ICS 03.220.20
CCS R04

团 体 标 准

T/ITS 0293.2-2025

自主式交通系统 交通语义表示语言 第 2 部分：语法规范

Autonomous transportation system —

Traffic semantic representation language — Part 2:Syntax specification

2025-11-26 发布 2025-11-26 实施

中 国 智 能 交 通 产 业 联 盟 发布

目　　次

前　言

本文件按照GB/T 1.1—2020《标准化工作导则　第1部分：标准化文件的结构和起草规则》的规定起草。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由中国智能交通产业联盟（C-ITS）提出并归口。

本文件主要起草单位：北京交通大学、北京邮电大学、交通运输部公路科学研究院、华路易云科技有限公司、联通智网科技股份有限公司、北京航空航天大学、株洲中车时代电气股份有限公司、交通运输部水运科学研究院。

本文件主要起草人：董宏辉、牛亚杰、王佳佳、袁泉、李静林、任毅龙、于海洋、李振华、佘红艳、辛亮、于朝阳、王泉东、周昱诚、谌仪、欧帆、潘小熙、李巍、吴昊、刘俊兰、倪佳颖、冯佳瑞、江培源、洪奕鹏、周伟杰、兰征兴、顾惠楠、马攀科、林军。

自主式交通系统 交通语义表示语言 第 2 部分：语法规范

1 范围

本文件规定了自主式交通系统中的交通知识语义表示相关语法规范，并支撑多交通主体之间的语义交互。

本文件适用于自主式交通系统中的交通知识表达，包括但不限于交通规则、交通标志、交通信号、车辆行为等。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 1.1—2020 标准化工作导则 第1部分：标准化文件的结构和起草规则

ISO/IEC 13211-1—1995 信息技术 编程语言 Prolog 第1部分：通用核心

ISO/IEC 14882—2020 信息技术 编程语言 C++

T/ITS 0292-2025 自主式交通系统 互操作机制模型

T/ITS 0293.1-2025 自主式交通系统 交通语义表示语言 第1部分：通用定义

3 术语和定义

T/ITS 0292-2025、T/ITS 0293.1-2025界定的以及下列术语和定义适用于本文件。

3.1

自主式交通系统 autonomous transportation system

以自主感知、自主决策、自主执行为特征的高度智能、高度自治的交通系统。

[来源：T/ITS 0292-2025]

3.2

交通语义表示语言 traffic semantic representation language

交通语义表示语言是一种以形式化方式准确描述交通内容的语言，具备语义表示、语义理解、语义交互、逻辑推理和互操作等能力。

[来源：T/ITS 0293.1-2025]

3.3

自主式交通主体 autonomous traffic agent

自主式交通主体是能够在复杂交通环境下，独立完成感知、认知、决策与控制闭环，实现预定交通任务的交通智能体。

[来源：T/ITS 0293.1-2025]

3.4

事实 fact

描述交通实体属性的基本语句， 由“fact”关键字定义。

3.5

关系 relation

描述交通实体间关联的语句，由“relation”关键字定义。

3.6

规则 rule

描述交通逻辑约束的条件语句， 由“rule”关键字定义。

4 缩略语

TSRL：交通语义表示语言（traffic semantic representation language）

5 交通语义表示语言词法规范

5.1 字符集和编码

在本交通语义表示语言中采用Unicode字符集，支持中英文交通术语，源文件编码格式为UTF-8。

5.2 标识符

标识符是 TSRL 中用于命名核心语言元素的符号载体，其命名规则直接影响语义描述的可读性、一致性及多主体交互时的理解效率。为确保不同开发者以及不同系统对同一语言元素的标识统一，以下明

确标识符的组成规则与命名规范：

——TSRL中标识符用于命名模块、类别、类、属性、谓词等， 由字母、数字、下划线组成；

——模块名、类名、实体名，首字母大写，如TrafficLib 、Vehicle；

——属性名、谓词名采用大写字母开头，如HasSpeed 、Distance；

——常量由大写字母开头，可以包含数字和下划线，如Vehicle_1 、Lane5；

——变量、函数由小写字母或下划线开头，若仅有下划线符号，则为匿名变量，如_ 、speed(Car1)。

5.3 关键字

关键字是TSRL中用于构建核心语法结构的基础符号，通过特定标识界定术语组织形式、实体属性、关联关系及操作逻辑，是构建规范化交通语义描述体系的核心元素。具体关键字及说明见表1。

表 1 关键字

关键字	说明
module	定义术语库模块
category	定义交通系统类别
class	定义交通实体类
fact	定义实体属性事实
relation	定义实体间关系
action	定义实体动作
rule	定义逻辑规则
let	定义交通实体控制操作
select	定义知识库查询操作
asserta	定义知识库前插入操作
assertz	定义知识库后插入操作
retract	定义知识库删除操作
ask	信息查询/询问
tell	传递/告知信息
print	定义打印语句
fun	定义自定义函数
false	表示逻辑假值
true	表示逻辑真值

nil	表示空值/无实体
and	逻辑与运算符
or	逻辑或运算符
not	逻辑非运算符
forall	全称量词∀
exists	存在量词 ∃

5.4 基本数据类型

5.4.1 数值常量

数值常量是 TSRL 中用于量化描述交通实体属性的基础数据形式，其类型及定义如下：

——整型：int型整数为32位有符号数，如：1 、20 、333。

——浮点型：float为小数数值对象的表示方法，32位单精度浮点数。

5.4.2 布尔值

实际值为布尔值的状态判断符类型，true为真值、false为假值。

5.4.3 空值

在TSRL语言中，使用nil表示空值。

5.4.4 字符串

由双引号包裹的文本（如"十字路口"、"green"），支持转义字符\n（换行）、\"（双引号）。

5.5 复合数据类型

5.5.1 表

表是 TSRL 语言中用于组织交通实体集合的可变容器型实体类型，支持动态管理多类型交通语义元素，适配交通场景中动态变化的实体组需求，例如某车道中的车辆队列、事件序列。表实体化的一般形式为：

$$l = [x_1, x_2, \ldots x_n];$$

——符号说明：[ ]为表的界定符；x_i(i= 1,2, ⋯,n)为表中的项，代表交通实体或语义元素；

——特殊形式：不含任何元素的表称为空表，记为[] ，空表可用于初始化待动态填充的交通实体集合。

5.5.2 字典

字典是 TSRL 语言中用于建立交通语义键值映射的可变容器型实体类型，通过唯一键关联交通实体或属性值，适配交通场景中“标识-语义”映射需求（如信号灯相位-时长映射、车辆 ID-状态映射）。字典实体化的一般形式为：

$$d = \{key_1 : value_1, key_2 : value_2, \ldots key_n : value_n\};$$

——符号说明：{}为字典的界定符；key_n: $value_n$为键值对，“:”是键与值的分隔符，键值对之间用“,”分隔；

——键的约束：键用于唯一标识交通语义项，值为关联的实体或属性。

5.6 运算符与分隔符

运算符与分隔符是 TSRL 实现交通数据计算、逻辑判断及语句结构界定的核心语法元素。运算符支撑数值运算、逻辑推理等功能性表达；分隔符明确语句组成单元的边界（见表 6），二者共同保障语义描述的严谨性与可解析性。其中，运算符包括：

——比较运算符：用于判断数值间大小或相等关系的符号（见表2）；

——算术运算符：用于执行数值加法、减法、乘法等运算的符号（见表3）；

——量词运算符：用于表达“所有”或“存在”逻辑概念的标识（见表4）；

——逻辑运算符：用于实现逻辑判断，支撑交通语义逻辑推理的符号或关键字组合（见表 5）。

表 2 比较运算符

符号	说明
>	判断左侧数值是否大于右侧数值
<	判断左侧数值是否小于右侧数值
>=	判断左侧数值是否大于等于右侧数值
<=	判断左侧数值是否小于等于右侧数值
==	判断两侧值是否相等（支持数值、字符串、布尔值）
!=	判断两侧值是否不相等

表 3 算术运算符

符号	说明
+	数值加法运算/字符串拼接
-	数值减法运算/表示负数值
*	数值乘法运算
/	浮点除法运算，结果保留小数
**	幂运算
//	整数除法运算，结果取整数部分
%	取余运算

表 4 量词运算符

等效逻辑标识	说明	格式	示例
forall	对应逻辑符号∀，表示全称量词	∀x P(x)	forall xP(x)
exists	对应逻辑符号 ∃，表示存在量词	∃x P(x)	exists xP(x)

表 5 逻辑运算符

等效逻辑标识	说明	优先级	示例
not	对应逻辑符号¬， 表示否定	1	not P(x)
and / &	对应逻辑符号∧， 表示合取	2	P(x) and Q(x) P(x) & Q(x)
or / |	对应逻辑符号∨， 表示析取	3	P(x) or Q(x) P(x) | Q(x)
=>	对应逻辑符号→， 表示蕴含	4	Q(x) => P(x)
<=>	对应逻辑符号↔， 表示等价	5	P(x) <=> Q(x)

表 6 分隔符

符号	说明
( )	包裹函数及谓词的参数列表/改变表达式运算优先级
[ ]	包裹表的元素列表/表的索引访问
{ }	包裹模块、类别、类的代码块/包裹字典的键值对
;	语句结束符
,	分隔参数/分隔元素/分隔键值对
:	分隔字典中的键和值

5.7 注释

TSRL语言中的注释由“#”开头，至物理行尾结束。示例：#定义车辆类属性。

6 交通语义表示语言句法规范

6.1 TSRL 句法结构

6.1.1 事实

事实的句法结构，用于描述交通系统中的具体信息和状态，规定采用谓词和常量等来表示这些事实，事实语句的句法结构有以下两种：

a) 多条语句结构

<fact>

<谓词名> (<项表>);

<谓词名> (<项表>);

.....

</fact>

b) 单条语句结构

fact <谓词名> (<项表>);

——谓词分为实体、类别、属性、关系。

——实体：Vehicle(ID)。例：Vehicle(x)表示ID为x的车辆实体。

——类别：Is谓词(参数1，参数2，...，参数n)。例：IsVehicle(x)，表示x是车辆。

——属性：Has谓词(参数1，参数2，...，参数n)。例：HasSpeed(Vehicle，Speed)表示车辆Vehicle的速度为Speed。

——关系：谓词(参数1，参数2，...，参数n)。例：On(Vehicle，Road)表示车辆在某条道路上。

6.1.2 规则

规则的句法结构，用于描述交通系统中的约束条件和行为规范，包含条件和结论，使用逻辑运算符和量词来表示，规则语句的句法结构有以下两种：

a) 多条语句结构

<rule>

<谓词名>(<项表>) :- <谓词名>(<项表>){,<谓词名>(<项表>)};

<谓词名>(<项表>) :- <谓词名>(<项表>){,<谓词名>(<项表>)};

.....

</rule>

b) 单条语句结构

rule <谓词名>(<项表>) :- <谓词名>(<项表>){,<谓词名>(<项表>)};

——“:-”号表示“if”（也可以直接写为if），其左部的谓词是规则的结论（亦称为头），右部的谓词是规则的前提（亦称为体）。

——“ {} ”表示零次或多次重复，逗号表示and（逻辑与）。

6.1.3 询问

询问的句法结构，用于获取系统中的信息，规定通过谓词和变量来表示查询条件和结果，提问语句的句法结构有以下两种：

a) 多条语句结构

<ask>

<谓词名>(<项表>){,<谓词名>(<项表>)};

<谓词名>(<项表>){,<谓词名>(<项表>)};

.....

</ask>

b) 单条语句结构

ask <谓词名>(<项表>){,<谓词名>(<项表>)};

6.1.4 告知

告知的句法结构，用于传递系统中告知的信息，规定通过谓词和变量来表示告知的条件和结果，告知语句的句法结构有以下两种：

a) 多条语句结构

<tell>

<谓词名>(<项表>){,<谓词名>(<项表>)};

<谓词名>(<项表>){,<谓词名>(<项表>)};

.....

</tell>

b) 单条语句结构

tell <谓词名>(<项表>){,<谓词名>(<项表>)};

6.1.5 操作

操作的句法结构，用于描述交通系统中实体的具体行为，规定将其分为控制、知识库操作、赋值操作、打印操作。

a) 控制：表示对某个对象执行的具体动作指令，通过 let 关键字，可以触发某个行为，形式如下：

let <谓词名>(<项表>){,<谓词名>(<项表>)};

b) 知识库操作：对知识库即内存中的动态数据结构进行操作，由事实组成，规定 4 个动态数据库操作谓词，形式如下：

select(<谓词名>);

asserta(<fact>);

assertz(<fact>);

retract(<fact>);

——<fact>表示对应事实的某条谓词表达式。

——select(<谓词名>)查询语句，程序返回动态知识库中包含对应谓词名的谓词表达式。

——asserta(<fact>)把fact插入当前动态数据库中的同名谓词的事实之前。

——assertz(<fact>)把fact插入当前动态数据库中的同名谓词的事实之后。

——retract(<fact>)把fact从当前动态数据库中删除。

c) 赋值操作：将某个表达式的值赋予另一个对象，同时返回被赋予的值。形式有以下两种：

<项> = <项>;

<项> is <项>;

——使用“=”进行赋值时，如果实体属于基本实体类型，会新建一个实体进行引用。

——使用赋值号is，则会使得两个实体指向同一地址。

d) 打印操作：print 语句用于打印表达式的值。

print Expression;

——如果表达式Expression为计算表达式且可计算，直接打印计算结果。

——如果表达式Expression为谓词表达式，直接打印出该谓词表达式，而不会验证其真值。

6.2 类

a) TSRL 中类的概念及特性

——类是对物理世界某一集体概念的抽象。

——类具有某些属性，这些属性实体是某个类型的实例化。

——类存在某些关系，表现为谓词/函词表达式。创建类时会自带一些谓词/函词表达式，用于实体类型检验、属性取值、属性检验等。用户也可自行构建类的谓词/函词表达式。

——类包含某些规则，这些规则使用规则语句表示。

——类的继承，继承关系满足谓词/函词表达式Subset(subclass, superclass)，同时派生类也有用于实体类型检验、属性取值、属性检验的谓词/函词表达式。

b) TSRL 中类的构造

构造类的基本表示形式为：

class classname(superclass) {

ClasstypeA propertyA;

PredicateNameA($entity_{p,1}$, $entity_{p,2}$, …);

FunctionNameA($entity_{f,1}$, $entity_{f,2}$, …) = $entity_{f,x}$;

RuleNameA($entity_{r,1}$, $entity_{r,2}$, …) :- PredicateName1($entity_{rp,1}$, $entity_{rp,2}$, …), …

};

c) TSRL 类的实例化/实体化

实例化语句：

classname(entityname);

——类实体化，若创建载具类Vehicle的实体Car1，TSRL语句为：Vehicle(Car1)。

d) TSRL 类的相关谓词与函词表达式

在构造类时，除了用户自己构建的谓词/函词表达式，TSRL 会自动生成谓词、函词表达式。

谓词表达式：

——Isclassname(entity)，用于检验实体entity是否属于类classname。

——PropertyA(entity，value)，用于表示实体entity中属性PropertyA的值为value。

函词表达式：

——PropertyA(classentity) = entityPropertyA，指向实体classentity的属性entityPropertyA。

关于类和实体，TSRL 中内置的谓词/函词表达式部分如下：

——Member(entity，classname)用于检验实体entity是否属于类classname。

——PropertyOf(entity，classname)用于检验实体entity是否是类classname的属性。

——PropertyOf(entity1，entity2)用于检验实体entity1是否是实体entity2的属性。

——TypeOf(entity1) = classname用于取得实体entity1所属类的类名classname。

7 交通语义表示语言使用规范

7.1 通用使用原则

为保障TSRL在不同交通场景、不同交互主体间的统一应用，需遵循以下核心原则：

a) 语义一致性原则：交通实体、属性、关系的命名与描述需遵循本文件第 3 章“术语和定义”及

第 5 章“词法规范”，例如车辆实体统一用 IsVehicle(entity)标识，速度属性统一用 HasSpeed(entity, value)描述，避免同名异义或同义异名。

b) 逻辑严谨性原则：规则描述需基于一阶逻辑表示，明确条件与结论的逻辑关系，使用逻辑运算符构建推理链路，禁止逻辑矛盾或条件缺失。

c) 交互规范性原则：多主体间信息交互需通过 ask 和tell 关键字实现，交互内容需包含主体身份标识、信息类型及时间戳，确保信息可追溯。

d) 数据适配性原则：根据交通场景需求选择适配的数据类型，例如量化属性用数值常量，实体集合用表。

7.2 交通知识表达规范

7.2.1 场景描述规范

场景描述是对交通应用场景的整体界定，需明确主体、边界与目标，为后续状态描述与规则制定提供场景框架，具体规范如下：

a) 明确场景主体，通过 IsXXX(entity)类谓词界定主体类型，示例：IsVehicle(v)、IsTrafficLight(tl)。

b) 界定场景边界条件，包括空间关联、环境状态、设备状态，示例：HasTrafficLightState(id, state)。

7.2.2 状态描述规范

状态描述聚焦交通实体的实时属性与运行参数，是规则判断与逻辑推理的关键数据来源，需按实体类型明确描述规范，确保数据的准确性与可用性。例如，车辆状态需包含标识属性 HasVehicleID(v, id)与运动属性 HasSpeed(v, s)等。

7.2.3 规则描述规范

规则需采用rule 关键字定义，结构遵循“结论:-前提”（见 6.1.2），前提部分用逻辑运算符组合谓词，结论部分明确输出结果，如预警触发、动作推荐。

7.2.4 信息交互规范

信息交互是多交通主体间共享TSRL描述信息的关键流程，需明确发起、传递与数据管理的规范，确保信息在主体间高效、准确流转：

a) 交互发起：需先通过 IsEntityType(entity, type)验证主体身份，再通过 ask 关键字声明交互需求。

b) 信息传递：需通过tell 关键字封装信息，消息体包含信息类型标识、语义数据及时间戳。

c) 知识库操作：需通过 select/asserta/assertz/retract 关键字实现动态数据管理，确保交互后知识库信息的准确性与时效性。

7.2.5 逻辑推理规范

a) 推理前需验证基础谓词真值，排除无效实体或错误状态。

b)　推理过程需计算关键参数，参数计算需符合运算符规则。

c)　推理后需匹配规则库，满足规则前提则推导结论，推理结果需通过 print 语句打印或tell 语句传递至目标主体。

附录 A
（资料性附录）
道路交通场景知识表达

A.1　场景描述

聚焦城市信号灯控制交叉口，涉及车辆、信号灯、环境三个主体，排除极端天气、突发障碍物等干扰因素，核心目标为通过逻辑推理判断车辆闯红灯风险并触发预警。场景描述如下：

（1）场景主体：

——IsVehicle(v)：v为车辆。

——IsTrafficLight(tl)：tl为信号灯。

——IsEnv(env)：env为环境。

（2）场景边界：

——IsInSameIntersection(v，tl)：车辆v与信号灯tl处于同一交叉口。

——HasRoadCondition(env，“good”)：环境env中道路状况良好。

——HasWeatherCondition(env，“sunny”)：环境env天气为晴。

——HasTrafficCongestion(env，“no”)：环境env无拥堵。

——HasTrafficLightState(env，“normal”)：信号灯tl运行正常。

A.2　状态描述

（1）车辆状态谓词

——HasVehicleID(v，id)：车辆v的唯一标识为id。

——HasSpeed(v，s)：车辆v的行驶速度为s，单位：km/h。

——HasDistanceToStopLine(v，d)：车辆v距交叉口停止线距离为d，单位：m。

（2）信号灯状态谓词

——HasTrafficLightID(tl，id)：信号灯tl的唯一标识为id。

——HasTrafficLightColor(id，lane，color)：标识为id的信号灯，lane车道的相位颜色为color，取值为red/green/yellow。

——HasPhaseRemainTime(tl, t)：信号灯tl当前相位剩余时间为t，单位：s

——HasTrafficLightState(tl, state)：信号灯tl的运行状态为state，取值normal/abnormal。

（3）环境状态谓词

——HasRoadCondition(env, rs)：环境env的路面状态为rs，取值good/bad。

——HasWeatherCondition(env, w)：环境env的天气状况为w，取值sunny/rainy/fog等。

——HasTrafficCongestion(env, c)：环境env的拥堵状况为c，取值no/low/medium/high。

A.3 规则描述

a)闯红灯预警触发规则1：信号灯为绿灯/黄灯场景

自然语言描述：

对任意车辆v、信号灯tl、环境env，若满足:

1)同一交叉口

2)信号灯为绿灯或黄灯

3)绿灯剩余时间与黄灯剩余时间之和＞0

4)车辆匀速到达停止线时间＞绿灯剩余时间+黄灯剩余时间

5)道路良好

6)信号灯正常

7)无拥堵

8)天气晴

则车辆v触发闯红灯预警，即：车辆到达时信号灯已转为红灯，存在闯红灯风险。

TSRL描述：

```
#闯红灯预警触发规则1：当前信号灯为绿灯/黄灯场景
rule RedLightWarnTriggered1(v) :-
    IsVehicle(v) & IsTrafficLight(tl) & IsEnv(env) & InSameIntersection(v, tl)
    & (HasTrafficLightColor(tl, "green") | HasTrafficLightColor(tl, "yellow"))
    & HasGreenRemainTime(tl, t_g) & HasYellowRemainTime(tl, t_y) & ((t_g + t_y) > 0)
    & HasDistanceToStopLine(v, d) & HasSpeed(v, s) & ((d/(s*1000/3600)) > (t_g + t_y))
    & HasRoadCondition(env, "good") & HasTrafficLightState(tl, "normal")
    & HasTrafficCongestion(env, "no") & HasWeatherCondition(env，"sunny");
```

b)闯红灯预警触发规则2：信号灯为红灯场景

自然语言描述：

对任意车辆v、信号灯tl、环境env，若满足：

1)同一交叉口

2)信号灯为红灯

3)红灯剩余时间＞0

4)车辆匀速到达停止线时间＜红灯剩余时间

5)道路良好

6)信号灯正常

7)无拥堵

8)天气晴

则车辆v触发闯红灯预警，即：车辆到达时信号灯仍为红灯，存在闯红灯风险。

TSRL描述：

```
#闯红灯预警触发规则2：当前信号灯为红灯场景
rule RedLightWarnTriggered2(v) :-
    IsVehicle(v) & IsTrafficLight(tl) & IsEnv(env) & IsInSameIntersection(v, tl)
    & HasTrafficLightColor(tl, "red") & HasRedRemainTime(tl, t_r) & (t_r > 0)
    & HasDistanceToStopLine(v, d) & HasSpeed(v, s) & ((d/(s*1000/3600)) < t_r))
    & HasRoadCondition(env, "good") & HasTrafficLight(tl, "normal")
    & HasTrafficCongestion(env, "no") & HasWeatherCondition(env，"sunny");
```

c)预警关联驾驶建议规则

自然语言描述：

对触发闯红灯预警的车辆v，若车辆实例有效，则推荐执行刹车“Brake”动作，并输出对应的推荐减速度参数decel_val。

TSRL实现：

```
#预警关联驾驶建议规则
rule RecommendAction(v, "Brake") :-
    (RedLightWarnTriggered1(v) | RedLightWarnTriggered2(v)) & IsVehicle(v);
rule RecommendDecel(v, decel_val) :-
```

(RedLightWarnTriggered1(v) | RedLightWarnTriggered2(v)) & IsVehicle(v);

附录 B

（资料性附录）

轨道交通场景知识表达

B.1 场景描述

以铁路区段站货运列车调度与客运列车优先权场景为例，存在三列列车、多个轨道区段以及负责调度决策的控制中心，为准确描述它们之间的关系与行为进行交通交互场景语义化表达，场景描述如下：

（1）场景主体：

——IsTrack(track)：表示track是轨道。

——IsPassengerTrain(train1)：表示train1是客运列车。

——IsFreightTrain(train2)：表示train2是货运列车。

——IsControlCentre(cc)：表示cc是调度控制中心。

——IsDispatcher(dp)：表dp是调度员。

（2）场景边界：

——HasTrackStatus(track，status)：表示轨道track的状态为status，如Occupied、Free；

——HasPlannedRoute(train，track)：表示列车train的计划运行轨道为track；

——HasTrackAvailable(track)：表示轨道track空闲可用；

——HasRightSwitchPosition(track)：表示通往轨道track的道岔位置正确；

——HasTrackPassable(train，track)：表示允许列车train通过轨道track；

——OnTrack(train，track)：表示列车train正在轨道track上行驶；

——HasConflictCondition(track，“no”)：表示轨道track上当前无列车冲突；

——HasConflict(train1，train2，track)：表示列车train1与train2在轨道track上存在运行冲突；

——HasPriority(train，p)：表示列车train的运行优先级为p。

B.2 状态描述

——HasTrainSpeed(x，speed)：表示列车x的速度是speed；

——HasTrainAcceleration(x，acceleration)：表示列车x的加速度是acceleration；

——HasTrainCapacity(x，capacity)：表示列车x的容量是capacity；

——HasTrainStop(x，stop)：表示列车x的停靠站是stop；

B.3 规则描述

a) 冲突检测规则

自然语言描述：

若货运列车FreightTrain1占用轨道TrackA，同时客运列车PassengerTrain的计划路径为轨道TrackA，且客运列车PassengerTrain的优先级最高，则存在冲突。

TSRL描述：

#冲突检测规则

rule Conflict(PassengerTrain, FreightTrain1, TrackA) :-

OnTrack(FreightTrain1, TrackA)

& PlannedRoute(PassengerTrain, TrackA)

& HasPriority(PassengerTrain, 1);

b) 调度转线规则

自然语言描述：

若客运列车 PassengerTrain 与货运列车 FreightTrain1 存在冲突，且 TrackC 空闲无冲突，道岔正确，则调度中心生成转线指令。

TSRL描述：

#调度转线规则

rule Control(FreightTrain1, TransferTo(TrackC)) :-

HasConflict(PassengerTrain, FreightTrain1, TrackA)

& HasTrackAvailable(TrackC) ∧ HasConflictCondition(TrackC, “no”)

& HasRightSwitchPosition(TrackC);

附录 C

（资料性附录）

水运交通场景知识表达

C.1 场景描述

船舶对遇预警通过有效的数据处理和警报系统预警驾驶员前方即将发生的碰撞，来减少前方碰撞事故。为准确描述它们之间的关系与行为进行交通交互场景语义化表达，场景描述如下：

（1）场景主体：

——IsVessel(v)：表示个体v是船舶；

——IIsMariner(m)：表示个体m是船员；

——IsPort(x)：表示x是港口；

——IsDock(x)：表示x是码头；

（2）场景边界：

——IsClosing(x，y)：表示x与y正在接近；

——HasCollisionRisk(x，y)：表示x与y之间存在碰撞风险；

——HasAlertType(x,Alert)：表示x警示的类型为Alert；

——HasWaterTrafficRule(x,rule) 表示x遵循的水运交通规则为rule；

——HasSegmentType(w,type)　表示水道段w 的类型是type；

C.2 状态描述

——HasVesselPosition(v,position)：表示船舶v的位置是position；

——HasVesselSpeed(v,speed)：表示船舶v的速度是speed；

——HasShipAcceleration(x，a)：表示船舶x的加速度为a；

——HasVesselCourse(v,course)：表示船舶v的航向是course；

——HasVisibility(x，level)：表示能见度等级为level（如良好、中等、不良）；

——HasSeaCondition(x，condition)：表示海况条件为condition（如平静、中浪、大浪）；

C.3 规则描述

船舶对遇碰撞风险规则

自然语言描述：

这个规则表示如果船舶x与船舶y处于对遇局面，两船距离小于安全距离，且相对速度表明正在接近，则存在碰撞风险。船舶x将收到对遇碰撞预警警报。

TSRL描述：

```
#船舶对遇碰撞风险规则
rule CollisionRisk(x, y) :-
    IsVessel(x) & IsVessel(y) & Distance(x, y, d) & IsHeadOnSituation(x, y)
    & IsClosing(x, y) & GreaterThan(safe_distance, d);
```

T/ITS 0293.2-2025

中国智能交通产业联盟

标准

自主式交通系统 交通语义表示语言 第 2 部分：语法规范

T/ITS 0293.2-2025

北京市海淀区西土城路 8 号（ 100088）

中国智能交通产业联盟印刷

网址 ： http://www.c-its.org.cn

2025 年 11 月第一版 2025 年 11 月第一次印刷