas associadas aos valores de risco agregado. Neste sentido, o uso de ontologias *fuzzy* e regras *fuzzy* pode contribuir para tornar o processo mais transparente e interpretável, ao considerar estruturas semânticas e termos linguísticos que são mais próximos da linguagem humana. Portanto, a aplicação de HyFOM e FT-FIS é apropriada neste domínio, sendo que seus resultados foram comparados à abordagem Chem-risk, considerada como a referência com relação ao risco agregado do analito das amostras disponíveis.

A partir do risco agregado do analito, é possível determinar ações de intervenção que devem ser aplicadas aos provedores das amostras de alimentos. Uma intervenção preventiva deve ser aplicada no caso de analitos com tendência de agravamento de curto prazo e que tenham risco agregado aproximadamente médio. Os casos mais graves demandam uma intervenção máxima, atribuída a analitos não-conformes com risco agregado que tende ao máximo. Além disso, existem as situações em que não é necessário realizar intervenções, quando os analitos presentes nas amostras são conformes e possuem risco agregado próximo do mínimo.

Com a supervisão de dois especialistas do domínio, pesquisadores da Embrapa Gado de Leite vinculados ao MAPA, os principais conceitos relacionados ao domínio de risco químico de analitos foram modelados em uma ontologia *fuzzy* representada tanto em sintaxe de fuzzyDL (Tabela 6.15) quanto na sintaxe estendida para FT-FIS (Tabela 6.16).

Os fatores de risco são representados na ontologia *fuzzy* como propriedades funcionais do domínio dos números reais. O vocabulário e os conjuntos *fuzzy* utilizados na definição dos

Tabela 6.15: Ontologia *fuzzy* sobre risco químico de analitos utilizada em HyFOM e fuzzyDL.

```

(define-fuzzy-logic zadeh)

(functional hasConcentrationRisk )
(range hasConcentrationRisk *real* 0.0 200.0 )
(define-fuzzy-concept negligibleCR left-shoulder(0.0, 200.0, 20.0, 40.0) )
(define-fuzzy-concept acceptableCR trapezoidal(0.0, 200.0, 20.0, 40.0, 80.0, 90.0) )
(define-fuzzy-concept nearCR trapezoidal(0.0, 200.0, 80.0, 90, 100.0, 100.0) )
(define-fuzzy-concept equivalentCR crisp(0.0, 200.0, 100.0, 100.0) )
(define-fuzzy-concept highCR trapezoidal(0.0, 200.0, 100.0, 100, 120.0, 130.0) )
(define-fuzzy-concept intolerableCR right-shoulder(0.0, 200.0, 120.0, 130.0) )

(functional hasConcentrationTrend )
(range hasConcentrationTrend *real* 0.0 3.0 )
(define-fuzzy-concept decreaseOrStabilize crisp(0.0, 3.0, 0.0, 0.0) )
(define-fuzzy-concept longTermIncrease crisp(0.0, 3.0, 1.0, 1.0) )
(define-fuzzy-concept mediumTermIncrease crisp(0.0, 3.0, 2.0, 2.0) )
(define-fuzzy-concept shortTermIncrease crisp(0.0, 3.0, 3.0, 3.0) )

(functional hasAdjustmentPeriod )
(range hasAdjustmentPeriod *real* 0.0 3.0 )
(define-fuzzy-concept unnecessaryAP crisp(0.0, 3.0, 0.0, 0.0) )
(define-fuzzy-concept shortAP crisp(0.0, 3.0, 1.0, 1.0) )
(define-fuzzy-concept mediumAP crisp(0.0, 3.0, 2.0, 2.0) )
(define-fuzzy-concept longAP crisp(0.0, 3.0, 3.0, 3.0) )

(functional hasAdjustmentCost )
(range hasAdjustmentCost *real* 0.0 3.0 )
(define-fuzzy-concept unnecessaryAC crisp(0.0, 3.0, 0.0, 0.0) )
(define-fuzzy-concept lowAC crisp(0.0, 3.0, 1.0, 1.0) )
(define-fuzzy-concept mediumAC crisp(0.0, 3.0, 2.0, 2.0) )
(define-fuzzy-concept highAC crisp(0.0, 3.0, 3.0, 3.0) )

(functional hasAggregateRisk )
(range hasAggregateRisk *real* 1.0 15.0 )
(define-fuzzy-concept negligibleAR crisp(1.0, 15.0, 1.0, 1.5) )
(define-fuzzy-concept veryLowAR triangular(1.0, 15.0, 1.5, 1.5, 4.75) )
(define-fuzzy-concept lowAR triangular(1.0, 15.0, 1.5, 4.75, 8.0) )
(define-fuzzy-concept mediumAR triangular(1.0, 15.0, 4.75, 8.0, 11.25) )
(define-fuzzy-concept highAR triangular(1.0, 15.0, 8.0, 11.25, 14.5) )
(define-fuzzy-concept veryHighAR triangular(1.0, 15.0, 11.25, 14.5, 14.5) )
(define-fuzzy-concept intolerableAR crisp(1.0, 15.0, 14.5, 15.0) )
(define-fuzzy-concept maximumAR right-shoulder(1.0, 15.0, 8.0, 14.0) )
(define-fuzzy-concept aroundMediumAR triangular(1.0, 15.0, 2.0, 8.0, 14.0) )
(define-fuzzy-concept minimumAR left-shoulder(1.0, 15.0, 2.0, 8.0) )

(define-concept CompliantAnalysis
  (and (<= hasConcentrationRisk 100) (some hasAggregateRisk minimumAR)))
(define-concept PreventiveIntervention
  (and (some hasConcentrationTrend shortTermIncrease)(some hasAggregateRisk aroundMediumAR)))
(define-concept MaximumIntervention
  (and (not (<= hasConcentrationRisk 100)) (some hasAggregateRisk maximumAR)))

(instance analysisBeef1 AnalyteSampleAnalysis 1.0)
(instance analysisBeef1 (= hasConcentrationRisk 105.7 ) 1.0 )
(instance analysisBeef1 (= hasConcentrationTrend 3.0 ) 1.0 )
(instance analysisBeef1 (= hasAdjustmentPeriod 1.0 ) 1.0 )
(instance analysisBeef1 (= hasAdjustmentCost 2.0 ) 1.0 )

```

predicados *fuzzy* foram modelados com base nos dados disponíveis e nas sugestões dos especialistas do domínio de risco químico de analitos. Como os valores disponíveis dos fatores de risco referentes à tendência da concentração, ao prazo e ao custo de ajustamento são dados nominais (não contínuos), os predicados *fuzzy* correspondentes foram modelados como conjuntos *crisp*. Também são representados conceitos abstratos para os tipos de intervenção, além do conceito correspondente às análises conformes (*CompliantAnalysis*), definidos em função dos valores de

**Tabela 6.16: Ontologia fuzzy sobre risco químico de analitos utilizada em FT-FIS.**

```

(define-fuzzy-logic zadeh)

(functional hasConcentrationRisk )
(range hasConcentrationRisk *real* 0.0 200.0 )
(define-fuzzy-predicate negligibleCR(x) *real* left-shoulder(x, 0.0, 200.0, 20.0, 40.0) )
(define-fuzzy-predicate acceptableCR(x) *real* trapezoidal(x, 0.0, 200.0, 20.0, 40.0, 80.0, 90.0) )
(define-fuzzy-predicate nearCR(x) *real* trapezoidal(x, 0.0, 200.0, 80.0, 90, 100.0, 100.0) )
(define-fuzzy-predicate equivalentCR(x) *real* crisp(x, 0.0, 200.0, 100.0, 100.0) )
(define-fuzzy-predicate highCR(x) *real* trapezoidal(x,0.0, 200.0, 100.0, 100, 120.0, 130.0) )
(define-fuzzy-predicate intolerableCR(x) *real* right-shoulder(x, 0.0, 200.0, 120.0, 130.0) )

(functional hasConcentrationTrend )
(range hasConcentrationTrend *real* 0.0 3.0 )
(define-fuzzy-predicate decreaseOrStabilize(x) *real* crisp(x, 0.0, 3.0, 0.0, 0.0) )
(define-fuzzy-predicate longTermIncrease(x) *real* crisp(x, 0.0, 3.0, 1.0, 1.0) )
(define-fuzzy-predicate mediumTermIncrease(x) *real* crisp(x, 0.0, 3.0, 2.0, 2.0) )
(define-fuzzy-predicate shortTermIncrease(x) *real* crisp(x, 0.0, 3.0, 3.0, 3.0) )

(functional hasAdjustmentPeriod )
(range hasAdjustmentPeriod *real* 0.0 3.0 )
(define-fuzzy-predicate unnecessaryAP(x) *real* crisp(x, 0.0, 3.0, 0.0, 0.0) )
(define-fuzzy-predicate shortAP(x) *real* crisp(x, 0.0, 3.0, 1.0, 1.0) )
(define-fuzzy-predicate mediumAP(x) *real* crisp(x, 0.0, 3.0, 2.0, 2.0) )
(define-fuzzy-predicate longAP(x) *real* crisp(x, 0.0, 3.0, 3.0, 3.0) )

(functional hasAdjustmentCost )
(range hasAdjustmentCost *real* 0.0 3.0 )
(define-fuzzy-predicate unnecessaryAC(x) *real* crisp(x, 0.0, 3.0, 0.0, 0.0) )
(define-fuzzy-predicate lowAC(x) *real* crisp(x, 0.0, 3.0, 1.0, 1.0) )
(define-fuzzy-predicate mediumAC(x) *real* crisp(x, 0.0, 3.0, 2.0, 2.0) )
(define-fuzzy-predicate highAC(x) *real* crisp(x, 0.0, 3.0, 3.0, 3.0) )

(functional hasAggregateRisk )
(range hasAggregateRisk *real* 1.0 15.0 )
(define-fuzzy-predicate negligibleAR(x) *real* crisp(x, 1.0, 15.0, 1.0, 1.5) )
(define-fuzzy-predicate veryLowAR(x) *real* triangular(x, 1.0, 15.0, 1.5, 1.5, 4.75) )
(define-fuzzy-predicate lowAR(x) *real* triangular(x, 1.0, 15.0, 1.5, 4.75, 8.0) )
(define-fuzzy-predicate mediumAR(x) *real* triangular(x, 1.0, 15.0, 4.75, 8.0, 11.25) )
(define-fuzzy-predicate highAR(x) *real* triangular(x, 1.0, 15.0, 8.0, 11.25, 14.5) )
(define-fuzzy-predicate veryHighAR(x) *real* triangular(x, 1.0, 15.0, 11.25, 14.5, 14.5) )
(define-fuzzy-predicate intolerableAR(x) *real* crisp(x, 1.0, 15.0, 14.5, 15.0) )
(define-fuzzy-predicate minimumAR(x) *real* left-shoulder(x, 1.0, 15.0, 2.0, 8.0) )
(define-fuzzy-predicate aroundMediumAR(x) *real* triangular(x, 1.0, 15.0, 2.0, 8.0, 14.0) )
(define-fuzzy-predicate maximumAR(x) *real* right-shoulder(x, 1.0, 15.0, 8.0, 14.0) )

(define-fuzzy-predicate minimum(v0,v1,v2,v3,v4) *real*
(and fuzzyRuleReasoning("RuleBase1";v0,v1,v2,v3,v4) minimumAR(v4)))
(define-fuzzy-predicate aroundMedium(v0,v1,v2,v3,v4) *real*
(and fuzzyRuleReasoning("RuleBase1";v0,v1,v2,v3,v4) aroundMediumAR(v4)))
(define-fuzzy-predicate maximum(v0,v1,v2,v3,v4) *real*
(and fuzzyRuleReasoning("RuleBase1";v0,v1,v2,v3,v4) maximumAR(v4)))

(define-concept CompliantAnalysis
(and (some hasConcentrationRisk.<= 100) (some hasConcentrationRisk,hasConcentrationTrend,
hasAdjustmentPeriod,hasAdjustmentCost,hasAggregateRisk.minimum)))
(define-concept PreventiveIntervention
(and (some hasConcentrationTrend.shortTermIncrease) (some hasConcentrationRisk,
hasConcentrationTrend,hasAdjustmentPeriod,hasAdjustmentCost,hasAggregateRisk.aroundMedium)))
(define-concept MaximumIntervention
(and (some hasConcentrationRisk.> 100) (some hasConcentrationRisk,hasConcentrationTrend,
hasAdjustmentPeriod,hasAdjustmentCost,hasAggregateRisk.maximum)))

(instance analysisBeef1 AnalyteSampleAnalysis 1.0)
(instance analysisBeef1 (= hasConcentrationRisk 105.7 ) 1.0 )
(instance analysisBeef1 (= hasConcentrationTrend 3.0 ) 1.0 )
(instance analysisBeef1 (= hasAdjustmentPeriod 1.0 ) 1.0 )
(instance analysisBeef1 (= hasAdjustmentCost 2.0 ) 1.0 )

```

risco agregado do analito. Por fim, é exemplificada uma instância de uma análise envolvendo um analito em uma amostra de carne bovina e os respectivos valores dos fatores de risco. Por questão de sigilo dos dados, os analitos, as amostras e os provedores não são identificados por seus nomes reais, sendo substituídos por códigos numéricos.

A relação entre os fatores de risco para derivar o risco agregado do analito foi modelada por meio de regras *fuzzy*, extraídas a partir dos dados disponibilizados pelo MAPA. Especificamente, foram selecionados dados de 114 análises laboratoriais referentes aos lotes 226 e 229 da matriz carne bovina, envolvendo 19 analitos distintos, por apresentarem uma maior variabilidade de fatores de risco e de valores de risco agregado. Utilizou-se uma árvore de decisão para induzir o conjunto de regras *fuzzy*, sendo que os valores de risco agregado do analito são definidos por Chem-risk, e os fatores de risco numéricos são substituídos pelo conjunto *fuzzy* de maior pertinência. Especialistas supervisionaram as regras geradas pela árvore de decisão, fazendo pequenas modificações de acordo com a semântica do domínio. Ao final, foram consideradas 17 regras *fuzzy*, traduzidas para a linguagem FML. A Tabela 6.17 descreve as regras sem a formatação FML, por ser muito extensa.

Neste domínio de risco químico em alimentos também é possível identificar os requisitos de modelagem descritos na Seção 5.2 do Capítulo 5, que indicam que HyFOM e FT-FIS podem ser aplicados para tratar a integração de raciocínio de ontologias *fuzzy* e regras *fuzzy*:

1. Valores de propriedades concretas definidos a partir de regras *fuzzy* que combinam outras propriedades concretas numéricas: é o caso do risco agregado do analito, cujo valor pode ser definido a partir de regras *fuzzy* que combinam valores imprecisos dos fatores de risco (Tabela 6.17);
2. Valores de propriedades concretas obtidos por meio de inferências da ontologia *fuzzy*: embora esse caso não tenha sido explorado na ontologia *fuzzy*, as categorias de analitos podem influenciar nos valores de tendência da concentração do analito;
3. Definição de conceitos abstratos em função de valores de propriedades concretas numéricas: é o caso dos conceitos modelados em função do risco agregado do analito como os conceitos relativos aos tipos de intervenção e o conceito que representa análises conformes.

Com base na ontologia *fuzzy* e nas regras *fuzzy* descritas nesta seção, HyFOM, FT-FIS e fuzzyDL foram aplicados para determinar o risco agregado do analito e sugerir as intervenções mais apropriadas para as amostras analisadas. As próximas subseções descrevem os resultados obtidos neste domínio e as comparações feitas com fuzzyDL.

**Tabela 6.17: Regras *fuzzy* para inferir risco agregado do analito.**

<p>If hasAdjustmentCost is unnecessaryAC and hasConcentrationRisk is negligibleCR Then hasAggregateRisk is negligibleAR</p> <p>If hasAdjustmentCost is lowAC and hasAdjustmentPeriod is shortAP Then hasAggregateRisk is mediumAR</p> <p>If hasAdjustmentCost is lowAC and hasAdjustmentPeriod is mediumAP Then hasAggregateRisk is mediumAR</p> <p>If hasAdjustmentCost is lowAC and hasAdjustmentPeriod is longAP Then hasAggregateRisk is highAR</p> <p>If hasAdjustmentCost is mediumAC and hasAdjustmentPeriod is unnecessaryAP Then hasAggregateRisk is highAR</p> <p>If hasAdjustmentCost is mediumAC and hasAdjustmentPeriod is shortAP Then hasAggregateRisk is highAR</p> <p>If hasAdjustmentCost is mediumAC and hasAdjustmentPeriod is mediumAP Then hasAggregateRisk is highAR</p> <p>If hasAdjustmentCost is mediumAC and hasAdjustmentPeriod is longAP Then hasAggregateRisk is veryHighAR</p> <p>If hasAdjustmentCost is highAC Then hasAggregateRisk is highAR</p> <p>If hasAdjustmentCost is unnecessaryAC and hasConcentrationRisk is acceptableCR Then hasAggregateRisk is veryLowAR</p> <p>If hasAdjustmentCost is unnecessaryAC and hasConcentrationRisk is nearCR Then hasAggregateRisk is veryLowAR</p> <p>If hasAdjustmentCost is unnecessaryAC and hasConcentrationRisk is equivalentCR Then hasAggregateRisk is lowAR</p> <p>If hasAdjustmentCost is lowAC and hasAdjustmentPeriod is unnecessaryAP and hasConcentrationRisk is negligibleCR Then hasAggregateRisk is veryLowAR</p> <p>If hasAdjustmentCost is lowAC and hasAdjustmentPeriod is unnecessaryAP and hasConcentrationRisk is acceptableCR Then hasAggregateRisk is veryLowAR</p> <p>If hasAdjustmentCost is lowAC and hasAdjustmentPeriod is unnecessaryAP and hasConcentrationRisk is nearCR Then hasAggregateRisk is veryLowAR</p> <p>If hasAdjustmentCost is lowAC and hasAdjustmentPeriod is unnecessaryAP and hasConcentrationRisk is equivalentCR Then hasAggregateRisk is veryLowAR</p> <p>If hasAdjustmentCost is lowAC and hasAdjustmentPeriod is unnecessaryAP and hasConcentrationRisk is highCR Then hasAggregateRisk is lowAR</p>
--

### 6.3.2 Experimentos com Entradas Numéricas

Nestes experimentos, os valores dos fatores de risco de cada análise são representados explicitamente na ontologia *fuzzy*, utilizados como entradas para que o SIF obtenha o risco agregado com base em regras *fuzzy*. Assim, o foco dos testes com entradas numéricas foi voltado para a integração *SIF* → *ontologia fuzzy*, analisando o uso das saídas do SIF na ontologia. Resultados preliminares comparando HyFOM e fuzzyDL foram publicados por Yaguinuma et al. (2013).



Foram consideradas 114 análises laboratoriais modeladas na ontologia *fuzzy*, conforme ilustrado nas tabelas 6.15 e 6.16. Os valores de risco agregado da abordagem Chem-risk foram usados como referência na comparação com os resultados de fuzzyDL. Assim como no domínio de atrações turísticas, as regras *fuzzy* foram representadas em fuzzyDL como implicações *fuzzy* e por meio de definições de conceito e operadores *fuzzy* de acordo com a sintaxe proposta por Bobillo e Straccia (2008). Neste contexto, dois tipos de consulta foram realizados: (i) para analisar os valores inferidos diretamente a partir das regras *fuzzy* e (ii) para analisar o uso desses valores em conceitos definidos na ontologia *fuzzy*.

### Consulta 1: risco agregado inferido a partir de regras *fuzzy*

Para comparar os resultados do raciocínio baseado em regras *fuzzy* providos por HyFOM, FT-FIS e fuzzyDL, foi submetida uma consulta para obter os valores de risco agregado das análises analito-amostra modeladas na ontologia *fuzzy*. No caso de HyFOM, como os valores inferidos pelo SIF são representados explicitamente na ontologia *fuzzy*, a consulta é feita diretamente pelo valor da propriedade de risco agregado para cada instância de análise analito-amostra. Em FT-FIS, definiu-se um predicado *fuzzy* a partir do predicado *fuzzyRuleReasoning*, utilizado para a definição do conceito abstrato *Analysis*:

```
(define-fuzzy-predicate minimum(v0,v1,v2,v3,v4) *real* fuzzyRuleReasoning("RuleBase1";v0,v1,v2,v3,v4))
(define-concept Analysis (some hasConcentrationRisk,hasConcentrationTrend,hasAdjustmentPeriod,
hasAdjustmentCost,hasAggregateRisk.minimum))
```

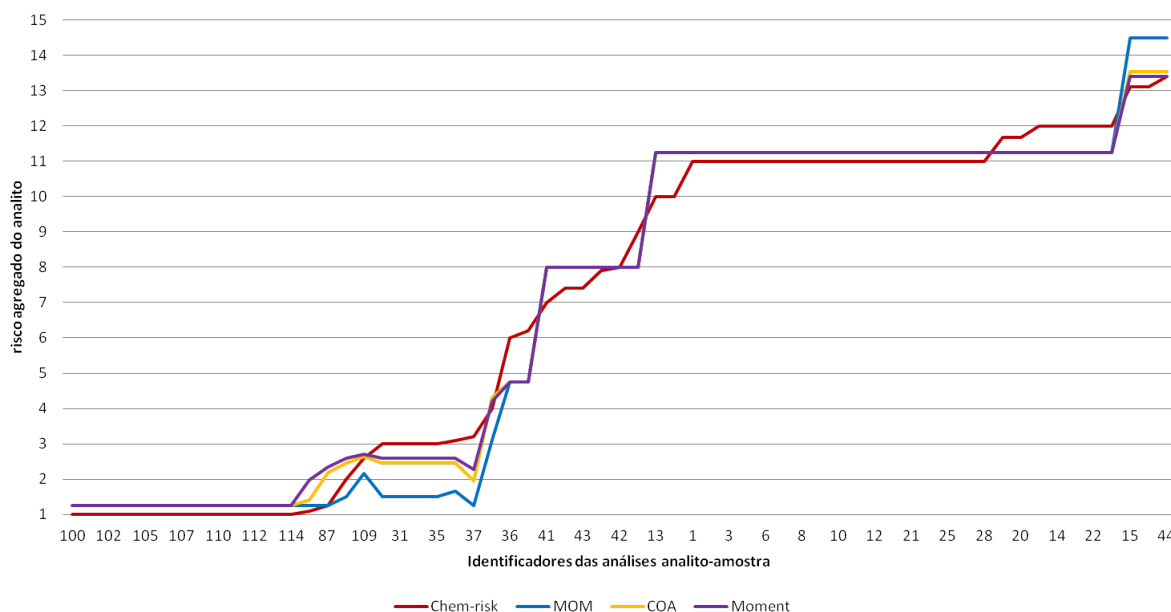
A consulta submetida a FT-FIS foi a verificação de satisfazibilidade para

$$\psi = \langle x : Analysis > 0.0 \rangle$$

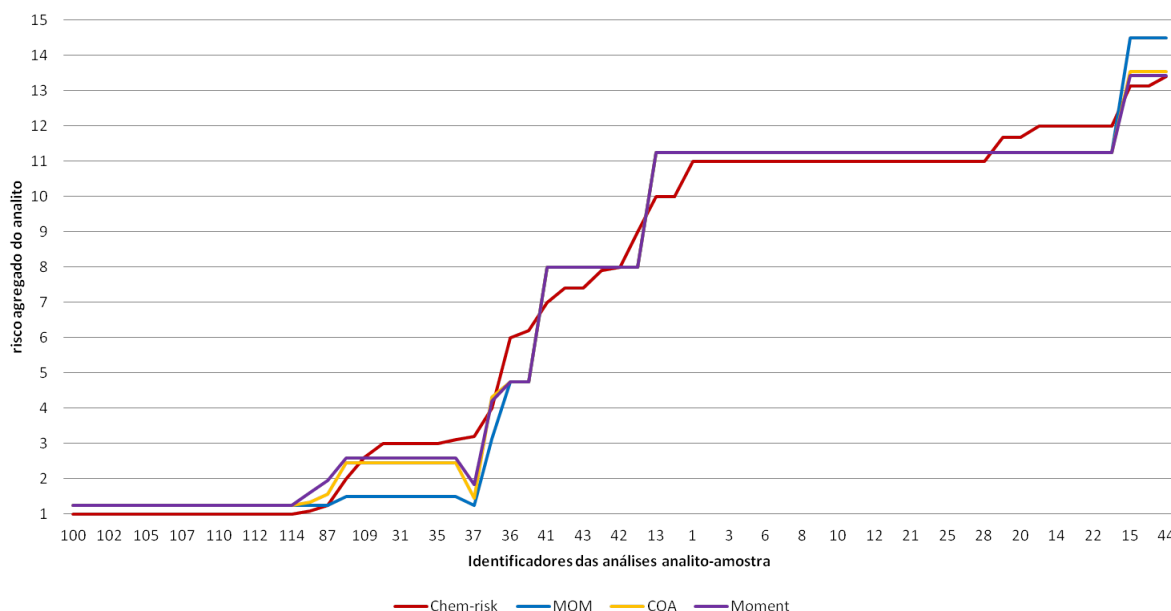
com defuzzificação dos valores de satisfação inferidos pelo SIF para todas instâncias  $x$  de análises analito-amostra. Assim como no domínio de atrações turísticas, a restrição de grau de verdade ( $> 0.0$ ) foi utilizada para que a saída *fuzzy* do SIF fosse considerada de forma direta. Os métodos de defuzzificação considerados foram meio dos máximos (MOM), centro de área (COA) e defuzzificação *Moment*.

Considerando essa consulta, os resultados de HyFOM e FT-FIS são apresentados de forma unificada, em função da mesma implementação de SIF. O risco agregado inferido segundo o método de Mamdani é ilustrado graficamente na Figura 6.6 e para o método de Larsen na Figura 6.7, combinados com diferentes métodos de defuzzificação. Nos gráficos, é exibida uma parte das análises disponíveis, pois as demais possuem o valor mínimo de risco agregado, assim

como as análises com os identificadores 100, 102, 105, 107, 110, 112 e 114. Em cada gráfico, os valores de risco agregado de Chem-risk também são apresentados para referência.



**Figura 6.6:** Risco agregado inferido por HyFOM e FT-FIS com método de Mamdani, combinado com diferentes métodos de defuzificação.



**Figura 6.7:** Risco agregado inferido por HyFOM e FT-FIS com método de Larsen, combinado com diferentes métodos de defuzificação.

É possível notar, pelos gráficos de HyFOM e FT-FIS, que os métodos de Mamdani e Larsen combinados com defuzificação COA e *Moment* proporcionam as melhores aproximações com relação aos resultados de Chem-risk. Novamente, assim como no domínio de atrações turísticas,

a defuzzificação MOM é menos precisa, gerando resultados com maior perda de informação em ambos métodos de inferência. Para um conjunto de análises (100 a 114; 36 a 15), os valores inferidos são os mesmos independentemente de método de inferência e defuzzificação porque as regras *fuzzy* foram disparadas em função de propriedades com valores *crisp*, como custo e período de ajustamento. Assim, o raciocínio aproximado não teve influência nestes resultados.

Para os testes com fuzzyDL e regras implicativas, foram submetidas consultas pela satisfazibilidade mínima e máxima com relação ao conceito *Analysis*, definido em função de valores de risco agregado. Além disso, foi realizada a consulta *show-concrete-fillers-for* para retornar os valores de risco agregado inferidos:

```
(define-concept Analysis (some hasAggregateRisk minimumAR))
(min-sat? Analysis analysisBeef1 )
(max-sat? Analysis analysisBeef1 )
(show-concrete-fillers-for analysisBeef1 hasAggregateRisk)
```

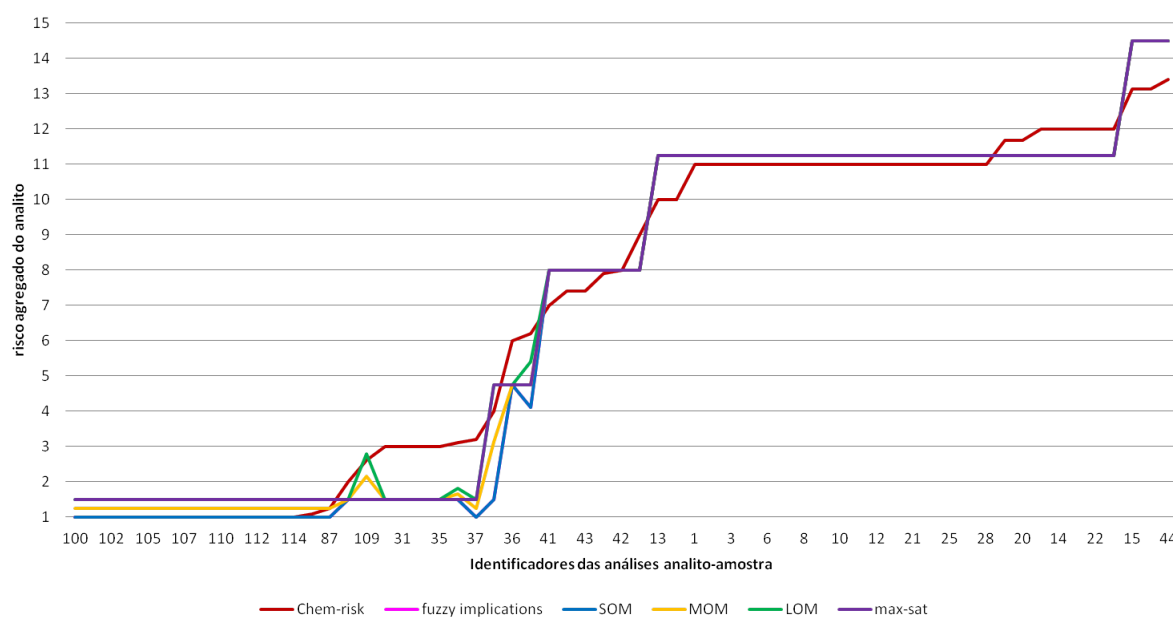
Com relação a fuzzyDL e regras *fuzzy* segundo a semântica de Mamdani, representadas pelo conceito *RuleSet*, foram realizadas consultas para cada método de defuzzificação (SOM, MOM, LOM) e para satisfazibilidade mínima e máxima:

```
(defuzzify-som? RuleSet analysisBeef1 hasAggregateRisk )
(defuzzify-mom? RuleSet analysisBeef1 hasAggregateRisk )
(defuzzify-lom? RuleSet analysisBeef1 hasAggregateRisk )

(min-sat? RuleSet analysisBeef1 )
(max-sat? RuleSet analysisBeef1 )
(show-concrete-fillers-for analysisBeef1 hasAggregateRisk)
```

Os valores de risco agregado do analito inferido pelas abordagens de fuzzyDL são apresentados na Figura 6.8. De acordo com os resultados, valores distintos de risco agregado são obtidos pelas diferentes abordagens de fuzzyDL em um conjunto de análises (100 a 36). A partir da análise 36 os resultados são idênticos, independente da abordagem de fuzzyDL. Os valores inferidos pelas implicações *fuzzy* são idênticos aos da consulta *max-sat(RuleSet)*, por isso os gráficos de ambas abordagens estão sobrepostos na figura. De forma geral, é possível notar pelos gráficos que os resultados de fuzzyDL são menos precisos que os resultados de HyFOM e FT-FIS, em comparação com os valores de referência providos por Chem-risk.

Além da análise por meio dos gráficos, é possível obter o erro médio quadrático com base nos valores de risco de Chem-risk disponíveis para os experimentos, apresentado na Tabela 6.18. Os valores de erro confirmam a análise realizada a partir dos gráficos: as combinações providas por HyFOM e FT-FIS (Mamdani-COA, Mamdani-Moment, Larsen-COA e Larsen-Moment) são mais próximas dos resultados de Chem-risk; enquanto fuzzyDL com implicações *fuzzy* e consulta *max-sat* possuem o pior desempenho considerando os valores de risco agregado.



**Figura 6.8: Risco agregado inferido pelas abordagens de fuzzyDL.**

Portanto, é possível verificar que HyFOM e FT-FIS proporcionaram inferências baseadas em regras *fuzzy* mais próximas de Chem-Risk, em comparação com os resultados das abordagens disponíveis em fuzzyDL.

**Tabela 6.18: Erro médio quadrático de HyFOM/FT-FIS e fuzzyDL.**

HyFOM/FT-FIS						fuzzyDL				
Mamdani			Larsen			implicações fuzzy	fuzzyDL com defuzzify			fuzzyDL com max-sat(RuleSet)
MOM	COA	Moment	MOM	COA	Moment		SOM	MOM	LOM	
0,3421	0,1988	0,1951	0,3553	0,2054	0,1926	0,4558	0,3970	0,3421	0,4245	0,4558

### Consulta 2: saída de SIF restrita por predicado *fuzzy*

Outro tipo de consulta analisada nos experimentos trata do uso da saída do SIF em um conceito da ontologia *fuzzy*, com foco na integração  $SIF \rightarrow ontologia\ fuzzy$ . É o caso do conceito abstrato *MaximumIntervention*, que indica o nível de intervenção máximo para análises não conformes que possuam risco máximo. Na ontologia *fuzzy*, este conceito é definido com base no predicado *fuzzy maximumAR*, representado pelo conjunto *fuzzy right-shoulder*(8.0, 14.0). Especificamente, esta consulta é útil para alertar aos especialistas quais análises demandam uma intervenção maior em função dos valores de risco agregado.

No caso de HyFOM, os valores de risco agregado inferidos pelo SIF, apresentados nas Figuras 6.6 e 6.7, são representados explicitamente na ontologia *fuzzy*, com a supervisão dos especialistas. Portanto, basta realizar uma consulta pela pertinência com relação ao conceito abstrato *MaximumIntervention* para cada análise. As análises podem, então, ser classificadas

de acordo com o grau de pertinência retornado, indicando quais necessitam de uma intervenção mais urgente. Conforme ilustrado nos gráficos das Figuras 6.6 e 6.7, da análise 13 a 22 os valores de risco agregado inferidos são iguais, independentemente de método de raciocínio e defuzzificação. Desta forma, os graus de pertinência para *MaximumIntervention* também são iguais para essas análises, pois são considerados valores de risco maior que 8, em função da restrição do predicado *fuzzy maximumAR* definido pelo conjunto *fuzzy right-shoulder(8.0, 14.0)*. As análises 15, 16 e 44, de risco mais próximo do valor máximo, apresentam grau de pertinência distintos em função dos valores de risco inferidos pelas diferentes combinações de método de raciocínio e defuzzificação providas por HyFOM.

Em FT-FIS, para representar o conceito *MaximumIntervention*, definiu-se o predicado *fuzzy maximum* a partir da conjunção dos predicados *fuzzyRuleReasoning* e *maximumAR*:

```
(define-fuzzy-predicate maximumAR(x) *real* right-shoulder(x, 1.0, 15.0, 8.0, 14.0) )
(define-fuzzy-predicate maximum(v0,v1,v2,v3,v4) *real*
  (and fuzzyRuleReasoning("RuleBase1";v0,v1,v2,v3;v4) maximumAR(v4)))

(define-concept MaximumIntervention
  (and (some hasConcentrationRisk.> 100) (some hasConcentrationRisk,hasConcentrationTrend,
  hasAdjustmentPeriod,hasAdjustmentCost,hasAggregateRisk.maximum)))
```

Assim, a consulta submetida a FT-FIS foi a verificação de satisfazibilidade para

$$\psi = \langle x : \text{MaximumIntervention} > 0.0 \rangle$$

com defuzzificação dos valores de satisfação para todas instâncias  $x$  de análises analito-amostra. Assim como nas demais consultas, a restrição de grau  $> 0.0$  foi utilizada para que a saída *fuzzy* seja considerada de forma direta. Os valores de risco agregado inferidos por FT-FIS, considerando os métodos de raciocínio de Mamdani e Larsen combinados com defuzzificação MOM, COA e *Moment*, são descritos graficamente na Figura 6.9. Nesta figura, os resultados obtidos por FT-FIS são os mesmos para Mamdani e Larsen dado um mesmo método de defuzzificação. Assim, os diferentes gráficos correspondem aos valores obtidos pelos métodos de defuzzificação disponíveis em FT-FIS para ambos os métodos de Mamdani e Larsen, indicados na legenda por M/L.

Para fuzzyDL com regras implicativas, foram submetidas consultas pela satisfazibilidade mínima e máxima com relação ao conceito *MaximumIntervention*:

```
(define-fuzzy-concept maximumAR right-shoulder(1.0, 15.0, 8.0, 14.0) )
(define-concept MaximumIntervention
  (and (not (<= hasConcentrationRisk 100)) (some hasAggregateRisk maximumAR)))

(min-sat? MaximumIntervention analysisBeef1 )
(max-sat? MaximumIntervention analysisBeef1 )
(show-concrete-fillers-for analysisBeef1 hasAggregateRisk)
```

No entanto, os resultados retornados por fuzzyDL e regras implicativas são exatamente iguais aos da consulta 1. Desta forma, as restrições sobre os valores das propriedades *hasConcentrationRisk* e *hasAggregateRisk* utilizadas para definir o conceito *MaximumIntervention* não são consideradas ao inferir os valores de risco agregado e determinar o grau de pertinência a *MaximumIntervention*. Por exemplo, análises que tenham valores de *hasConcentrationRisk* menores que 100 deveriam ter grau de pertinência zero para *MaximumIntervention*, por serem análises conformes. Entretanto, os resultados das regras implicativas retornaram grau máximo para todas as análises consultadas por *max-sat*. Em função destas incoerências, os resultados das regras implicativas não foram considerados para comparação com FT-FIS, cujos resultados foram gerados conforme as restrições definidas para o conceito *MaximumIntervention*.

Com relação à consulta submetida a fuzzyDL e regras *fuzzy* segundo a semântica de Mamdani, o conceito *MaximumIntervention* foi definido a partir da conjunção entre o conceito *RuleSet*, que representa as regras *fuzzy*, e as restrições dos valores de *hasConcentrationRisk* e *hasAggregateRisk* pelo predicado *fuzzy maximumAR*. Foram realizadas consultas para cada método de defuzzificação (SOM, MOM, LOM) e para satisfazibilidade mínima e máxima envolvendo o conceito *MaximumIntervention*:

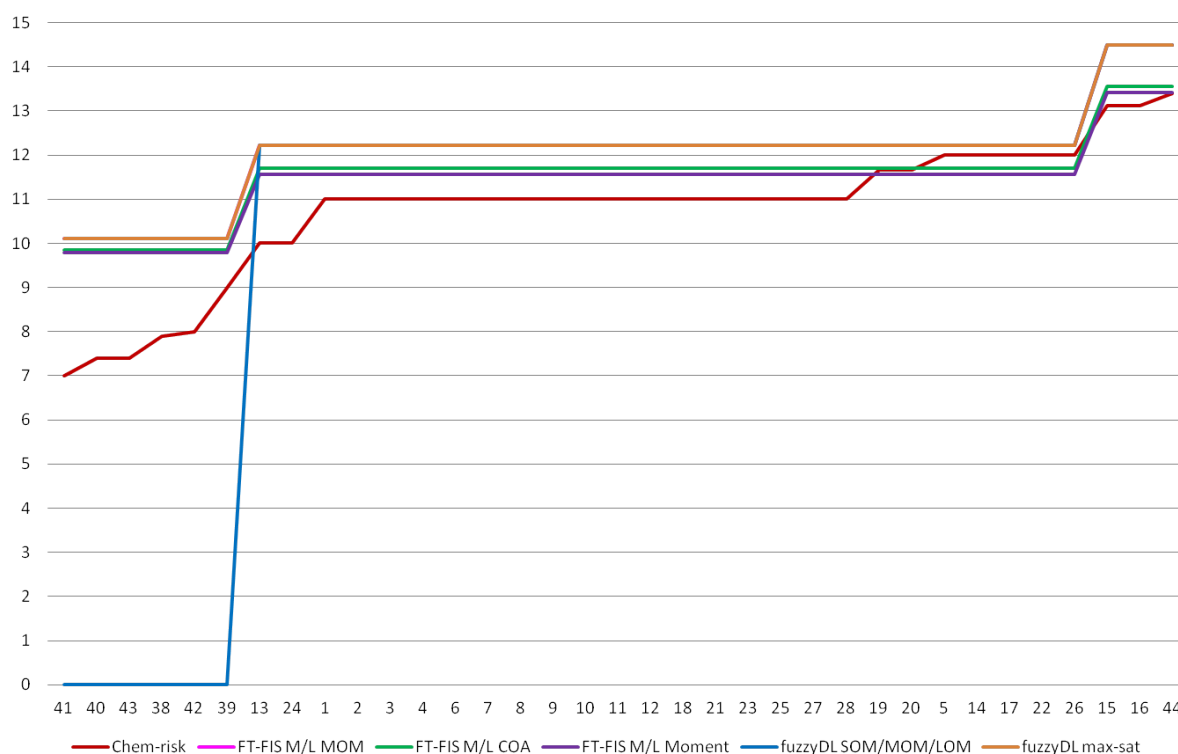
```
(define-concept MaximumIntervention
  (and RuleSet (not (<= hasConcentrationRisk 100)) (some hasAggregateRisk maximumAR)))

(defuzzify-som? MaximumIntervention analysisBeef1 hasAggregateRisk )
(defuzzify-mom? MaximumIntervention analysisBeef1 hasAggregateRisk )
(defuzzify-lom? MaximumIntervention analysisBeef1 hasAggregateRisk )

(min-sat? MaximumIntervention analysisBeef1 )
(max-sat? MaximumIntervention analysisBeef1 )
(show-concrete-fillers-for analysisBeef1 hasAggregateRisk)
```

Os resultados dessas consultas também são apresentados graficamente na Figura 6.9. Os valores de risco agregado inferidos por FT-FIS com MOM, por fuzzyDL com regras de Mamdani e SOM, MOM e LOM e, ainda, pela consulta por *max-sat* em fuzzyDL são idênticos, por esta razão seus gráficos estão sobrepostos na figura e, portanto, somente o gráfico de *max-sat* é visível. Estes resultados ocorrem porque os conjuntos *fuzzy* inferidos a partir das regras *fuzzy* possuem um único ponto máximo, que defuzificado por SOM, MOM e LOM geram os mesmos valores.

Para as análises 41 a 39, o risco agregado não é inferido por fuzzyDL com regras de Mamdani e SOM, MOM e LOM devido à inconsistência da ontologia *fuzzy*. Note que as demais abordagens não resultam em inconsistência, possivelmente causada por algum problema de precisão com relação à implementação de MILP considerada nas consultas de defuzzificação em fuzzyDL. Por fim, considerando todas as abordagens analisadas, os resultados de FT-FIS



**Figura 6.9: Resultados de FT-FIS e fuzzyDL para consulta 2 e entradas numéricas.**

combinando Mamdani ou Larsen com COA ou *Moment* proporcionaram os valores de risco agregado mais aproximados de Chem-risk, seguindo as restrições sobre a saída do SIF estabelecidas na definição do conceito *MaximumIntervention* na ontologia *fuzzy*.

De modo geral, a partir dos resultados obtidos com HyFOM, FT-FIS e fuzzyDL para as consultas realizadas, é possível concluir que os métodos de Mamdani ou Larsen, combinados com defuzzificação COA ou *Moment*, obtiveram as melhores aproximações com relação ao método Chem-risk. Essas combinações de métodos de raciocínio e de defuzzificação estão disponíveis nas implementações de HyFOM e de FT-FIS, porém não são oferecidas por fuzzyDL.

### 6.3.3 Experimentos com Entradas *Fuzzy*

O objetivo dos experimentos com entradas *fuzzy* é mostrar situações em que as inferências da ontologia *fuzzy* contribuem para definir as entradas do SIF, com foco na integração *ontologia fuzzy*  $\rightarrow$  SIF.

No entanto, com relação à ontologia *fuzzy* e às regras *fuzzy* da Seção 6.3.1, a maioria das propriedades concretas numéricas está associada a predicados definidos como conjuntos *crisp*, em função dos dados disponíveis para os testes. A única propriedade que envolve predicados *fuzzy* é o risco de concentração (*concentrationRisk*), que é obtido diretamente por meio de

medições laboratoriais de concentração de analitos em amostras de alimentos. Por se tratar de um valor resultante de medições específicas, não é possível modelar um conceito abstrato que generalize os valores de concentração, diferentemente do que foi modelado com as preferências em função das categorias de atrações turísticas.

Embora haja um esforço em conjunto com os especialistas do PNCRC para estender as definições das propriedades e considerar mais termos linguísticos, este é um trabalho ainda em andamento, que requer maior disponibilidade de dados e de tempo dos especialistas envolvidos. Desta forma, não foram realizadas consultas com entradas *fuzzy* no domínio de risco químico de analitos, devido às características das informações disponíveis.

Um exemplo de consulta que explora inferências da ontologia, embora não sejam necessariamente *fuzzy*, envolve o conceito de intervenção preventiva (*Preventive Intervention*), representada na ontologia *fuzzy* ilustrada nas Tabelas 6.15 e 6.16. Este conceito representa análises com risco aproximadamente médio, porém com uma tendência de concentração para agravar em curto prazo. Intervenções preventivas podem evitar que essas análises tornem-se não-conformes no futuro. Neste contexto, é possível representar conceitos referentes a tipos de analito que influenciem nos valores de tendência de concentração. Por exemplo, analitos da categoria de Micotoxinas podem, por definição, possuir uma tendência de estabilização da concentração detectada em uma amostra. Assim, é possível inferir que análises contendo analitos do tipo Micotoxina possuam tendência da concentração com valor 0, que corresponde à tendência de diminuir ou estabilizar a concentração detectada, conforme modelado na ontologia. Esta é uma das entradas do SIF que podem ser obtidas por meio de inferências da ontologia, embora não envolvam predicados *fuzzy*.

É evidente que esse tipo de informação depende do conhecimento do domínio, portanto a interação com especialistas é fundamental para modelar a ontologia e obter inferências consistentes. Inspiradas neste exemplo, pesquisas futuras podem tratar a aplicação de HyFOM e FT-FIS para oferecer suporte a intervenções preventivas envolvendo categorias de analitos.

## 6.4 Considerações Finais

Neste capítulo, foram apresentados os experimentos realizados com HyFOM e FT-FIS em dois domínios de aplicação distintos. Foram realizadas consultas com entradas numéricas, entradas *fuzzy* e com restrição dos valores inferidos pelas regras *fuzzy*. Os resultados obtidos foram comparados aos resultados de fuzzyDL com o objetivo de analisar tanto a semântica do raciocínio baseado em regras *fuzzy* e quanto a integração com as inferências da ontologia *fuzzy*.



Em termos de semântica das inferências baseadas em regras *fuzzy*, os experimentos mostraram que os métodos de Mamdani e de Larsen combinados com defuzzificação COA e Moment produzem os resultados mais apropriados quando comparados aos resultados de fuzzyDL. As regras implicativas de fuzzyDL podem ser muito restritivas, dependendo da semântica do domínio, resultando em inconsistências em algumas situações em que HyFOM e FT-FIS produzem resultados pertinentes. As regras de Mamdani utilizadas em conjunto com os métodos de defuzzificação disponíveis em fuzzyDL descartam informação relevante ao tratar somente intervalos de grau máximo dos conjuntos *fuzzy* resultantes do raciocínio baseado em regras *fuzzy*. Além disso, fuzzyDL com consultas de satisfazibilidade também proveem resultados limitados, muitas vezes similares aos resultados obtidos com as outras abordagens de fuzzyDL.

Com relação à integração das inferências, os resultados mostraram que tanto HyFOM quanto FT-FIS são capazes de considerar tanto a integração  $SIF \rightarrow \text{ontologia fuzzy}$  quanto  $\text{ontologia fuzzy} \rightarrow SIF$ . Dentre as abordagens analisadas, somente fuzzyDL com regras implicativas apresentaram alguns resultados incoerentes em função de problemas na integração das inferências.

A partir dos resultados obtidos com os experimentos realizados e da análise dos trabalhos correlatos, o próximo capítulo descreve as conclusões da tese, discutindo as principais contribuições e limitações das abordagens desenvolvidas. Além disso, são apresentadas as possibilidades de expansão da pesquisa com trabalhos futuros.

# Capítulo 7

## CONCLUSÕES E TRABALHOS FUTUROS

---

---

### 7.1 Considerações Iniciais

Voltadas para representação de conhecimento, ontologias visam expressar a semântica de domínios específicos por meio de modelos compartilhados tanto por humanos quanto por sistemas computacionais. Em especial, ontologias *fuzzy* são utilizadas para modelar informações cujas definições não são claras nem precisas, ao abordarem a representação e o processamento de imprecisão presente em alguns domínios, inclusive na linguagem humana.

Alguns tipos de aplicação demandam o tratamento de imprecisão sob o ponto de vista conceitual, referente aos principais conceitos e relacionamentos do domínio, como também envolvendo atributos ou propriedades concretas. Dependendo da semântica do domínio, uma representação que permita expressar a transição gradual entre os diferentes estados dessas propriedades pode ser mais apropriada que aquela que faz uso de limites rígidos ou bem definidos. Neste sentido, ontologias *fuzzy* podem ser combinadas com regras *fuzzy* para inferir valores de propriedades numéricas em função de outras propriedades, levando em consideração a imprecisão na definição de seus estados. Esse mecanismo de raciocínio aproximado, utilizado com sucesso no contexto de sistemas de inferência *fuzzy*, pode contribuir para prover inferências expressivas em conjunto com o conhecimento conceitual modelado por ontologias *fuzzy* (LOIA, 2011).

Para abordar estas questões, o objetivo desta tese de doutorado foi investigar a integração do raciocínio de ontologias *fuzzy* e de sistemas de inferência *fuzzy*, considerando aspectos como a semântica do raciocínio baseado em regras *fuzzy* e a colaboração entre as inferências de ambos tipos de raciocínio. Neste capítulo, são apresentadas conclusões sobre as contribuições realizadas, além das limitações evidenciadas na condução da pesquisa e possíveis trabalhos futuros.

## 7.2 Contribuições

A principal contribuição do trabalho de pesquisa realizado e descrito nesta tese de doutorado é a definição de abordagens capazes de combinar o raciocínio baseado em ontologias *fuzzy* formais com o raciocínio de sistemas de inferência *fuzzy* por meio de arquiteturas flexíveis e adaptáveis à semântica das regras *fuzzy* mais apropriada para o domínio de aplicação. Como resultado desta pesquisa, foram definidas e desenvolvidas as abordagens HyFOM e FT-FIS.

O motor de inferência HyFOM baseia-se em uma arquitetura híbrida cujos componentes são motores de inferência para ontologias, ontologias *fuzzy* e sistemas de inferência *fuzzy*. Com relação ao raciocínio baseado em regras *fuzzy*, a arquitetura de HyFOM provê um nível de abstração sobre a implementação de SIF, para que seja adaptada ao tipo de SIF mais apropriado de acordo com a semântica do domínio. Como HyFOM inclui o motor de inferência fuzzyDL, o conjunto de inferências envolvendo imprecisão também compreende o conhecimento conceitual representado pela ontologia *fuzzy*. Desta forma, HyFOM considera o raciocínio de ontologias *fuzzy* para obter os valores de entrada para o SIF e disponibiliza os resultados do SIF sob a forma de asserções explícitas na ontologia *fuzzy* com a supervisão de especialistas. Portanto, HyFOM trata tanto a integração  $SIF \rightarrow ontologia\ fuzzy$  quanto  $ontologia\ fuzzy \rightarrow SIF$ , diferentemente de alguns trabalhos correlatos, conforme discutido no Capítulo 4. Os resultados obtidos por HyFOM no domínio de recomendação de atrações turísticas foram publicados por Yaguinuma et al. (2013) e no domínio de risco químico em alimentos por Yaguinuma et al. (2013b, 2013a).

FT-FIS, por sua vez, aborda uma arquitetura flexível que estende as propostas de Pan (2007) e de Wang e Ma (2008) para tratar os requisitos específicos da integração entre um algoritmo baseado em *tableau fuzzy* e sistemas de integração *fuzzy*. Em FT-FIS, a arquitetura de integração também foi projetada para ser flexível e adaptável à implementação de SIF mais apropriada para o domínio, não restrita a métodos específicos de raciocínio baseado em regras *fuzzy*. A integração realizada por FT-FIS é mais fortemente acoplada que em HyFOM, pois o raciocínio de SIF é invocado, quando necessário, por um algoritmo baseado em *tableau fuzzy*, ou seja, nas situações em que o predicado *fuzzyRuleReasoning* for utilizado na ontologia *fuzzy*. Esse predicado, definido pela abordagem FT-FIS, permite associar propriedades concretas numéricas da ontologia *fuzzy* com entradas e saídas de SIF, sendo que seus valores podem ser representados por conjuntos *fuzzy*. Como o SIF é acionado depois da aplicação das regras de expansão do *tableau fuzzy*, os valores submetidos ao SIF podem resultar do raciocínio de ontologias *fuzzy*. Os resultados do SIF são tratados como conjuntos *fuzzy*, para que sejam devidamente combinados com os predicados *fuzzy* modelados na ontologia. Desta forma, FT-FIS também atende aos requisitos de integração  $ontologia\ fuzzy \rightarrow SIF$  e  $SIF \rightarrow ontologia\ fuzzy$ , além de ser flexível

com relação à semântica do raciocínio baseado em regras *fuzzy*.

Os experimentos realizados nos domínios de recomendação de atrações turísticas e de risco químico em alimentos mostraram que HyFOM e FT-FIS instanciados com SIF sob a semântica de Mamdani ou de Larsen, combinados com o método de defuzzificação *Moment*, obtiveram resultados mais apropriados com relação à semântica das aplicações que fuzzyDL, o principal trabalho correlato de acordo com as características analisadas no Capítulo 4. Os resultados das abordagens de fuzzyDL para processamento de regras *fuzzy* e obtenção de valores de propriedades concretas numéricas apresentaram algumas limitações quanto aos requisitos dos domínios considerados:

- Implicações *fuzzy*: produziram inferências mais restritivas para o contexto de recomendação de atrações turísticas, com diversas situações de inconsistência da ontologia *fuzzy* em função do uso de implicações como semântica das regras e da combinação dos resultados das regras por conjunção. De acordo com Guillaume e Charnomordic (2012), a semântica das regras implicativas considera que a combinação de várias regras conduz a um conjunto mais restrito de situações possíveis que uma única regra. Dependendo do domínio, esta abordagem pode ser adequada, porém não foi o caso dos domínios considerados nos experimentos, cujos resultados mais significativos foram obtidos tanto com a semântica de Mamdani quanto com a de Larsen, que combinam as regras de forma disjuntiva, ampliando o escopo de cada regra;
- Regras de Mamdani e defuzzificação: em comparação com os resultados de HyFOM e FT-FIS sob semântica de regras de Mamdani, a diferença entre os resultados foi influenciada pelos métodos de defuzzificação disponíveis. Os métodos providos por fuzzyDL (SOM, MOM, LOM) são baseados em graus máximos, portanto desconsideram o formato do conjunto *fuzzy* resultante do raciocínio baseado em regras *fuzzy*. Esta é a razão pela qual se verificou perda de informação com relação aos resultados providos pelos métodos COA e *Moment*, considerados em HyFOM e FT-FIS, que geram valores numéricos em função da área do conjunto *fuzzy*, expressando melhor o equilíbrio entre as diversas regras *fuzzy* ativadas pelo raciocínio de SIF;
- Regras de Mamdani e consulta por satisfazibilidade: em geral, os resultados verificados nos experimentos são similares aos obtidos com as regras de Mamdani e defuzzificação de fuzzyDL. O grau de verdade retornado pelas consultas por satisfazibilidade máxima pode ser utilizado para classificar situações em que valores inferidos a partir das regras *fuzzy* são idênticos. Mesmo assim, observou-se que os resultados foram menos significativos,

com relação à semântica das aplicações, que os resultados de HyFOM e FT-FIS, tanto na recomendação de atrações turísticas quanto no domínio de risco químico em alimentos.

Embora as abordagens definidas no âmbito deste doutorado tenham obtido resultados pertinentes nos experimentos, as abordagens de fuzzyDL podem ser adequadas para outros contextos de aplicação, dependendo do domínio em questão. As regras implicativas ou mesmo os métodos de defuzzificação baseados em graus máximos podem atender a determinados domínios que demandem esse tipo de semântica.

Neste contexto, a contribuição de HyFOM e FT-FIS está nas arquiteturas flexíveis, baseadas em FML, que podem ser adaptadas para o tipo de raciocínio baseado em regras *fuzzy* mais apropriado para o domínio. Essa característica não é tratada por fuzzyDL, que oferece suporte apenas para métodos de raciocínio e de defuzzificação que sejam representados como problemas de otimização MILP (*Mixed Integer Linear Programming*), como o caso das regras de Mamdani e defuzzificação SOM, MOM e LOM, além da verificação de graus máximos e mínimos de satisfazibilidade. O método de Larsen, por exemplo, não é tratado por fuzzyDL, pois a operação de produto resulta em um problema de otimização MIQCP (*Mixed Integer Quadratically Constrained Programming*) (BOBILLO; STRACCIA, 2009), que não é solucionado por um otimizador MILP. Com relação ao raciocínio baseado em regras *fuzzy*, HyFOM e FT-FIS foram definidos para tratar os problemas que são solucionados por sistemas de inferência *fuzzy*, permitindo a instanciação de implementações de SIF de forma independente do raciocínio de ontologias *fuzzy*. Assim, é possível considerar métodos de inferência de Mamdani ou de Larsen, métodos de defuzzificação baseados no formato dos conjuntos *fuzzy*, entre outros métodos, desde que estejam em conformidade com os requisitos do domínio.

Com relação a outros trabalhos correlatos discutidos no Capítulo 4, como as abordagens baseadas em FML, HyFOM e FT-FIS destacam-se por explorarem não somente o raciocínio baseado em regras *fuzzy* mas também o raciocínio de ontologias *fuzzy*. Enquanto alguns trabalhos consideram ontologias *crisp* e outros utilizam ontologias *fuzzy* que não são fundamentadas por formalismos lógicos, HyFOM inclui o motor de inferência fuzzyDL e FT-FIS considera a Lógica de Descrição *fuzzy*  $F-ALC(G)$  e um algoritmo baseado em *tableau fuzzy* para o tratamento de inferências de ontologias *fuzzy*. De fato, conforme discutido em outros capítulos, a definição de valores necessários para a inferência de SIF pode depender de inferências de ontologias *fuzzy*, assim como algumas tarefas de raciocínio de ontologias *fuzzy* podem levar em consideração os resultados de SIF. Essas situações foram abordadas nos experimentos, mostrando que HyFOM e FT-FIS são capazes de processar tanto conhecimento impreciso envolvendo regras *fuzzy* quanto conhecimento conceitual a partir do raciocínio de ontologias *fuzzy*.

Além das principais contribuições desta tese de doutorado, foram obtidos resultados referentes a estudos preliminares com foco na representação de ontologias *fuzzy*. Neste contexto, foi definida uma meta-ontologia para representar conceitos, relacionamentos e propriedades *fuzzy* (YAGUINUMA et al., 2010a, 2010b, 2012, 2014), que pode ser instanciada por linguagens de ontologias tradicionais, evitando a modificação das linguagens e, conseqüentemente, incompatibilidade com aplicações existentes. A meta-ontologia foi aplicada em conjunto com sistemas de classificação baseados em regras *fuzzy* e técnicas de mineração de texto para classificar documentos de acordo com categorias e propriedades *fuzzy* modeladas na ontologia. Embora os estudos preliminares não tenham explorado especificamente aspectos da integração de raciocínio de ontologias *fuzzy* e de regras *fuzzy*, eles foram essenciais para identificar as situações que demandam esse tipo de integração. A partir dos requisitos identificados, foi possível definir e desenvolver as duas abordagens que combinam o raciocínio de ontologias *fuzzy* com sistemas de inferência *fuzzy*, no âmbito desta tese de doutorado.

Por fim, vale destacar que o trabalho de pesquisa realizado e descrito nesta tese de doutorado obteve duas premiações ao longo de seu desenvolvimento:

- *Distinguished paper award* no *Joint 5th International Workshop on Vocabularies, Ontologies and Rules for The Enterprise (VORTE)* and *International Workshop on Metamodels, Ontologies and Semantic Technologies (MOST)* em 2010 e
- *Best Student Paper Award in the Area of Artificial Intelligence and Decision Support Systems* na *International Conference on Enterprise Information Systems (ICEIS)* em 2013.

## 7.3 Limitações e Trabalhos Futuros

Normalmente, a delimitação do escopo do trabalho e as decisões estratégicas durante o desenvolvimento da pesquisa podem implicar em limitações que direcionam futuros trabalhos. Assim, as abordagens definidas ao longo da pesquisa realizada apresentam algumas limitações e indicações de possíveis trabalhos futuros:

- HyFOM:
  - *Adição de novas asserções na ontologia fuzzy*: embora necessária para arquiteturas que combinam motores de inferência independentes, a modificação do estado da ontologia *fuzzy* altera o conjunto de interpretações possíveis, o que pode ser inapropriado para alguns domínios. Diversos trabalhos da literatura também seguem essa estratégia, como algumas abordagens baseadas em FML e SWRLTab (O'CONNOR et

al., 2005), que permite executar regras SWRL e incorporar seus resultados em uma ontologia *crisp*. Para contornar essa limitação, SWRLTab baseia-se na interação com o usuário para supervisionar o processo e validar as inferências incorporadas explicitamente, uma abordagem também adotada em HyFOM. Essa limitação não ocorre em FT-FIS, pois as inferências providas pelo SIF são consideradas somente durante o processo de expansão do *tableau fuzzy*, sendo representadas de forma implícita por meio dos nós de tipos de dados *fuzzy* gerados pelas regras de expansão;

- *Supervisão de especialistas*: a dependência por especialistas ao adicionar novas asserções na ontologia *fuzzy* pode limitar a aplicabilidade de HyFOM em alguns domínios. Para reduzir a interação com especialistas, HyFOM pode ser estendido para verificar a consistência da ontologia *fuzzy* a cada nova asserção adicionada, para que sejam supervisionadas somente as situações que gerarem inconsistências. Porém, essa funcionalidade deve ser opcional, pois alguns domínios mais restritivos podem exigir que as asserções sejam verificadas mesmo quando não produzam um estado inconsistente da ontologia *fuzzy*;
- *Defuzificação da saída do SIF*: pode haver perda de informação em comparação com uma saída sob a forma de conjunto *fuzzy*. O valor defuzificado, por ser mais restrito, pode não ter interseção com os predicados *fuzzy* modelados na ontologia, enquanto que a saída do SIF como conjunto *fuzzy* poderia ter. Algumas dificuldades para se adicionar uma asserção envolvendo um conjunto *fuzzy* proveniente do SIF são: (1) como o especialista poderia interpretar o conjunto *fuzzy* para validar a nova asserção e (2) como representar conjuntos *fuzzy* irregulares resultantes do SIF, pois fuzzyDL representa apenas conjuntos *fuzzy* normais. Essa limitação é tratada por FT-FIS com o apoio do FuzzyJ Toolkit (ORCHARD, 2001), que oferece suporte para representar conjuntos *fuzzy* resultantes do raciocínio de SIF e realizar operações sobre os mesmos;

- FT-FIS:

- *Expressividade da ontologia fuzzy*: por considerarem o formalismo de  $F\text{-}\mathcal{ALC}(G)$ , as ontologias *fuzzy* em FT-FIS não tratam relacionamentos abstratos transitivos, reflexivos ou inversos, tampouco restrições de cardinalidade. Para abordar esses e outros construtores de conceito e de relacionamento abstratos, pretende-se generalizar FT-FIS considerando outras Lógicas de Descrição *fuzzy* mais expressivas, assim como a teoria de grupos de tipos de dados *fuzzy* foi gradativamente incorporada a *fuzzy*  $\mathcal{ALC}$  (WANG; MA, 2008), *fuzzy*  $\mathcal{SHOIN}$  (WANG et al., 2008) e a *fuzzy*

*SROIQ* (YAN; ZHANG; MA, 2012). Para tanto, deve-se investigar a influência do aumento de expressividade do formalismo da ontologia *fuzzy* na abordagem de integração com SIF;

- *Análise formal das propriedades da abordagem de integração*: o algoritmo baseado em *tableau fuzzy* para  $F\text{-}\mathcal{ALC}(G)$  é decidível (WANG; MA, 2008), assim como os métodos de raciocínio de SIF. Deste modo, a abordagem de integração de FT-FIS combina dois mecanismos de inferência decidíveis. Embora os experimentos tenham mostrado que as inferências obtidas por FT-FIS foram pertinentes, é necessário analisar formalmente as propriedades de correção, completude, término, decidibilidade e complexidade computacional de FT-FIS, que são importantes para a definição de procedimentos de raciocínio em Lógicas de Descrição;
- *Consultas com restrição de grau*: como o método de raciocínio de  $F\text{-}\mathcal{ALC}(G)$  depende de restrições de grau de verdade para verificar a consistência da ontologia *fuzzy*, as consultas para FT-FIS também devem ser especificadas com inequações do tipo  $\bowtie n$ , onde  $\bowtie \in \{<, \leq, >, \geq\}$  e  $n \in [0, 1]$ . As restrições de grau também são necessárias em outros procedimentos de inferência de ontologias *fuzzy*, como os propostos por Stoilos, Stamou e Pan (2010) e por Bobillo, Delgado e Gómez-Romero (2012). No caso de fuzzyDL, as consultas são baseadas em graus mínimos e máximos, pois o otimizador MILP é capaz de obter os valores mínimos e máximos de funções lineares envolvendo os graus de verdade da ontologia *fuzzy*. Futuras extensões de FT-FIS podem considerar consultas desse tipo, para evitar que o usuário tenha que definir restrições para graus específicos.

Além de abordar as limitações mencionadas, trabalhos futuros devem considerar aplicações envolvendo dados reais, para verificar a aplicabilidade de HyFOM e FT-FIS em contextos práticos. Especificamente, a aplicação no domínio de risco químico em alimentos pode ser estendida para tratar diferentes conjuntos de regras *fuzzy* voltados para outros tipos de risco, como o risco da amostra e o risco de associações químicas perigosas. Outro ponto a ser investigado é a fuzzificação dos fatores de risco, a fim de proporcionar inferências que explorem de modo mais abrangente a imprecisão presente no domínio. Para tanto, há pesquisas em andamento com especialistas da Embrapa Gado de Leite, que podem resultar em um projeto de pós-doutorado.

Com relação aos experimentos realizados, trabalhos futuros também devem investigar de forma mais detalhada os motivos pelos quais alguns resultados de fuzzyDL não foram coerentes com os resultados esperados. Em especial, devem ser analisadas as situações de inconsistência



nos testes envolvendo regras de Mamdani e defuzificação, além dos resultados de fuzzyDL que não consideraram as entradas *fuzzy* de forma apropriada.

Por fim, outra atividade em andamento é a escrita de artigos para periódicos relacionados à área de representação de conhecimento. O artigo *Hybrid Integration of Fuzzy Ontology Reasoning and Fuzzy Inference System*, que aborda a generalização de HyFOM para instanciação de diferentes tipos de SIF, está em fase final de escrita. Além disso, está sendo elaborado o artigo *FT-FIS: A Reasoning Architecture Combining Fuzzy Tableau and Fuzzy Inference System* que descreve a abordagem FT-FIS, os experimentos realizados e seus resultados.

## REFERÊNCIAS BIBLIOGRÁFICAS

- ACAMPORA, G.; LOIA, V. Fuzzy control interoperability and scalability for adaptive domotic framework. *IEEE Trans. Industrial Informatics*, v. 1, n. 2, p. 97–111, 2005.
- AGARWAL, S.; HITZLER, P. Modeling fuzzy rules with description logics. In: *OWLED Workshop on OWL: Experiences and Directions*. Galway, Ireland: CEUR-WS.org, 2005.
- BAADER, F. Appendix 1: Description logic terminology. In: *The Description Logic Handbook*. New York, USA: Cambridge University Press, 2003. p. 495–505.
- BAADER, F.; HORROCKS, I.; SATTLER, U. Description Logics. In: van Harmelen, F.; LIFSCHITZ, V.; PORTER, B. (Ed.). *Handbook of Knowledge Representation*. Oxford, UK: Elsevier, 2007.
- BAADER, F.; NUTT, W. Basic description logics. In: *The Description Logic Handbook*. New York, USA: Cambridge University Press, 2003. p. 43–95.
- BERNERS-LEE, T.; HENDLER, J.; LASSILA, O. The semantic web. *Scientific American*, v. 284, n. 5, p. 34–43, 2001.
- BLANCO, I. J.; MARÍN, N.; MARTÍNEZ-CRUZ, C.; MIRANDA, M. A. V. About the use of ontologies for fuzzy knowledge representation. In: *Conference of the European Society for Fuzzy Logic and Technology*. Barcelona, Spain: [s.n.], 2005. p. 106–111.
- BOBILLO, F.; DELGADO, M.; GÓMEZ-ROMERO, J. DeLorean: A reasoner for fuzzy OWL 2. *Expert Systems with Applications*, v. 39, n. 1, p. 258–272, 2012.
- BOBILLO, F.; DELGADO, M.; GÓMEZ-ROMERO, J.; STRACCIA, U. Fuzzy description logics under Gödel semantics. *International Journal of Approximate Reasoning*, v. 50, n. 3, p. 494–514, 2009.
- BOBILLO, F.; DELGADO, M.; GÓMEZ-ROMERO, J.; LÓPEZ, E. A semantic fuzzy expert system for a fuzzy balanced scorecard. *Expert Systems with Applications*, v. 36, n. 1, p. 423–433, 2009.
- BOBILLO, F.; STRACCIA, U. fuzzyDL: An expressive fuzzy description logic reasoner. In: *International Conference on Fuzzy Systems*. Hong Kong, China: IEEE Computer Society, 2008. p. 923–930.
- BOBILLO, F.; STRACCIA, U. Fuzzy description logics with general t-norms and datatypes. *Fuzzy Sets and Systems*, v. 160, n. 23, p. 3382–3402, 2009.

- BOBILLO, F.; STRACCIA, U. Representing fuzzy ontologies in OWL 2. In: *IEEE International Conference on Fuzzy Systems*. Barcelona, Spain: IEEE Computer Society, 2010. p. 2695–2700.
- BOBILLO, F.; STRACCIA, U. Fuzzy ontology representation using OWL 2. *International Journal of Approximate Reasoning*, v. 52, n. 7, p. 1073–1094, 2011.
- BRAGAGLIA, S.; CHESANI, F.; CIAMPOLINI, A.; MELLO, P.; MONTALI, M.; SOTTARA, D. An hybrid architecture integrating forward rules with fuzzy ontological reasoning. In: *Hybrid Artificial Intelligence Systems*. Berlin / Heidelberg: Springer, 2010, (Lecture Notes in Computer Science, v. 6076). p. 438–445.
- BUNGE, M. *Treatise on Basic Philosophy: Volume 3: Ontology I: The Furniture of the World*. 1. ed. New York, NY: Springer, 1977.
- CORDÓN, O.; JESUS, M. J. D.; HERRERA, F. A proposal on reasoning methods in fuzzy rule-based classification systems. *International Journal of Approximate Reasoning*, v. 20, n. 1, p. 21–45, 1999.
- DAMÁSIO, C. V.; PAN, J. Z.; STOILOS, G.; STRACCIA, U. Representing uncertainty in RuleML. *Fundamenta Informaticae*, v. 82, n. 3, p. 265–288, 2008.
- GANGEMI, A.; PRESUTTI, V. Ontology design patterns. In: STAAB, S.; STUDER, R. (Ed.). *Handbook on Ontologies*. Berlin / Heidelberg: Springer, 2009, (International Handbooks on Information Systems). p. 221–243.
- GRAU, B. C.; HORROCKS, I.; MOTIK, B.; PARSIA, B.; PATEL-SCHNEIDER, P.; SATTLER, U. OWL 2: The next step for OWL. *Web Semantics: Science, Services and Agents on the World Wide Web*, Elsevier, Amsterdam, The Netherlands, v. 6, p. 309–322, 2008.
- GRUBER, T. R. A translation approach to portable ontology specifications. *Knowledge Acquisition*, v. 5, p. 199–220, 1993.
- GRUBER, T. R. Ontology. In: LIU, L.; ÖZSU, M. T. (Ed.). *Encyclopedia of Database Systems*. New York, NY, USA: Springer, 2009. p. 1963–1965.
- GUILLAUME, S.; CHARNOMORDIC, B. Fuzzy inference systems: An integrated modeling environment for collaboration between expert knowledge and data using FisPro. *Expert Systems with Applications*, v. 39, n. 10, p. 8744–8755, 2012.
- GUIZZARDI, G. *Ontological Foundations for Structural Conceptual Models*. Tese (Doutorado) — Centre for Telematics and Information Technology, University of Twente, 2005.
- HORRIDGE, M.; BECHHOFFER, S. The OWL API: A java API for OWL ontologies. *Semantic Web*, v. 2, n. 1, p. 11–21, 2011.
- HORROCKS, I. Implementation and optimisation techniques. In: *The Description Logic Handbook*. New York, USA: Cambridge University Press, 2003. p. 313–355.
- HORROCKS, I. Ontologies and the semantic web. *Communications of the ACM*, ACM, New York, NY, USA, v. 51, p. 58–67, 2008.

- HORROCKS, I.; SATTLER, U. Ontology reasoning in the  $\mathcal{SHOQ}(D)$  description logic. In: *Proc. of the 17th Int. Joint Conf. on Artificial Intelligence (IJCAI 2001)*. Los Altos: Morgan Kaufmann, 2001. p. 199–204.
- HUANG, H.-D.; ACAMPORA, G.; LOIA, V.; LEE, C.-S.; KAO, H.-Y. Applying FML and fuzzy ontologies to malware behavioural analysis. In: *IEEE International Conference on Fuzzy Systems*. Taipei: IEEE Computer Society, 2011. p. 2018–2025.
- KLIR, G. J.; YUAN, B. *Fuzzy Sets and Fuzzy Logic - Theory and Applications*. Upper Saddle River, USA: Prentice Hall PTR, 1995.
- KUROZUMI, K.; LAN, S.-T.; WANG, M.-H.; LEE, C.-S.; KAWAGUCHI, M.; TSUMOTO, S.; TSUJI, H. FML-based japanese diet assessment system. In: *IEEE International Conference on Fuzzy Systems*. Hyderabad, India: IEEE Computer Society, 2013. p. 1–6.
- LARSEN, P. M. Industrial applications of fuzzy logic control. *International Journal of Man-Machine Studies*, v. 12, n. 1, p. 3–10, 1980.
- LEE, C.-S.; WANG, M.-H.; ACAMPORA, G.; LOIA, V.; HSU, C.-Y. Ontology-based intelligent fuzzy agent for diabetes application. In: *IEEE Symposium on Intelligent Agents*. [S.l.: s.n.], 2009. p. 16–22.
- LEE, C.-S.; WANG, M.-H.; ACAMPORA, G.; HSU, C.-Y.; HAGRAS, H. Diet assessment based on type-2 fuzzy ontology and fuzzy markup language. *International Journal of Intelligent Systems*, v. 25, n. 12, p. 1187–1216, 2010.
- LEE, C.-S.; WANG, M.-H.; YAN, Z.-R.; CHEN, Y.-J.; DOGHMEN, H.; TEYTAUD, O. FML-based type-2 fuzzy ontology for computer go knowledge representation. In: *International Conference on System Science and Engineering*. [S.l.: s.n.], 2010. p. 63–68.
- LOIA, V. Fuzzy ontologies and fuzzy markup language: A novel vision in web intelligence. In: *Advances in Intelligent Web Mastering*. Berlin / Heidelberg: Springer, 2011, (Advances in Intelligent and Soft Computing, v. 86). p. 3–10.
- LUKASIEWICZ, T.; STRACCIA, U. Managing uncertainty and vagueness in description logics for the semantic web. *Journal of Web Semantics*, v. 6, n. 4, p. 291–308, 2008.
- LUTZ, C. Description logics with concrete domains - a survey. In: BALBIANI, P.; SUZUKI, N.-Y.; WOLTER, F.; ZAKHARYASCHEV, M. (Ed.). *Advances in Modal Logic*. [S.l.]: King's College Publications, 2003. v. 4, p. 265–296.
- MAGALHÃES JÚNIOR, W. C. P. de. *Chem-risk: Avaliação e Gerenciamento de Risco Químico em Alimentos Empregando Técnicas de Lógica Nebulosa e Ontologias*. Dissertação (Mestrado) — Departamento de Computação, Universidade Federal de São Carlos, 2011.
- MAGALHÃES JÚNIOR, W. C. P. de; BONNET, M.; FEIJÓ, L. D.; SANTOS, M. T. P. Risk-off method: Improving data quality generated by chemical risk analysis of milk. In: *Cases on SMEs and Open Innovation: Applications and Investigations*. Hershey, PA: IGI Global, 2012. p. 40–64.
- MAILIS, T. P.; STOILOS, G.; STAMOU, G. B. Expressive reasoning with horn rules and fuzzy description logics. In: *International Conference on Web Reasoning and Rule Systems*. Innsbruck, Austria: Springer, 2007. v. 4524, p. 43–57.

- MAIO, C. de; FENZA, G.; FURNO, D.; LOIA, V.; SENATORE, S. OWL-FC: An upper ontology for semantic modeling of fuzzy control. *Soft Computing*, v. 16, n. 7, p. 1153–1164, 2012.
- MAMDANI, E. H.; ASSILIAN, S. An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, v. 7, n. 1, p. 1–13, 1975.
- MOTIK, B.; SHEARER, R.; HORROCKS, I. Hypertableau reasoning for description logics. *Journal of Artificial Intelligence Research*, v. 36, p. 165–228, 2009.
- NARDI, D.; BRACHMAN, R. J. An introduction to description logics. In: *The Description Logic Handbook*. New York, USA: Cambridge University Press, 2003. p. 5–44.
- NICOLETTI, M. C.; CAMARGO, H. A. *Fundamentos da Teoria de Conjuntos Fuzzy*. 2004. EdUFSCar. Série Apontamentos.
- O'CONNOR, M. J.; KNUBLAUCH, H.; TU, S. W.; GROSOFF, B. N.; DEAN, M.; GROSSO, W. E.; MUSEN, M. A. Supporting rule system interoperability on the semantic web with SWRL. In: *International Semantic Web Conference*. Galway, Ireland: Springer, 2005. p. 974–986.
- ORCHARD, R. Fuzzy reasoning in Jess: The FuzzyJ toolkit and fuzzy Jess. In: *International Conference on Enterprise Information Systems*. Setubal, Portugal: [s.n.], 2001. p. 533–542.
- PAN, J. Z. A flexible ontology reasoning architecture for the semantic web. *Knowledge and Data Engineering, IEEE Transactions on*, v. 19, n. 2, p. 246–260, 2007.
- PAN, J. Z.; HORROCKS, I. Extending datatype support in web ontology reasoning. In: *Proc. of the 2002 Int. Conference on Ontologies, Databases and Applications of Semantics (ODBASE 2002)*. Berlin / Heidelberg: Springer, 2002. (Lecture Notes in Computer Science, 2519), p. 1067–1081.
- PAN, J. Z.; STOILOS, G.; STAMOU, G. B.; TZOUVARAS, V.; HORROCKS, I. f-SWRL: A fuzzy extension of SWRL. *Journal on Data Semantics VI*, Special Issue on Emergent Semantics, p. 28–46, 2006.
- PEDRYCZ, W.; GOMIDE, F. *An Introduction to Fuzzy Sets: Analysis and Design*. Cambridge, USA: MIT Press, 1998.
- REFORMAT, M.; GOLMOHAMMADI, S. Updating user profile using ontology-based semantic similarity. In: *IEEE International Conference on Fuzzy Systems*. Jeju Island, Korea: IEEE Computer Society, 2009. p. 1062–1067.
- SCHMIDT-SCHAUSS, M.; SMOLKA, G. Attributive concept descriptions with complements. *Artificial Intelligence*, v. 48, n. 1, p. 1–26, 1991.
- SCOWEN, R. S. Extended BNF - a generic base standard. In: *Proceedings of the Software Engineering Standards Symposium*. [S.l.: s.n.], 1993. v. 3, p. 6–2.
- STOILOS, G.; SIMOU, N.; STAMOU, G.; KOLLIAS, S. Uncertainty and the semantic web. *IEEE Intelligent Systems*, v. 21, n. 5, p. 84–87, 2006.

- STOILLOS, G.; STAMOU, G.; PAN, J. Fuzzy extensions of OWL: Logical properties and reduction to fuzzy description logics. *International Journal of Approximate Reasoning*, v. 51, n. 6, p. 656–679, 2010.
- STOILLOS, G.; STAMOU, G. B.; PAN, J. Z.; TZOUVARAS, V.; HORROCKS, I. Reasoning with very expressive fuzzy description logics. *Journal of Artificial Intelligence Research*, v. 30, p. 273–320, 2007.
- STOILLOS, G.; STAMOU, G. B.; PAN, J. Z.; SIMOU, N.; TZOUVARAS, V. Reasoning with the Fuzzy Description Logic f-SHIN: Theory, Practice and Applications. In: *ISWC International Workshop on Uncertainty Reasoning for the Semantic Web, Revised Selected and Invited Papers*. Berlin / Heidelberg: Springer, 2008. v. 5327, p. 262–281.
- STRACCIA, U. Reasoning within fuzzy description logics. *Journal of Artificial Intelligence Research*, v. 14, p. 137–166, 2001.
- STRACCIA, U. Description logics with fuzzy concrete domains. In: *Proceedings of the Twenty-First Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-05)*. Arlington, Virginia: AUAI Press, 2005. p. 559–567.
- STRACCIA, U. Fuzzy ALC with fuzzy concrete domains. In: *Proceedings of the International Workshop on Description Logics (DL-05)*. Edinburgh, Scotland: CEUR, 2005. p. 96–103.
- STRACCIA, U. Answering vague queries in fuzzy DL-LITE. In: *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Paris, France: [s.n.], 2006. p. 2238–2245.
- STRACCIA, U. A fuzzy description logic for the semantic web. In: SANCHEZ, E. (Ed.). *Fuzzy Logic and the Semantic Web*. [S.l.]: Elsevier, 2006, (Capturing Intelligence). p. 73–90.
- TAKAGI, T.; SUGENO, M. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man and Cybernetics*, SMC-15, n. 1, p. 116–132, 1985.
- USCHOLD, M.; GRÜNINGER, M. Ontologies and semantics for seamless connectivity. *SIGMOD Record*, v. 33, n. 4, p. 58–64, 2004.
- WANG, H.; MA, Z.; YIN, J. FRESG: A kind of fuzzy description logic reasoner. In: *Database and Expert Systems Applications*. Berlin / Heidelberg: Springer, 2009, (Lecture Notes in Computer Science, v. 5690). p. 443–450. ISBN 978-3-642-03572-2.
- WANG, H.; MA, Z. M. A decidable fuzzy description logic F-ALC(G). In: *International Conference on Database and Expert Systems Applications*. Krakow, Poland: Springer, 2008. p. 116–123.
- WANG, H.; MA, Z. M.; YAN, L.; CHENG, J. A fuzzy description logic with fuzzy data type group. In: *IEEE International Conference on Fuzzy Systems*. Hong Kong, China: IEEE Computer Society, 2008. p. 1534–1541.
- WANG, M.-H.; TSAI, Y.-T.; LIN, K.-H.; LEE, C.-S.; LIU, C.-H. FML-based decision support system for solar energy supply and demand analysis. In: *IEEE International Conference on Fuzzy Systems*. Hyderabad, India: IEEE Computer Society, 2013. p. 1–8.

- WANG, X.; MA, Z. M.; YAN, L.; MENG, X. Vague-SWRL: A fuzzy extension of SWRL. In: *International Conference on Web Reasoning and Rule Systems*. [S.l.: s.n.], 2008. p. 232–233.
- WLODARCZYK, T. W.; O'CONNOR, M.; RONG, C.; MUSEN, M. SWRL-F: A fuzzy logic extension of the Semantic Web Rule Language. In: *International Workshop on Uncertainty Reasoning for the Semantic Web (URSW)*. Shanghai, China: Springer, 2010.
- YAGUINUMA, C. A.; MAGALHÃES JÚNIOR, W. C. P. de; SANTOS, M. T. P.; CAMARGO, H. A.; REFORMAT, M. Combining fuzzy ontology reasoning and Mamdani fuzzy inference system with HyFOM reasoner. In: *Lecture Notes on Business Information Processing*. Berlin / Heidelberg: Springer, 2013. Submetido em setembro de 2013.
- YAGUINUMA, C. A.; MAGALHÃES JÚNIOR, W. C. P. de; SANTOS, M. T. P.; CAMARGO, H. A.; REFORMAT, M. HyFOM reasoner: Hybrid integration of fuzzy ontology and Mamdani reasoning. In: *International Conference on Enterprise Information Systems*. Angers, France: SciTePress, 2013. v. 1, n. 15, p. 372–380.
- YAGUINUMA, C. A.; NOGUEIRA, T. M.; FERRAZ, V. R. T.; SANTOS, M. T. P.; CAMARGO, H. A. A model for representing vague linguistic terms and fuzzy rules for classification in ontologies. In: *International Conference on Enterprise Information Systems*. Funchal, Madeira, Portugal: SciTePress, 2010. v. 2, p. 438–442.
- YAGUINUMA, C. A.; NOGUEIRA, T. M.; SANTOS, M. T. P.; CAMARGO, H. A. A meta-ontology approach for representing vague linguistic terms and fuzzy rules for classification in ontologies. In: *Workshop Proceedings of the 14th IEEE International Enterprise Distributed Object Computing Conference*. Vitória, ES: IEEE Computer Society, 2010. p. 263–271.
- YAGUINUMA, C. A.; SANTOS, M. T. P.; CAMARGO, H. A.; NICOLETTI, M. C.; NOGUEIRA, T. M. Fuzz-Onto: A meta-ontology for representing fuzzy elements and supporting fuzzy classification rules. In: *12th International Conference on Intelligent Systems Design and Applications*. Kochi, India: IEEE Computer Society, 2012. p. 166–171.
- YAGUINUMA, C. A.; SANTOS, M. T. P.; CAMARGO, H. A.; REFORMAT, M. A FML-based hybrid reasoner combining fuzzy ontology and Mamdani inference. In: *IEEE International Conference on Fuzzy Systems*. Hyderabad, India: IEEE Computer Society, 2013. p. 1–8.
- YAGUINUMA, C. A.; SANTOS, M. T. P.; CAMARGO, H. A.; NICOLETTI, M. C.; NOGUEIRA, T. M. A meta-ontology for modeling fuzzy ontologies and its use in classification tasks based on fuzzy rules. *International Journal of Computer Information Systems and Industrial Management Applications*, v. 6, p. 89–101, 2014. Aceito para publicação.
- YAN, L.; ZHANG, F.; MA, Z. M. f-SROIQ(G): an expressive fuzzy description logic supporting fuzzy data type group. In: *Proceedings of the 27th Annual ACM Symposium on Applied Computing*. New York, NY, USA: ACM, 2012. p. 320–325.
- ZADEH, L. Fuzzy logic and approximate reasoning. *Synthese*, Kluwer Academic Publishers, v. 30, n. 3-4, p. 407–428, 1975.
- ZADEH, L. A. Fuzzy sets. *Information and Control*, v. 8, n. 3, p. 338–353, 1965.

## LISTA DE SIGLAS E ABREVIATURAS

---

---

**BTVB** – *Best Truth Value Bound*

**COA** – *Center of Area*

**DLs** – *Description Logics*

**EBNF** – *Extended Backus-Naur Form*

**FML** – *Fuzzy Markup Language*

**FT-FIS** – *Fuzzy Tableau and Fuzzy Inference System*

**HyFOM** – *Hybrid integration of Fuzzy Ontology and Mamdani reasoning*

**JAXP** – *Java API for XML Processing*

**LOM** – *Largest of Maxima*

**MAPA** – *Ministério da Agricultura, Pecuária e Abastecimento*

**MILP** – *Mixed Integer Linear Programming*

**MIQCP** – *Mixed Integer Quadratically Constrained Programming*

**MOM** – *Middle of Maxima*

**NNF** – *Negation Normal Form*

**OWL** – *Web Ontology Language*

**PNCRC** – *Plano Nacional para Controle de Resíduos e Contaminantes*

**SBRF** – *Sistemas Baseados em Regras Fuzzy*

**SIF** – *Sistema de Inferência Fuzzy*

**SOM** – *Smallest of Maxima*

**TSK** – *Takagi-Sugeno-Kang*



**XML** – *eXtensible Markup Language*

**XSLT** – *eXtensible Stylesheet Language Transformations*

**glb** – *greatest lower bound*

**lub** – *least upper bound*