基于约束逻辑的非单调属性委托*)

陈 波[1] 曾国荪[2] 李 莉[1]
(同济大学计算机科学及工程系 上海 201804)[1]
(国家高性能计算机工程技术中心同济分中心 上海 201804)[2]

摘 要 分析了目前分布式协作环境下属性委托模型否定授权能力的现状,引入约束逻辑中否定推理的原理,提出一个支持否定授权的属性委托模型,并给出相应的操作语义,最后对该模型进行可靠性和完备性验证及约束域的分析。理论分析表明,提出的基于约束逻辑否定推理的属性委托模型在扩展的范围约束域上是可行的,与原有的类似模型相比具有更强的表达能力。

关键词 约束逻辑,否定推理,属性访问控制,代理,授权

Non-monotone Attribute Delegation Based on Constraint Logic

CHEN Bo[1] ZENG Guo-sun[2] LI Li[1]
(Department of Computer Science and Technology, Tongji University, Shanghai 201804, China)[1]
(Tongji Branch, National Engineering and Technology Center of High Performance Computer, Shanghai 201804, China)[2]

Abstract The current state of negative authorization ability for most attribute delegation models under distributed coalition environment is analyzed. By introducing the principle of constraint logic with constructive negation, an attribute based delegation model with negation is proposed. The corresponding operation semantics is given. The soundness and completeness of the model are verified and its suitable constraint domain is analyzed. The result of theoretical analysis indicates that the proposed model has more expressive ability than other similar models and is tractable on extended range constraint domain.

Keywords Constraint logic, Constructive negation, Access control, Delegation, Authorization

1 引言

基于属性的访问控制机制(Attribute Based Access Control, ABAC)具有 ABAC 性质,因此被用来解决分布式协作计算环境的安全问题[1]。国内外的学者提出了一些 ABAC 模型,主要有 RT[1], $ABDM_A$[2], DL[3], SD3[4], dRBAC[5],其中 $ABDM_A$ 是在 RBAC 上扩展而来,基于属性的委托模型,有非单调的授权委托能力,但需要一个中心的可信的权威来管理委托,不适应分布式协作环境的需要[5],Li[3]提出的 DL 是一种分布协作计算环境下的 ABAC 模型,但在实际中难于使用,因为它缺乏本地链接的主体名字形式的抽象表达能力,不能清楚简洁地重复表达属性的委托[1],SD3 与 DL 类似,而且没有委托的构造(constructs),在分布式协作环境下具有较好表达能力的 ABAC 模型是 dRBAC 和 RT 家族,前者也是由 RBAC 扩展而来,适用于分布式协作环境,满足 ABAC 的特征,后者从 RT_1 开始支持 ABAC 特性,尤其是基于约束域的 RT_1^C,它支持丰富的属性约束的表达,支持层次资源的授权委托。但实际应用中,我们发现有许多应用场合需要非单调的否定授权委托,而 dRBAC 和 RT 仍然不能满足这些场合授权委托的需求,原因是它们缺乏否定授权委托的表达能力,考虑下面的例子。

例 1:假设某个 Web 图书出版商 Epub 要在高校内对优秀理科学生开展图书打折销售,Epub 把对优秀理科学生的鉴别权威委托给教育管理机构 Eorg,教育管理机构再把这样的权威委托给各个高校,其过程用 RT_1^C[1]的授权模型描述如下

Epub. discount←Eorg. science∩Eorg. excellent
Eorg. science←Eorg. university. science
Eorg. exellent←Eorg. university. excellent
Eorg. university←UA
UA. science←Wand.
UA. excellent(avergrade≥80)←UA. science

如果大学 UA 规定:未受到处分的学生才是优秀学生,上面的授权规则体中关于优秀学生的授权表达要加入描述未受到处分的否定授权谓词,如¬punished,它属于否定授权规则,但 RT_1^c 并不支持这样的授权规则,其语义基础为一种特殊的约束逻辑(constraint logic programming CLP)- datalog^c,程序中的规则子句均为 horn 子句,规则体中不能包含否定谓词,这样实现的授权委托模型,其逻辑程序的操作语义(计算过程)相对简单,可以支持常见的约束域,而且计算的时间复杂度较好[6],但不能满足现实中否定授权委托的要求。

事实上,约束逻辑(CLP)十多年来的研究有了很大的发展,为建立支持否定的授权委托模型提供了基础,1994 年 Jaffar[7]给出了完整的正常 CLP 的操作语义、模型论语义和不动点语义,在正常 CLP 之上构建支持否定推理逻辑(CLP-CN)

*)本课题得到 863 项目(2007AA01Z425),973 计划前期研究专项(2007CB316502),国家自然基金项目(60673157)资助。陈 波 博士研究生,研究领域为分布式安全、Web 服务及可信计算;曾国荪 博士,教授,博导,主要研究领域为网格计算、信息安全;李 莉 博士研究生,研究领域为信息安全、可信计算。

的研究也取得了进展，Lloyd[8]提出仅支持否定实例谓词的SLDNF方法，Chan[9]和Drabent[10]提出基于Herbrand域的非实例谓词否定方法，Stuckey[11]提出一般约束域的CLP-CN方法。

本文考虑扩展RT模型，用更一般的约束逻辑代替datalog^{c}作为模型的语义基础，同时引入CLP中构建否定推理的语义，提出一个属性授权模型，满足ABAC的特征要求，支持非单调的否定授权，而且支持较好的约束域。

2 CLP中否定推理的构造

本节给出的CLP中构造否定推理的操作语义，其原理参照了Stuckey[11]的构造否定推理的思想，基础的CLP的操作语义则参照了Jaffar[7]的工作。我们采用如下符号约定，CLP中的谓词集合表示为$\Pi=\Pi_C\cup\Pi_P$，其中Π_C表示约束谓词的符号集合，Π_P表示程序谓词的符号集合，D是Π上解释，V表示变量的集合，变量或变量的列表用$\tilde{X},\tilde{Y},\cdots$表示，约束或约束集合用$C$表示，以下是几个概念。

定义2.1(约束逻辑程序) 一个CLP(C)程序P是指有限个如下规则的集合：

$$P(\tilde{X})\leftarrow C\wedge\tilde{B}$$

其中$P=P(\tilde{X})$是一个程序谓词原子，称为规则头部，$\tilde{X}=(x_1,x_2,\cdots,x_n)$，$x_i$，$1\leqslant i\leqslant n$，是变量，$\tilde{B}$为$\Pi_P$上正文字合取，$C$为$(\Pi_C,D)$上形为$C=C(\tilde{X})=\varphi_1\wedge\cdots\varphi_n$的约束公式。本文考虑的约束域仍在Li[6]的datalog^{c}约束域范围内，没有函数符，这样的程序具有描述性(declarative)的语义，目的是简化计算，使计算的时间复杂度较好。特殊情形，一个没有头部的规则称为目标。

CLP(C)程序P的推导过程为，从初始目标(状态)$(c_0|\tilde{B}_0)$出发，按照一定的计算规则R，选择状态(目标)中的原子和规则头部进行匹配替换的过程[7]。

定义2.2(P^*理论) 一个CLP(C)程序P按照如下转换得到的公式集合，对程序中所有定义谓词P的子句

$$P(\widetilde{Z_1})\leftarrow c_1\wedge\widetilde{B_1}$$
$$\vdots$$
$$P(\widetilde{Z_n})\leftarrow c_n\wedge\widetilde{B_n}$$

转换为P^*公式

$\forall\tilde{X}(P(\tilde{X})\leftrightarrow\vee_{i=1}^{n}\exists Y_i(\tilde{X}=\tilde{Z}_i\wedge C_j\wedge\tilde{B}_i)$，$1\leqslant i\leqslant n$。其中如果有一个谓词符$P$没有在规则头部出现，则$P*$包含$\forall\tilde{X}\neg P(\tilde{X})$。谓词和逻辑连接符$\wedge,\vee,\neg,\rightarrow,\exists$是三值逻辑{true, false, undefined}谓词和连接词，$\leftrightarrow$为二值连接词[11]。

定义2.3(边界) 一个推导树的边界(frontier)是指CLP(C)程序P的推导树中的一组节点集合，其中不包括根节点，程序的每一个推导过程要么是有限失败，要么经过而且仅经过边界中的一个节点。边界可以从根节点开始，重复对非成功节点代以其孩子节点而得到[11]。

如果状态$\langle c_1|B_1\rangle,\cdots,\langle c_k|B_k\rangle$构成程序从目标$(C|B)$开始的推导树的边界，有$c_rP^*\vdash_3\widetilde{\forall}(c\wedge B\leftrightarrow\exists Y_1C^1\wedge B_1\vee\cdots\vee\exists Y_kC^k\wedge B_k)$ (*)

其中$\vdash_3$表示强三值蕴含[11]。

CLP中的否定指的是规则体中多个程序谓词或约束合取的否定，包含否定的CLP程序称为CLP-CN (constructive negation)程序，具体如下。

定义2.4(否定子目标) 一个否定子目标为形式，$\neg(c|B_1,\cdots,B_n)$，其中B_i是程序谓词或递归形式否定子目标，c是约束。

定义2.5(否定目标) 一个否定目标为$(c|B)$，c为约束，B为多个否定子目标或正常子目标的合取。

Stuckey构造否定推理的思想为，如果当前的目标为$(c|\neg G)$，$G=(D_1,\cdots,D_j,\cdots,D_n)$，$D_i$为正常子目标或否定子目标，对$(c|G)$进行推导，直到其边界，假设计算规则选择对$D_j$进行替换，如果$D_j$为原子，则按正常的分解进行；如果$D_j=\neg(C_q|\tilde{A})$是一个否定子目标，考虑子目标$(C\wedge C_q|\tilde{Q})$的推导，假设其边界为$F=\{(c\wedge c_1|B_1),\cdots,(c\wedge c_m|B_m)\}$，若$F$为空集，则推导从状态$(c|G)$进入状态$(c|D_1,\cdots,D_{j-1},D_{j+1},\cdots,D_n)$，否则，由(*)式有，$C,P^*\vdash_3\widetilde{\forall}(c_q\wedge\tilde{Q}\leftrightarrow\exists\tilde{Y}_1c_1\wedge B_1\vee\cdots\vee\exists\tilde{Y}_kc_k\wedge B_k)$经过下面的析取化

$$C\vdash(\neg\exists Y_1(c_1\wedge B_1)\wedge\cdots\wedge\neg\exists Y_m(c_m\wedge B_m)\leftrightarrow(c'_1\wedge N_1)\vee\cdots\vee(c'_1\wedge N_P)$$

找到从D_j开始推导的边界，如果$c\wedge c'_i$是可满足的，则$(c\wedge c'_i|D_1,\cdots,D_{j-1},N_i,D_{j+1},\cdots,D_n)1\leqslant i\leqslant p$，就是$(c|G)$的孩子节点，得到其边界后，相类似，经过否定得到$(c|\neg G)$的孩子节点。

3 基于CLP否定推理的授权委托

本节将stuckey[11]否定推理的操作语义引入到授权模型中，给出非单调的属性授权模型，该模型比RT1c具有更广的适用范围。

3.1 授权谓词和授权规则

CLP的否定推理中的否定是对规则体中的谓词及约束的否定，规则的头部则是谓词原子，没有否定，考虑到否定授权的实际情形有如下几种：1)否定条件，肯定授权，即当B不满足某些条件时，A授权给B；2)肯定条件，否定授权，即当B满足某些条件时，A不能授权给B；3)否定条件，否定授权，即当B不满足某些条件时，A不能授权给B。因此，实际场合中要求对应的CLP(C)规则中既能在规则体中有否定，形成否定规则体和否定子目标，又能在规则头部进行否定，为此，我们在授权谓词中引入一个特殊的变量ω，用以表示授权的肯定与否。

授权谓词：$R(x,z,\omega,h_1,\cdots,h_m)$，$x,z$表示实体，$h_i$表示实体$z$的属性变量，当$\omega=1$时，若$z$满足某些属性条件，则$x$将授予$z$其$R$属性(权限)，谓词$R=$true；当$\omega=0$时，若$z$满足某些条件时，$x$否决授予$z$某种$R$属性(权限)，谓词$R=$false；其它情形谓词$R$取值undefine，$R$满足Kleene强三值逻辑运算。四种授权策略：

$R(A,D,1,\tilde{H})\leftarrow\psi(\tilde{H})$ (授权事实)

$R(A,D,1,\tilde{H})\leftarrow\neg\psi(\tilde{H})$ (授权事实的否定)

$R(A,D,0,\tilde{H})\leftarrow\psi(\tilde{H})$ (肯定条件，否定授权)

$R(A,D,0,\tilde{H})\leftarrow\neg\psi(\tilde{H})$ (否定条件，否定授权)

类似双否定的情形在下面省略。

$R(A,C,1,\tilde{H})\leftarrow R_1(B,C,1,\tilde{S})$，$\psi(\tilde{H},\tilde{S})$ (授权)，表示A将其属性R授权给B，只要B授权的对象C满足约束ψ。

$R(A,C,1,\tilde{H})\leftarrow\neg R_1(B,C,1,\tilde{S}),\psi(\tilde{H},\tilde{S})$(否定授权)

$R(A,C,0,\tilde{H})\leftarrow R_1(B,C,1,\tilde{S}),\psi(\tilde{H},\tilde{S})$ (否定授权)

$R(A,C,1,\tilde{H})\leftarrow R_1(A,B,1,\tilde{S}),R_2(B,C,1,\tilde{T}),\psi(\tilde{H},\tilde{S},\tilde{T})$(链式委托)

$R(A, C, 1, \widetilde{H}) \leftarrow\!\!\rightarrow R_1(A, B, 1, \widetilde{S}), R_2(B, C, 1, \widetilde{T}), \psi\widetilde{H}, \widetilde{S}, \widetilde{T})$(否定链式委托)

$R(A, C, 0, \widetilde{H}) \leftarrow R_1(A, B, 1, \widetilde{S}), R_2(B, C, 1, \widetilde{T}), \psi(\widetilde{H}, \widetilde{S}, \widetilde{T})$ (否定链式委托)

$R(A, C, 1, \widetilde{H}) \leftarrow R_1(B_1, C, 1, \widetilde{S}), R_2(B_2, C, 1, \widetilde{T}), \psi$(合并委托)

$R(A, C, 1, \widetilde{H}) \leftarrow\!\!\rightarrow R_1(B_1, C, 1, \widetilde{S}), R_2(B_2, C, 1, \widetilde{T}), \psi$(否定合并授权)

$R(A, C, 0, \widetilde{H}) \leftarrow R_1(B_1, C, 1, \widetilde{S}), R_2(B_2, C, 1, \widetilde{T}), \psi$(否定合并授权)

对任何授权谓词 R 有：

$R(A, x, 1, \widetilde{H}) \leftarrow\!\!\rightarrow R(A, x, 0, \widetilde{H})$ 和 $R(A, x, 0, \widetilde{H}) \leftarrow\!\!\rightarrow R(A, x, 1, \widetilde{H})$

以上授权委派关系均由 A 签发，其含义是表达式左部 R 为 A 决定的角色，表示授予的权限。1 类型表达直接授权，2，3，4 类型具有委派性质，如 2 类型，授权的具体对象是由 B 来决定的 R_1 成员。

由以上的关系，看出授权模型具有比 $RT_1^{\neg}$ 更强的授权表达能力，其具有否定授权的表达能力。

3.2 委托模型的操作语义

当实体 A 接到实体 B 的访问请求时，A 需要判断是否应允许 B 访问，这时 A 开始进行授权程序的计算，计算出发点为初始目标。假设逻辑公式具有改名性质，即公式值不受变量名改变的影响，在这一假设下授权程序计算的结果将会受到两种不确定因素影响，规则和文字的选取，在具有否定推理的委托操作计算中，引入文字索引方法，采用宽度优先不依赖于规则的公平文字选取[7]。参考 Stuckey 的 CLP-CN[11] 的思想，我们给出委托模型的否定操作计算的语义。

定义 3.1(文字的索引 index) 每一个文字有一个元组 $\langle i, j\rangle$ i 表示文字第一次出现在推导中的状态，j 表示文字在这个状态中的位置。

设一个委托程序的规则集为 R，当前状态处于推导树的 i 层，当前的目标为否定目标，即推导状态表示为 $S_i = (C_i \mid \neg G)$，C_i 是约束，$G = D_1 \wedge \cdots \wedge D_n$，$D_j$ 是正常子目标或否定子目标，操作计算的过程描述如下：

Negation-Delegation(R, S_i)

Step1 按照类似定义 2.5 标示 G 中每个 D_j 的索引为 $\langle i, j\rangle$。

Step2 把(true|G)按照$\langle i, j\rangle$的顺序分别依次选择子目标 D_j，推导一层，假设有 m 个子目标结果，记为 $(c^1 \mid B_1), \cdots, (c^m \mid B_m)$，并记 $\prod = \neg \exists \widetilde{Y}_i (c^1 \mid B_1) \wedge \cdots \wedge \neg \exists \widetilde{Y}_i (c^m \mid B_m)$，其中 $\widetilde{Y}_i$ 表示在 $(c_i \mid B_i)$ 中出现但不在 G 中出现的变量集合。

Step3 将 step2 的结果析取化为 $\widetilde{D} = (c'_1 \wedge E_1) \vee \cdots \vee c'_l \wedge E_l$，即 $\prod \leftrightarrow \widetilde{D}$。

Step4 取 step3 中使得式子 $c_i \wedge c'_j$ 可满足部分，即与 S_i 状态相容的部分，假设 $\widetilde{D}$ 的前 k 个为可满足的。

Step5 得到 $i+1$ 层的子目标节点为 $(c_i \wedge c'_1 \mid E_1), \cdots, (C_i \wedge C'_k \mid E_k)$。

如果委托模型的子目标为正常子目标，则参照 Jaffar[7] 的宽度优先操作计算推导，我们可以得到委托模型的完整操作语义。

普通 CLP(C)的一次推导，结果可能有成功推导，有限失败和无穷推导三种结果，委托模型看作有实际的授权应用背景的 CLP(C)程序特例，我们要求：不允许出现自授权，即出现 $P \leftarrow \cdots, P, \cdots$. 形式的授权规则，也不能出现间接自授权形式，即出现公式 $P \leftarrow, \cdots, q. \cdots$ 和 $q, \cdots, p, \cdots$. 形式的授权规则，则有：

定理 3.1 假设在每次具体的委托操作推导时，节点 A 收集的策略和证书是有限的，其中涉及的主体名也是有限的，则委托操作的推导不会出现无穷推导，并且采用本节定义的操作语义时，文字选取策略仍然是独立的。

证明：假设在委托的某个推导过程中，出现无穷推导，由于在推导路径上的每个节点处，都至少应用了一次规则的分解，由假设，A 收集的策略和证书有限，在这些策略和证书中的名字个数有限，则谓词符号和实例谓词(谓词变量被赋予某个常量值)均为有限个数，由此得出，在无穷推导中路径上，至少存在一个规则被用于分解两次，这时，规则左部表达的授权会出现直接或间接自授权的形式，由此得出矛盾。因此，委托的推导要么有限失败，要么成功。操作计算过程中，若子目标为正常目标，采用本节定义的操作语义时，文字选取策略是独立的[7]，即若操作推导结果为 $(c \mid \square)$，$c = c_1 \wedge \cdots \wedge c_k$，$c_i$ 为约束，则采用别的文字选取策略的结果仅仅是 c 结果的重排序；若推导子目标为否定子目标 $(c \mid \neg G)$，由于操作推导过程先推导(true| G)，这样的文字选取策略是独立的，经过否定的结果也是独立的。

3.3 委托模型可靠性、完备性和可行性

由于委托模型当推导子目标为正常子目标时，操作计算按 Jaffar[7] 的宽度优先操作计算，当子目标为否定子目标时，按 Stuckey 的 CLP-CN[11] 操作计算，由 Jaffar[7] 和 Stuckey[11] 的工作，容易得出：

定理 3.2 委托模型按照文字索引构成的宽度优先次序进行推导，并且遵循 3.2 节中的计算规则，则该模型的操作计算语义是可靠的和完备的。

可靠性和完备性是关注模型的计算操作是否正确和充分，可行性则关注模型的计算操作能否在多项式时间完成。从模型的操作计算过程中得知：计算是否有效可行取决于约束合取的可满足性能否有效判断，如对否定推导 Negation-Delegation 计算的可行性的关键在于 step4，即约束合取 $c_i \wedge c'_j$ 的可满足性判断，与正常子目标推导有效性相比，这里 c'_j 不是正常匹配展开时的约束，而是经过否定和析取后得来，可行性问题表示为[12]：对 $\prod_C$ 上的解释 D，寻找约束子集 AC (admissible constraints)，$D \models \widetilde{\exists} \varphi, \varphi \in$ AC，且是可行的(decidable)。对正常子目标推导 φ 是正常约束合取式，可行性可表示为：

定义 3.2 设 D 为 $\prod_C$ 上的解释，对每一个约束的合取公式 φ，都存在一个无量词的约束的析取公式 ψ，满足 $D \models \widetilde{\forall}(\widetilde{\exists} \varphi \leftrightarrow \psi)$，且 ψ 能有效计算，称 D 允许量词消除的。

如果是否定推导子目标，可行性为：

定义 3.3 设为 D 为 $\prod_C$ 上的解释，如果对每一个 D 上的约束合取公式 $\varphi[\widetilde{X}, \widetilde{Y}]$，存在一个约束公式 $\varphi[\widetilde{X}, \widetilde{Y}]$，满足

1. φ 是定义 2.1 约束的析取式，

2. $D \models \forall \widetilde{X}(\neg \exists \widetilde{Y} \varphi[\widetilde{X}, \widetilde{Y}] \leftrightarrow \exists \widetilde{Z} \psi[\widetilde{X}, \widetilde{Y}])$

如果 ψ 可满足性能有效判断，称 D 是可构造性允许闭的(constructively admissible closed)。

该定义对 Stuckey[11] 的定义进行了简化，Stuckey 的约束定义是约束域上一般的一阶公式，本文定义的约束按照 Li[6] 的定义，与 datalog^c 的约束定义相同。

在 RT_1^- 中，Li[6]给出了两个约束域，树约束域(tree domain)和范围约束域(range domain)，Li 证明这两个约束域是量词可消除的，并且消除量词后的可满足性判断具有相对于程序大小的多项式时间解。我们证明扩展的范围约束域同样满足允许闭性，并且程序的计算时间复杂度为 PTIME。

定义 3.4 一个扩展的范围约束域，是一个 Li[6]描述的范围约束域，其上增加一种原始约束 $x \neq y$。

定理 3.3 一个扩展的范围约束域满足量词可消除的。

证明：考虑 $\widetilde{\exists}\varphi_1 \wedge \cdots \varphi_n$，其中 φ_i 是原始约束，当原始约束为 $x \neq y$ 时，可取 $x=c, c$ 为常量，$y=c_1, c_1$ 为大于 c 的常量，将 $x \neq y$ 代以 $(x=c) \wedge (y=c_1)$ 这样原始约束的存在量词可消除，当原始约束为别的形式时，参考 Li[6]的证明，量词消除可成立。

定理 3.4 设 C 是扩展的范围约束域，D 是其上的解释，则 D 是允许闭的(admissible closed)。

证明：考虑原始约束的否定，由范围约束域的定义，原始约束只有如下几种：$x=y, x \neq y, x=c, x \in (c_1, c_2)$(括弧可以是方括弧)，可以验证每个原始约束的否定也是原始约束或原始约束的析取，如 $x \notin (c_1, c_2) \leftrightarrow x \in (*, c_1) \vee (c_2, *)$，由定理 3.3 和文献[11]可得扩展范围约束域上的结构 D 是允许闭的。

由此得知：模型的操作计算在扩展的范围约束域上是可行的。

结束语 本文通过将 Stuckey 的 CLP-CN 的语义引入 RT 授权模型中，给出了在分布协作环境中的一个支持否定授权的委托模型，给出了否定推理的操作计算语义，讨论了该模型的可靠性和完备性，并对适合该模型的约束域进行了简单讨论，给出了可行的约束域。进一步的工作是找到更多的约束域类型以适应应用环境的需求，这是我们今后的研究工作。

参 考 文 献

[1] Li Ninghui, Mitchell J C, Winsborough W H. Design of a role-based trust management framework // Proceedings of the 2002 IEEE Symposium on Security and Privacy. Oakland, CA, USA, 2002: 114-130

[2] Ye Chunxiao, et al. An attribute-based extended delegation model. Journal of Computer Research and Development, 2006, 43(6): 1050-1057

[3] Li Ninghui, et al. Delegation logic: a logic-based approach to distributed authorization. ACM Transactions on Information and System Security, 2003, 6(1): 128-171

[4] Trevor J. SD3: a trust management system with certificate evaluation // Proceedings of the 2001 IEEE Symposium on Security and Privacy. IEEE Computer Society Press, May 2001: 106-115

[5] Freudenthal E, et al. dRBAC: distributed role-based access control for dynamic coalition Environment // Proc. 22nd International Conference on Distributed Computing Systems (ICDCS'02). Vienna: IEEE, 2002: 294-306

[6] Li Ninghui, Mitchell J C. Datalog with constraints: A foundation for trust management languages // Proceedings of the Fifth International Symposium on Practical Aspects of Declarative Languages, (PADL 2003). Lecture Notes in Computer Science, vol. 2562, New York, Springer-Verlag: 58-73

[7] JaffarJ, Maher M, Marriott K, et al. The semantics of constraint logic programs. Journal of logic programming, 1994, 19(20): 1-679

[8] Lloyd J W. Foundations of logic programming. Springer-Verlag, 1987

[9] Chan D. Constructive Negation Based on the Completed Data-Base // Proceedings of 5th International Conference and Symposium on Logic Programming. 1988: 111-125

[10] Drabent W. What is Failure? An Approach to Constructive Negation. Acta Inf., 1995, 32(1): 27-29

[11] Stuckey J C. Negation and constraint logic programming. Information and Computation, 1995, 118(1): 12-33

[12] Dovier A, Pontelli E, Rossi G. A necessary condition for constructive negation in constraint logic programming. Information Processing Letters, 2000, 74(3/4): 147-156

(上接第 201 页)

$d(A \oplus H, B \oplus H) = d(H, H) + d(A \odot H^C, B \odot H^C) = d(A \odot H^C, B \odot H^C)$，

同理，即知

$d(A \oplus H^C, B \oplus H^C) = d(H^C, H^C) + d(A \odot H, B \odot H) = d(A \odot H, B \odot H)$，

因此

$d(A, B) = d(A \odot H, B \odot H) + d(A \odot H^C, B \odot H^C) = d(A \oplus H^C, B \oplus H^C) + d(A \oplus H, B \oplus H)$。

对于"⇒"类似可证。

易证：

命题 6.2 设 d 为 IIVFS(X)上的一个距离测度，对于任意的 $A, B, H \in$ IIVFS(X)，如果 $A \subseteq B \subseteq H$，则有

$d(A, B \oplus H) \geqslant d(B, A \oplus H)$。

命题 6.3 设 d 为 IIVFS(X)上的一个距离测度，对于任意的 $A, B, H \in$ IIVFS(X)，如果 $A \subseteq B$，则有

$d(A \odot H, B \oplus H) \geqslant d(A, B)$。

命题 6.4 设 d 为 IIVFS(X)上的一个距离测度，对于任意的 $A, B \in$ IIVFS(X)，则有

$d(A \odot B, A \oplus B) \geqslant d(A \cap B, A \cup B)$。

7 加法与乘法下的相似测度

相似测度与距离测度有着较为相似的性质，我们可以类似地证明下列命题：

命题 7.1 设 s 为 IIVFS(X)上的相似测度，则对于任意的 $A, B \in$ IIVFS(X)，任意的分明集 H，有

$s(A, B) = s(A \odot H, B \oplus H^C) + s(A \odot H^C, B \oplus H) \Leftrightarrow s(A, B) = s(A \odot H, B \oplus H^C) + s(A \oplus H, B \odot H^C)$。

命题 7.2 设 s 和 d 为 IIVFS(X)上相互诱导的相似测度和距离测度，则对于任意的 $A, B \in$ IIVFS(X)，任意的分明集 H，有

$s(A, B) = s(A \odot H, B \oplus H^C) + s(A \odot H^C, B \oplus H) \Leftrightarrow d(A, B) = d(A \oplus H, B \oplus H) + d(A \oplus H^C, B \oplus H^C)$。

命题 7.3 设 s 是 IIVFS(X)上的相似测度，且对于任意的 $A, B \in$ IIVFS(X)，任意的分明集 H，均有 $s(A, B) = s(A \odot H, B \oplus H^C) + s(A \odot H^C, B \oplus H)$，则由 s 诱导的熵为 γ-熵。

参 考 文 献

[1] Atanassov K, Gargov G. Interval-valued intuitionistic fuzzy sets [J]. Fuzzy Sets and Systems, 1989, 31: 343-349

[2] Atanassov K. Operations over interval-valued intuitionistic fuzzy sets[J]. Fuzzy Sets and Systems, 1994, 64: 159-174

[3] Mondal T, Samanta S. Topology of interval-valued intuitionistic fuzzy sets[J]. Fuzzy Sets and Systems, 2001, 119: 483-194

[4] 郭效芝. 模糊不确定性度量的探讨及扩展[D]. 西北大学硕士学位论文. 2004: 48-50

[5] Liu X. Entropy distance measures and similarity measure of fuzzy sets and their relations [J]. Fuzzy sets and systems, 1992, 72: 331-348