

使用本体和 SWRL 验证作战计划的方法

钱 猛, 刘 忠, 都业宏, 姚 莉

QIAN Meng, LIU Zhong, DU Ye-hong, YAO Li

国防科技大学 信息系统与管理学院, 长沙 410073

School of Information System and Management, National University of Defense Technology, Changsha 410073, China

E-mail: qianmeng@nudt.edu.cn

QIAN Meng, LIU Zhong, DU Ye-hong, et al. Course of Action validation approach using Ontology and SWRL. Computer Engineering and Applications, 2009, 45(8): 208-212.

Abstract: Course of Action (COA) validation is the fundamental step in military planning process. Based on the research of plan representation and plan reasoning, authors propose an ontology and rule based validation approach for COA. In this approach, authors construct the COA validation process with three sub-processes, which are ontological COA representation, description logic and first order logic based COA knowledge abstraction and COA reasoning, to realize the COA validation. A COA of an anti-landmine task is given as the validation example. The example shows that the approach can extend the knowledge of experts and improve the efficiency of correctness validation for COA.

Key words: ontology; rule; reasoning; Course of Action (COA) validation; Web Ontology Language (OWL); Semantic Web Rule Language (SWRL); First Order Logic (FOL)

摘 要: 计划验证是计划编制过程的重要环节。从知识管理的角度, 在已有计划表示和计划推理相关研究的基础上, 提出了一种基于本体和规则的计划验证方法框架, 通过对计划知识进行基于本体的表示、基于描述逻辑、一阶谓词逻辑的规则知识抽取和计划推理, 以实现对计划的验证。并以某排雷作战计划的验证为例进行了实验。实验表明, 该方法实现了对专家知识的扩展, 提高了作战计划正确性检验的效率。

关键词: 本体; 规则; 推理; 作战计划验证; Web 本体语言; 语义网规则语言; 一阶逻辑

DOI: 10.3778/j.issn.1002-8331.2009.08.063 **文章编号:** 1002-8331(2009)08-0208-05 **文献标识码:** A **中图分类号:** TP391.9

1 引言

作战计划 (Course of Action, COA) 编制过程涉及到不同的作战指挥机构、计划机构和不同的执行单位, 涉及到丰富的、分布的、不一致的信息和知识模型。如何集成这些模型, 并进行面向决策的智能化计划验证、推理和动态生成, 是作战决策过程的重要组成部分; 其中, 作战计划验证是特别基础而关键的环节。

国外对作战计划的研究开展较早, 计划表示的模型也较为成熟, 如 <I-N-OVA>^[1]、CPR^[2] 系列、KRSL-Plan^[3]、SPAR^[4]、JFACC^[5] 等。文献 [6] 表明上述研究对作战计划的表示趋于一致, 并给出上述计划编制模型对应的计划本体。其中, 乔治梅森大学的 DISCIPLE-COA 项目通过构造评估 agent 的方法, 利用任务层次进行类比推理, 以实现作战计划的验证^[7]。而南加州大学的 PLANET 项目则通过构建评估本体, 利用 PLANET-COA 评估工具对作战计划进行验证^[8]。

本文采用语义网技术, 提出一种基于本体和规则的计划验证方法。该方法以结合 OWL^[9] 描述的本体和 SWRL^[10] 以及 SWRL FOL^[11] 描述的规则进行推理的方式来实现计划的验证。由于本

方法遵循知识的“表示-抽取-推理”的过程, 故简称为 R²A (Representation-Abstraction-Reasoning) 方法。

2 R²A 计划验证方法框架

在 R²A 方法中, 计划的验证过程分为计划知识表示、计划知识抽取和计划知识推理三个部分。在计划知识表示阶段, 将计划按照计划概念、计划规则和计划实例以计划本体的形式进行表示。在计划本体的基础上, 进行计划知识的抽取, 分别得到计划的公理集、实例集、实例判断集以及规则集 (包含规则转换方法)。依据获得的计划知识, 进行计划知识的推理, 实现计划验证。

根据计划知识的推理的类型, 将计划验证划分成两个相关的部分, 分别为描述逻辑的验证和一阶谓词逻辑的验证。

描述逻辑验证是指验证概念之间、概念与实例之间以及实例之间的关系, 使其与建模人员的知识一致, 主要用于检验建模过程的正确性, 如一致性及完整性检验; 同时, 用户根据自己所关心的限制条件, 查询实际计划中的实例是否与限制条件相

基金项目: 国家自然科学基金 (the National Natural Science Foundation of China under Grant No.70271004, No.70771109)。

作者简介: 钱猛 (1968-), 男, 讲师, 博士研究生, 主要研究方向为知识管理和知识工程; 刘忠 (1968-), 男, 博士, 教授, 主要研究方向为计划生成与评估技术; 都业宏 (1986-), 男, 硕士生; 姚莉 (1965-), 女, 博士, 教授, 主要研究方向为分布式人工智能, 智能决策。

收稿日期: 2008-09-17

修回日期: 2008-12-08

违背,如计划中对某武器系统的数量做出限制,然后查询使用到该资源的任务个数,如果超出限制则为资源冲突,这就需要用户定义自己的推理规则,在语法层次的基础上验证语义层次的正确性。

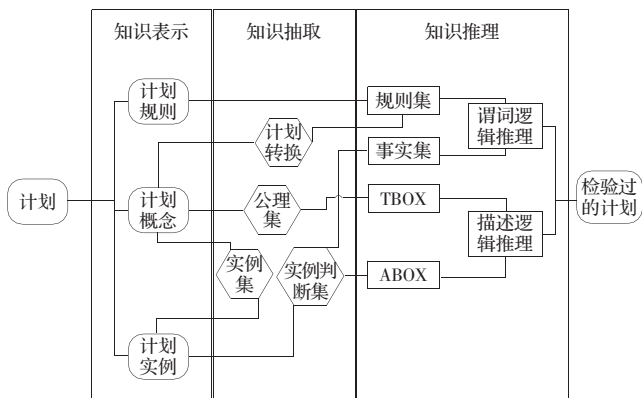


图1 R²A方法框架示意图

一阶谓词逻辑的验证是专家根据领域的知识定义的概念以及个体之间相互关系的规则。它无法纳入描述逻辑的框架之内,但采用标准的本体表述,通过对个体判断形成的事实库以及专家定义形成规则库的共同一阶谓词逻辑推理,从领域知识判断计划的合理性。

3 基于本体的作战计划表示

3.1 作战计划本体

作战计划本体描述作战计划的客观知识。这类客观知识的描述是指对已经存在的、正在进行的和未来可获取的作战过程/作战系统/作战装备/作战人员以及它们之间的相互关系(依赖)的描述。根据作战计划的特点,作战计划的内容可以归结为目标、对象、过程、消息、依赖、时间描述、空间描述和计划这几个问题域(类的集合)进行分析。基于本体的作战计划概念模型(Concept Model of COA based on Ontology,CMCO)可以定义为一个八元组:

$$CMCO = \langle Objective, Object, Process, Message, Dependency, TimeSpec, SpaceSpec, Plan \rangle$$

其中,Objective 对应目标域, Object 对应对象域, Process 对应过程域, Message 对应消息域, Dependency 对应依赖域, TimeSpec 和 SpaceSpec 对应时空域, Plan 对应计划域。如图2所示。

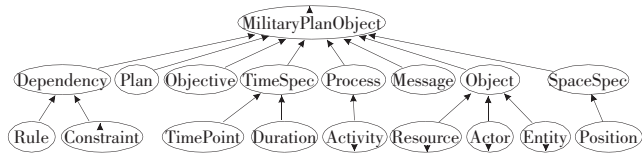


图2 作战计划本体

目标域:描述作战计划的任务内容以及拟达到的作战效果。

对象域:描述作战的参与者、作战实体和作战时涉及的资源。表示为一个三元组:

$$Object = \langle Actor, Entity, Resource \rangle$$

其中, Actor 表示作战的参与者,如(敌我双方)作战人员; Entity 表示作战实体,如作战装备、武器系统、指挥信息系统等; Resource 表示实施作战所涉及的资源,如油料、弹药等。如图3所示。此处特举工兵、探雷器和地雷的例子来表示对象域。

过程域:描述作战过程的结构,一个作战过程包括一个或

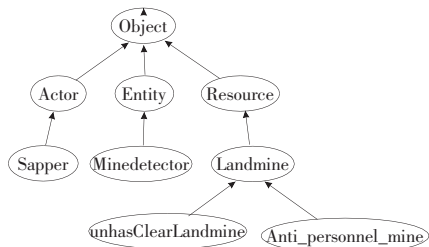


图3 对象域

多个作战活动,每个作战活动由行动(原子活动)组成。

消息域:描述作战过程之间的通信。如命令、情报简报等。

依赖域:描述作战过程之间、作战对象之间、作战过程/对象之间的相互关系,表示为约束与规则的二元组:

$$Dependency = \langle Constraint, Rule \rangle$$

时间域:描述作战过程的时间点与时间段,表示为一个二元组:

$$TimeSpec = \langle TimePoint, Duration \rangle$$

空间域:描述作战过程发生的空间范围。

3.2 作战计划本体中概念的属性集

对于作战计划的概念,可通过定义属性进一步地刻画。同时也通过属性来反应概念间的关系。如对象域的工兵类、探雷器类和地雷类,可以定义属性如表1、表2和表3。

表1 工兵类的属性

属性 ID	属性名	注释
1	hasKnow	了解(地雷)
2	unhasKnow	不了解(地雷)
3	hasClearmine	清除地雷
4	hasClearmineVelocity	扫雷速度
5	hasClearmineMediumVelocity	中等扫雷速度
6	hasClearmineHighrateVelocity	高效扫雷速度
7	hasClearmineQuickVelocity	快速扫雷速度
8	hasClearmineSlowVelocity	一般扫雷速度

表2 探雷器类的属性

属性 ID	属性名	注释
1	hasDetect	探测(地雷)
2	hasPrompt	通知(工兵)

表3 地雷类的属性

属性 ID	属性名	注释
1	hasTimeRestrictions	时间限制
2	unhasKnownBy	不被(工兵)了解
3	hasKnownBy	被(工兵)了解
4	hasMinePosition	位置信息
5	hasLongitude	经度
6	hasLatitude	纬度
7	hasHeight	高程

将作战计划任务中的实际事物映射为对应概念(类)的个体,并给这些概念个体赋值,使得 ABox 具有具体知识。如表4、表5、表6中,对工兵1、探雷器1和反步兵地雷1的个体属性分别进行赋值。

4 基于本体和规则的作战计划验证

4.1 计划的描述逻辑验证

描述逻辑验证知识库主要由 TBox 和 ABox 两部分组成。

表4 工兵个体属性值列表

		Anti_personnel_mine_3\
		Anti_personnel_mine_8\
		Anti_personnel_mine_6\
		Anti_personnel_mine_2\
Sapper_1	hasKnow	Anti_personnel_mine_5\ Anti_personnel_mine_10\ Anti_personnel_mine_1\ Anti_personnel_mine_7\ Anti_personnel_mine_4\ Anti_personnel_mine_9
	unhasKnow	Anti_personnel_mine_11

表5 探雷器个体属性值表

	hasDetect	Anti_personnel_mine_1\ Anti_personnel_mine_2\ Anti_personnel_mine_3\ Anti_personnel_mine_4
Mine_detector_1		Sapper_1\Sapper_2\ Sapper_3\Sapper_4
	hasPrompt	Sapper_3\Sapper_4

表6 地雷个体属性值列表

	hasTimeRestrictions	30
	unhasKnownBy	
	hasKnownBy	Sapper_1\Sapper_2\ \Sapper_3\Sapper_4
Anti_personnel_mine_1		E101°18'19"\ N23°18'19"\ 45
	hasLongitude	E101°18'19"
	hasLatitude	N23°18'19"
	hasHeight	45

TBox 定义了特定知识领域的结构并包含一系列公理,可以通过已有概念构成新的概念。ABox 包含了 TBox 中概念的实例,即个体集。将 TBox 和 ABox 相分离,保证了知识结构不被破坏,将类和实体分开,使 TBox 具有复用性。图 4 是 TBox 相关的公理集所包含的公理的示例:

```

探雷器是一种作战实体
MineDetector ⊆ Entity
每名工兵每小时最多排雷 5 颗;
Sapper ⊆ (≤5)hasClearMine
反步兵地雷可定时 30 分钟内爆炸;
Anti_personnel_mine ⊆ (≤30)hasTimeRestrictions
.....

```

图4 TBox 示例

ABox 是实例断言的集合,用于指明个体的属性或个体之间的关系。ABox 是将与 TBox 中的类相对应的实例填入,所建立的实例要符合 TBox 中类设计的限制和属性,这些类的实体(individual)称为实例(instance),由这些实例可以将原来只具有概念的架构,组合为可以表现实体知识关系的架构。ABox 相关的一些断言示例如图 5 所示。

图 5 中,Landmine 是还需要排除的地雷,Minedetector 为探雷器,Sapper 为实施排雷的工兵。目前的任务是 4 名工兵将使用 4 只排雷器对 20 枚地雷进行排除作业。hasPrompt 是属性断言,表示探雷器在探测到地雷时发出提示信号。hasTimeRe-

```

Landmine(?x)
Minedetector(?y)
Sapper(?s)
hasPrompt(?y,Sapper_2)
hasKnow(Sapper_2,?x)
hasTimeRestrictions(?x,20)
unhasClearmine(Sapper_1,?x)
hasClearmine(Sapper_2,?x)
unhasClearmine(Sapper_3,?x)
unhasClearmine(Sapper_4,?x)
.....

```

图5 ABox 示例

strictions 是地雷的属性断言,表示该种地雷将在 20 min 以后爆炸,等等。

将计划的一致性检验归结为描述逻辑的一致性检测(或称为可满足性检测),可以对计划进行逻辑检验。一致性检验与描述逻辑间的映射关系如图 6 所示^[12]:

```

实例个数限制的一致性检验
¬ ∃y(ancestor(y,x) ∧ (x.minCardinality < y.minCardinality)
∧ (x.maxCardinality > y.maxCardinality))
属性定义域/值域的一致性检验
¬ ∃y(ancestor(y,x) ∧ (ancestor(x.domain,y.domain)
∧ (ancestor(x.range,y.range))))
避免类和属性的循环定义
¬ ∃x1x2...xn(superClass(x1,x2) ∧ superClass(x2,x3) ∧ ... ∧ superClass(xn,x1))
类之间关系和属性之间关系的一致性检验
¬ ∃x,y(complementClass(x,y) ∧ (superClass(x,y) ∧ subClass(x,y) ∧ equivalent-
Class(x,y)))

```

图6 一致性检验与描述逻辑间的映射关系

通过对计划进行描述逻辑推理,可以得出计划的语法层次的验证,可以借助支持描述逻辑的推理系统来实现。国内外现在有很多成熟的系统如 Jena、Racer 和 Pallet 等,都能很好地支持 OWL 语法的推理验证。本文采用 Racer 对建立的计划模型进行基于描述逻辑的验证。

4.2 计划的一阶逻辑验证

一阶逻辑检验与应用结合,根据预算的需要给相应的限制条件,在检验方案时,查询与用户自定义的限制条件,检查是否存在冲突。对采用描述逻辑语言进行自定义规则,这样一方面自定义规则具备强大的逻辑基础,有利于规则之上的隐含推理,另外一方面也有利于精炼自定义规则成领域规则,增加领域推理机规则。

基于规则的 FOL 推理系统中存在两种不同形式的谓词公式,即事实(Fact)和规则(Rule)。事实是不含蕴涵符号的谓词公式,表示与求解问题有关的客观情况、证据等。规则就是包含前提条件和结论的蕴涵式。例如图 7:

```

Minedetector(?x) ∧
Landmine(?y) ∧
Sapper(?z) ∧
hasDetect(?x,?y) ∧
hasKnow(?z,?y)
→hasPrompt(?x,?z)

```

图7 一阶逻辑表示的规则

通过将 OWL 描述的本体和 SWRL/SWRL FOL 表示的规则统一转换为一阶谓词逻辑公式,利用现有的一些功能强大的

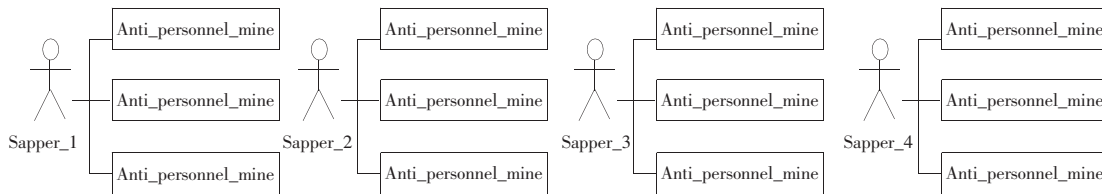


图 8 一种粗略的排雷计划

FOL 推理机来实现推理。SWRL/SWRL FOL 表示的规则转换是自然的,OWL 本体向 FOL 映射方法主要如下:

- 将 OWL 本体中的类映射为 FOL 中的一目谓词;
- 将 OWL 本体中的属性映射为 FOL 中的二目谓词;
- 将 OWL 中的实例映射为 FOL 中的实例常量;
- 如果 OWL 本体中 a 是类 C 的一个实例,映射为: $C(a)$, a 为实例常量;

如果 OWL 本体中 a 的 P 属性的值为 b ,映射为: $P(a,b)$, a , b 为实例常量;

OWL 中的公理(Axioms)映射为 FOL 中的蕴涵式。具体的映射方法见表 7:

表 7 OWL 本体向 FOL 映射方法

OWL 语法	描述逻辑语法	一阶谓词表达式	解释
C	C	$C(x)$	类
P	P	$P(x,y)$	属性
subClassOf	$C \sqsubseteq D$	$D \supset C$	C 是 D 的子类
subPropertyOf	$P \sqsubseteq Q$	$\forall x,y P(x,y) \rightarrow Q(x,y)$	P 是 Q 的子属性
transitiveProperty	$P \sqsubseteq P$	$\forall x,y,z (P(x,y) \wedge P(y,z)) \rightarrow P(x,z)$	P 具有传递性
inverseOf	$P \equiv Q$	$\forall x,y P(x,y) \leftrightarrow Q(y,x)$	P, Q 互逆
intersectionOf	$C_1 \sqcap \dots \sqcap C_n$	$C_1(x) \wedge \dots \wedge C_n(x)$	交
unionOf	$C_1 \sqcup \dots \sqcup C_n$	$C_1(x) \vee \dots \vee C_n(x)$	并
complementOf	$\neg C$	$\neg C(x)$	补
oneOf	$\{a_1, \dots, a_n\}$	$x \neq a_1 \vee \dots \vee x \neq a_n$	枚举
hasClass	$\exists PC$	$\exists y (P(x,y) \wedge C(y))$	存在 P 的值域为 C
toClass	$\forall PC$	$\forall y (P(x,y) \rightarrow C(y))$	所有 P 的值域为 C

OWL 本体和 SWRL/SWRL FOL 规则都建立好之后,通过映射为一阶谓词逻辑公式,就可以利用 FOL 推理引擎进行推理了。

对于一阶逻辑的验证,本文采用了 Protege 中 SWRL 编辑器与 JESS 引擎结合的方式进行。

5 实例

现有一项排雷任务需要制订和检验作战计划。该任务共派遣工兵 4 名,每人配发探雷器一件,雷区共分布约 12 枚地雷,其中有定时雷和未知性能的地雷。人工制订计划时,一般地,每名工兵人均排雷 3 枚,因此存在这样一种可能的、粗略的排雷计划,如图 8 所示。该计划对排雷任务的目标考虑较多,对排雷过程及其各种相关因素考虑较少,实施时往往是遇到具体问题再考虑对策。

根据专家的领域知识,当工兵在正确使用探雷器的前提下,如果了解所排除地雷的工作原理及特性,并在一定的时间约束下(如地雷定时爆炸的时间),能够独立排除地雷。根据本文的方法,把专家的领域知识转化为相应的若干条基于逻辑的规则,使得每一个个体而言,都应该适用这些规则。

例如:规则 1 表示为:如果 x 为地雷, y 为探测器, y 通知 Sapper_1(x 的有关信息),Sapper_1 了解 x 的参数等信息, x 的时限为 30 min,那么 Sapper_1 负责拆除 x ,Sapper_2,Sapper_3,Sap-

per_4 不参与。

规则 1 用描述逻辑描述如图 9。

```

Landmine(x) ∧ Minedetector(y) ∧ hasPrompt(y, Sapper_1) ∧ hasKnow(Sapper_1, x) ∧
hasTimeRestrictions(x, 30)
=> hasClearmine(Sapper_1, x) ∧ unhasClearmine(Sapper_2, x) ∧ unhasClearmine
(Sapper_3, x) ∧ unhasClearmine(Sapper_4, x)
    
```

图 9 规则 1 的描述逻辑表示

规则 1 用 SWRL 表示如图 10。

```

Landmine(?x) ∧
Minedetector(?y) ∧
hasPrompt(?y, Sapper_1) ∧
hasKnow(Sapper_1, ?x) ∧
hasTimeRestrictions(?x, 30)
→ hasClearmine(Sapper_1, ?x) ∧
unhasClearmine(Sapper_2, ?x) ∧
unhasClearmine(Sapper_3, ?x) ∧
unhasClearmine(Sapper_4, ?x)
    
```

图 10 规则 1 的 SWRL 表示

排雷规则的表示和排雷计划的推理过程如图 11 所示:

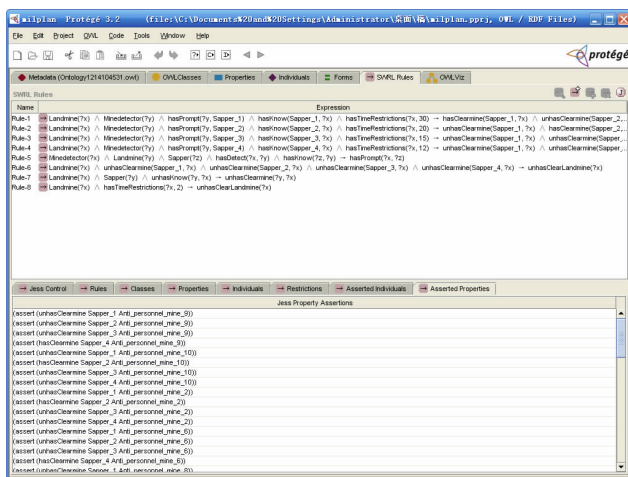


图 11 JESS 推理得到的属性断言结果

从推理得到的扫雷行动计划中可以发现这样的规律:每个地雷无论其是否被清除,对应该本体中的 4 个工兵都具有 4 个断言和 2 种状态(被清除和不被清除)。其中, Anti_personnel_mine_11 的参数 4 个工兵都不了解其参数,所以没有工兵去拆除它,进而可以判断 Anti_personnel_mine_11 是未清除的地雷。 Anti_personnel_mine_12 的时限为 2 min,超出了所有的工兵能力之所及,所以亦是未清除的地雷。 Anti_personnel_mine_1 到 Anti_personnel_mine_10 共有 10 颗地雷符合被清除的各项条件,对应到每个地雷有一个相同的特点:每个地雷只被 1 个工兵清除掉,其他工兵不参与,地雷被清除这一事件具有唯一性。由于此本体中的探测器和工兵是理想化的,可能多个探测器均

探测到了某一地域存在地雷,多个工兵均了解某一地雷的参数,所以生成的计划中存在多个探测器通知某一工兵的情况。

依据本文的方法,推理得到的工兵个体的属性的断言用JESS的断言格式表示为表8~表10。依据这些断言,可以生成较为周密的排雷计划,包括三个方面:清除任务、不清除任务和不能清除(地雷),分别如图12~图14。

表8 清除地雷的行动的断言

```

(assert(hasClearmine Sapper_4 Anti_personnel_mine_6))
(assert(hasClearmine Sapper_4 Anti_personnel_mine_8))
(assert(hasClearmine Sapper_3 Anti_personnel_mine_4))
(assert(hasClearmine Sapper_4 Anti_personnel_mine_7))
(assert(hasClearmine Sapper_2 Anti_personnel_mine_10))
(assert(hasClearmine Sapper_2 Anti_personnel_mine_2))
(assert(hasClearmine Sapper_3 Anti_personnel_mine_5))
(assert(hasClearmine Sapper_4 Anti_personnel_mine_9))
(assert(hasClearmine Sapper_1 Anti_personnel_mine_1))
(assert(hasClearmine Sapper_3 Anti_personnel_mine_3))

```

可以发现,推理生成的计划对比图9有较大的差异,主要是考虑了排雷行动过程中的人员、实体和资源的不同的约束。同时,还可以规范作战人员的职责范围,如图10所示。并将约束条件下无法实现的任务进行了预计,如图11。即Anti_personnel_mine_11和Anti_personnel_mine_12的杀伤范围的空间内存在威胁,排雷部队机动时应该与其保持一定的距离。

从以上实例可以看出,利用描述逻辑和一节逻辑可以对本体建立的知识进行推理,推理的结果可以对人工制订的计划进行验证、修订和增补,并自动生成计划内容。为了说明问题,此处的实例仅考虑了比较简单的约束条件,在实际情况下,如果加入更多的战场反馈信息来修改约束和规则,还可以对计划推理结果进行实时调整,使之更加符合完成作战任务的要求。

6 结束语

计划验证是作战决策过程基础而关键的环节。通过计划验证,

表9 不进行清除地雷的行动的断言

```

(assert(unhasClearmine Sapper_1 Anti_personnel_mine_6))
(assert(unhasClearmine Sapper_2 Anti_personnel_mine_6))
(assert(unhasClearmine Sapper_3 Anti_personnel_mine_6))
(assert(unhasClearmine Sapper_1 Anti_personnel_mine_8))
(assert(unhasClearmine Sapper_2 Anti_personnel_mine_8))
(assert(unhasClearmine Sapper_3 Anti_personnel_mine_8))
(assert(unhasClearmine Sapper_1 Anti_personnel_mine_4))
(assert(unhasClearmine Sapper_2 Anti_personnel_mine_4))
(assert(unhasClearmine Sapper_4 Anti_personnel_mine_4))
(assert(unhasClearmine Sapper_1 Anti_personnel_mine_7))
(assert(unhasClearmine Sapper_2 Anti_personnel_mine_7))
(assert(unhasClearmine Sapper_3 Anti_personnel_mine_7))
(assert(unhasClearmine Sapper_1 Anti_personnel_mine_10))
(assert(unhasClearmine Sapper_3 Anti_personnel_mine_10))
(assert(unhasClearmine Sapper_4 Anti_personnel_mine_10))
(assert(unhasClearmine Sapper_1 Anti_personnel_mine_2))
(assert(unhasClearmine Sapper_3 Anti_personnel_mine_2))
(assert(unhasClearmine Sapper_4 Anti_personnel_mine_2))
(assert(unhasClearmine Sapper_1 Anti_personnel_mine_5))
(assert(unhasClearmine Sapper_2 Anti_personnel_mine_5))
(assert(unhasClearmine Sapper_4 Anti_personnel_mine_5))
(assert(unhasClearmine Sapper_1 Anti_personnel_mine_9))
(assert(unhasClearmine Sapper_2 Anti_personnel_mine_9))
(assert(unhasClearmine Sapper_3 Anti_personnel_mine_9))
(assert(unhasClearmine Sapper_2 Anti_personnel_mine_1))
(assert(unhasClearmine Sapper_3 Anti_personnel_mine_1))
(assert(unhasClearmine Sapper_4 Anti_personnel_mine_1))
(assert(unhasClearmine Sapper_1 Anti_personnel_mine_3))
(assert(unhasClearmine Sapper_2 Anti_personnel_mine_3))
(assert(unhasClearmine Sapper_4 Anti_personnel_mine_3))

```

表10 未清除地雷的断言

```

(assert(unhasClearLandmine(name Anti_personnel_mine_11)))
(assert(unhasClearLandmine(name Anti_personnel_mine_12)))

```

可以发现计划知识的不一致,确保计划编制的正确;并扩充

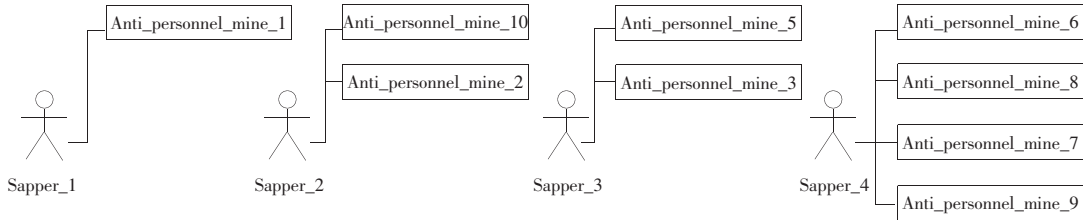


图12 推理出的排雷计划图示1(应排雷的分工)

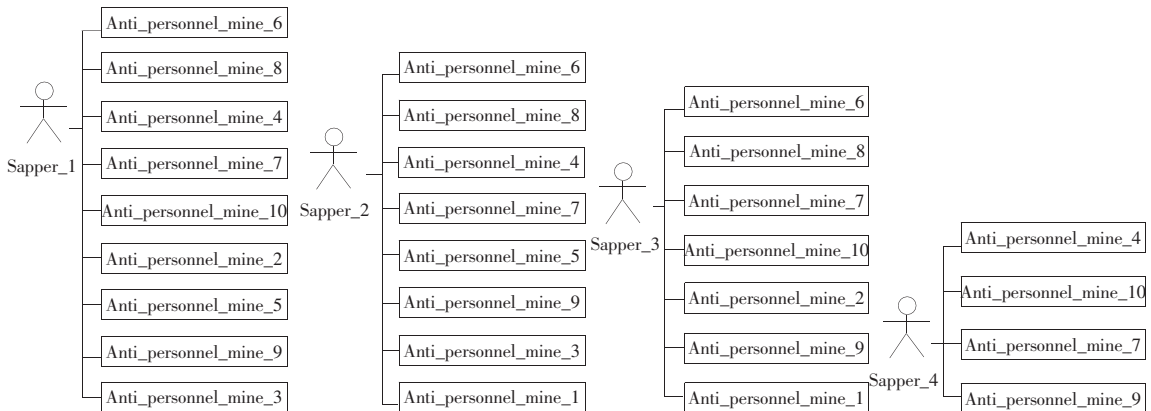


图13 推理出的排雷计划图示2(各工兵不须清除的地雷)