

团体标准

T/ITS 0293.4-XXXX

自主式交通系统 交通语义表示语言 第4部分：推理框架

Autonomous transportation system

Traffic semantic representation language part 4: Inference Framework

(征求意见稿)

(本稿完成时间：2025年9月)

在提交反馈意见时，请将您知道的相关专利连同支持性文件一并附上。

20XX-XX-XX 发布

20XX-XX-XX 实施

中国智能交通产业联盟 发布

中国智能交通产业联盟

目 次

前 言 错误！未定义书签。

1 范围 错误！未定义书签。

2 规范性引用文件 错误！未定义书签。

3 术语和定义 错误！未定义书签。

4 缩略语 错误！未定义书签。

5 交通语义推理框架 错误！未定义书签。

6 交通推理规则表示 错误！未定义书签。

7 交通推理引擎运行机制 错误！未定义书签。

附录 A （资料性） 交通语义推理规则示例 11

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：待定

本文件主要起草人：待定

自主式交通系统 交通语义表示语言 第4部分：推理框架

1 范围

本标准规定了面向自主式交通系统的语义逻辑推理框架，涵盖基于知识图谱、交通语义表示规则和本体推理技术的体系结构、规则表示方法和推理机制。

本标准适用于道路、铁路与水运等多领域交通系统的语义一致性建模与推理，支持多智能体之间的互操作与语义协同。旨在支持智能体之间的信息语义一致性与逻辑一致性，提升跨主体间的语义协同效率与行为决策准确性。

2 规范性引用文件

本标准引用下列文件中的条款作为本规范的条款。凡未注明日期的引用文件，其最新版（包括所有修改单）适用于本规范：

W3C 网络本体语言2（OWL 2 Web Ontology Language）

W3C SPARQL 1.1 查询语言（SPARQL 1.1 Query Language）

W3C 语义网规则语言（Semantic Web Rule Language (SWRL)）

ISO/IEC 24707: 通用逻辑（Common Logic(CL)）

3GPP TS 24.379: 关键任务一键通（Mission Critical Push To Talk (MCPTT)）

3 术语和定义

本规范采用下列术语和定义：

3.1

语义推理 semantic inference

基于已有事实和规则，导出隐含知识的过程。

3.2

交通智能体 traffic agent

具有自主决策能力的交通参与主体，如车辆、路侧设备、调度系统等。

3.3

本体推理 ontology inference

基于 OWL 公理体系进行的一致性检查、分类推理与实例化推理。

3.4

交通语义规则 traffic semantic rule

表示交通实体间关系与行为逻辑的形式规则，用以支持语义推理。

3.5

合一 unification

将两个谓词匹配并绑定变量的过程。

3.6

前向链接 forward chaining

从事实出发逐步推出结论的推理方式。

3.7

反向链接 backward chaining

从查询目标出发回溯匹配规则与事实的推理方式。

3.8

知识图谱 knowledge graph

以 RDF 三元组为基础，结合本体模型和推理规则，统一表示交通实体、关系与属性的知识库。

3.9

增量推理 incremental inference

对动态交通数据进行局部更新和局部推理的模式，避免全局重算。

4 缩略语

ABox: 断言事实 (Assertional Box)

TBox: 术语公理 (Terminological Box)

DL-Safe: 描述逻辑安全规则 (Description Logic Safe Rules)

SIP: 会话发起协议 (Session Initiation Protocol)

MSRP: 消息会话中继协议 (Message Session Relay Protocol)

MQTT: 消息队列遥测传输 (Message Queuing Telemetry Transport)

CoAP: 受限应用协议 (Constrained Application Protocol)

SPARQL: SPARQL 协议与 RDF 查询语言 (SPARQL Protocol and RDF Query Language)

TSRL: 交通语义表示语言 (Traffic Semantic Representation Language)

TKB: 交通知识库 (Transportation Knowledge Base)

TRB: 交通规则库 (Transportation Rule Base)

5 交通语义推理框架

5.1 交通语义推理框架

交通语义推理框架由数据与事件采集、语义集成、知识图谱、推理引擎、查询与解释以及行动与消息等六个层次构成,如图1所示。框架通过多源数据采集与标准化处理,统一转化为交通语义表示语言;依托知识图谱和规则库支撑,推理引擎实现事件分类、规则触发与增量更新;在此基础上,提供语义查询与推理溯源功能,并将推理结论通过协议适配与消息封装转译为控制指令、告警或服务建议,实现从数据感知到智能决策的闭环。

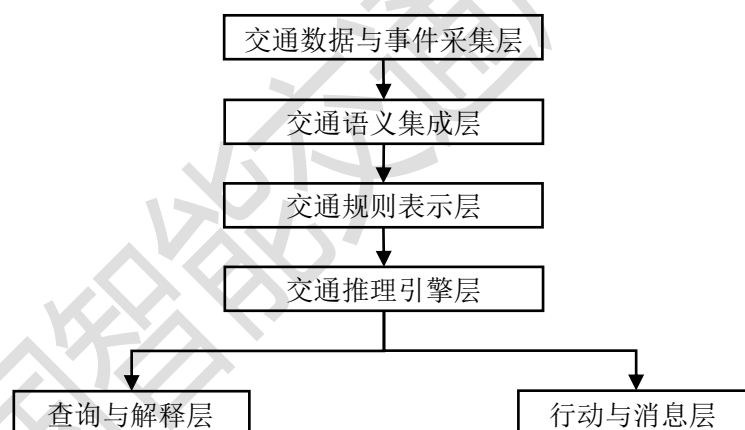


图1 交通语义推理框架

5.2 交通推理机制

交通语义推理框架的运行机制依托于从交通环境中获取的实时数据,并通过多层次处理与推理形成语义化认知并输出,推理机制如图2所示。

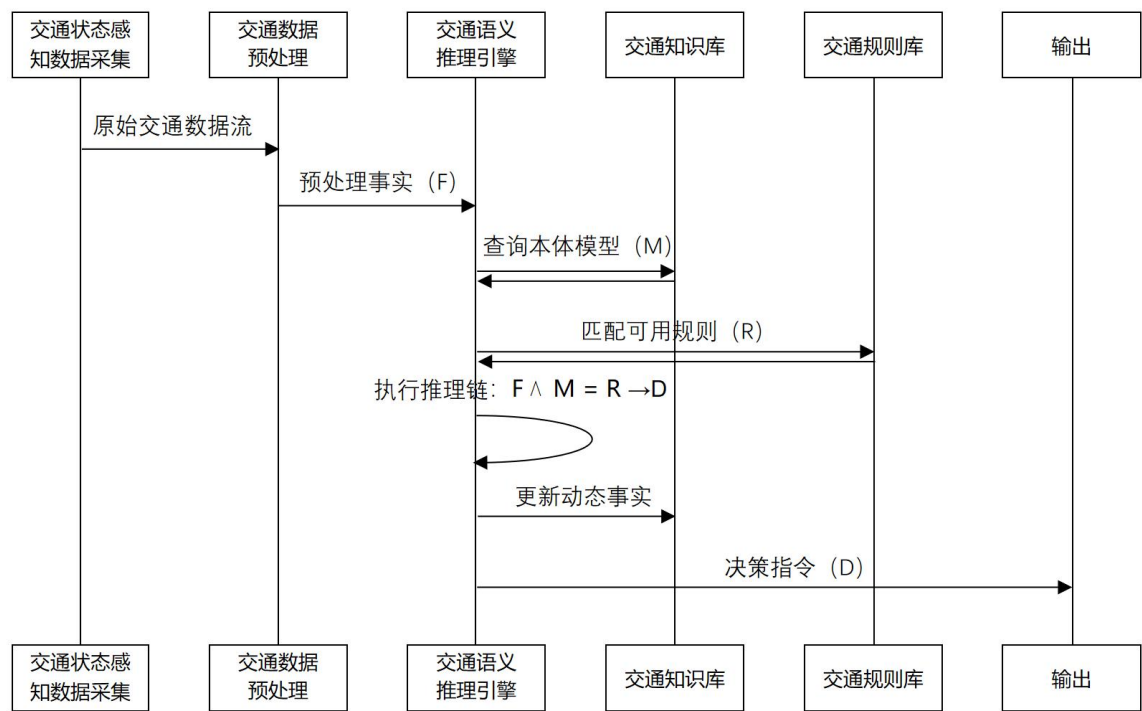


图2 推理机制

5.3 交通时空推理

5.3.1 交通时空数据

交通时空数据建模的目标：建立统一的时空信息表示方式，为推理提供基础，具体步骤：

- 定义交通主体（车辆、列车、信号设备、道路设施）在空间中的位置表示方式，如坐标系、轨道区段编号、道路节点与边的拓扑关系；
- 定义时间维度，包括绝对时间（时刻表、红绿灯周期）、相对时间（事件持续时间、延迟）、时间窗约束（必须在某一时间区间内完成的动作）；
- 建立“时空点”“时空区间”“时空轨迹”等基础数据结构。

5.3.2 交通时空关系

交通时空关系抽取的目标：从建模后的数据中提取能被推理利用的时空关系，常见的关系：

- 交通空间关系：相邻（Adjacent）、包含（Contains）、冲突（Conflict）、可达（Reachable）；
- 交通时间关系（Allen 时间代数）：先于（Before）、同时（During）、重叠（Overlap）、延续（Meets）；
- 交通时空复合关系：如“在时间区间T内，主体A与主体B占用同一空间S”，用于冲突检测或协同判断。

5.3.3 交通约束与交通规则

交通约束与交通规则建模的目标：用时空约束表达交通运行规则，确保推理结果符合安全与调度要求，具体内容：

- 交通时空安全约束：如“同一区段在同一时刻最多允许一列车”；
- 交通时空容量约束：如“车道在某一时间段内可容纳的车辆数”；
- 交通时空逻辑规则：如“若事件A在B之前完成，且两者位于相邻区段，则允许主体进入下一区段”；
- 优化目标：最短旅行时间、最小延误、最大化吞吐量等。

5.3.4 交通推理机制

- 交通推理机制的目标：基于时空模型和规则，推导出交通主体的行为与状态演化，具体执行步骤：
- 交通时空一致性检测：检查交通主体行为是否违反时空约束（如列车冲突、道路占用重叠）；
 - 交通轨迹预测与匹配：根据当前速度与路径，预测未来轨迹，并判断与其他主体轨迹是否冲突；
 - 交通调度/控制推理：生成满足时空约束的调度策略（如调整运行时刻表、改变车道/股道分配）；
 - 动态修正：若外部环境变化，推理过程需实时迭代。

6 交通推理规则表示

交通语义推理规则是在TSRL已定义的交通领域本体与知识表示体系的基础上，利用逻辑表达式和运算符对交通实体、属性及其关系进行条件描述，并通过交通推理引擎执行以生成新的语义信息或触发相应动作的形式化定义。

6.1 交通推理规则结构模式

交通推理规则应具备的结构元素如表1所示：

表1 交通推理规则所包含元素

元素名称	描述	要求
RuleID	交通规则唯一标识符	全局唯一
RuleName	交通规则简要名称	反映交通规则用途
Condition（前提）	交通语义逻辑表达式,用交通语义表示语言描述	必须引用交通知识库已定义的谓词与参数类型
Conclusion（结果）	交通推理生成的交通事实或	与交通知识库概念一致

	动作	
Priority	交通规则优先级	决定交通规则触发顺序

6.2 交通推理规则表达模版

交通推理规则首先定义交通实体的层级关系、约束与一致性检查，并表达复杂的时序、空间、行为逻辑。交通推理规则采用扩展的谓词逻辑表达式形式书写，使用前件-后件（IF-THEN）模式，并支持标准逻辑运算符与内建函数。

通用表达模板：

RuleID <RuleName>

<结论谓词表达式>:- <谓词表达式1> [逻辑连接符] <谓词表达式2> ...

[Priority = n]

谓词表达式组成：

- a) 谓词（Predicate）：表示实体、属性或关系；
- b) 变量（Variable）：表示规则中的通用符号；
- c) 逻辑连接符（ $\wedge$, $\vee$, $\neg$ ）：用于连接多个条件；
- d) 内建函数（Built-in）：用于比较、计算或转换。

6.3 交通推理规则编写要求

交通推理规则编写要求：

- a) 唯一性：每条规则必须具有唯一RuleID；
- b) 可解释性：规则名称与元数据应能明确反映其用途；
- c) 一致性：条件与结果使用的概念、关系应与交通本体定义一致；
- d) 模块化：单条规则应只描述单一逻辑关系，便于维护与组合；
- e) 可机读性：规则应符合规定的语法格式，支持推理引擎直接解析。

6.4 交通推理规则分类

6.4.1 交通状态识别规则

定义：用于从原始感知数据中识别出交通系统当前状态，例如交通主体的位置、速度、信号状态等。

6.4.2 交通行为触发规则

定义：用于基于当前交通状态条件，触发相应的交通行为或系统响应，如紧急刹车。

6.4.3 交通优先级推理规则

定义：用于处理交通参与者之间的优先通行关系推理，如让行、避让等。

6.4.4 交通语义关系扩展规则

定义：用于对交通概念之间的语义关系进行扩展或补全，如“紧急车辆”隐含“可越过红灯”。

6.4.5 交通时序推理规则

定义：用于处理具有时间顺序关系的交通行为逻辑，如先到先行、限时通行等。

6.4.6 交通空间逻辑规则

定义：用于判断交通参与者之间的交通空间位置关系，如并道、跟车距离、盲区等。

6.4.7 交通协同推理规则

定义：用于交通多智能体间的协同认知与动作协调逻辑，例如变道协商、交汇避让等。

6.4.8 交通异常检测规则

定义：用于识别不符合交通规范或常理的交通行为状态，如逆行、占用应急车道等。

6.4.9 交通规则一致性与冲突处理

所有交通规则需满足以下约束以保证系统一致性：

- a) 不得存在结论自相矛盾的规则组合；
- b) 所有谓词语义应在知识本体中有明确定义；
- c) 优先级规则间需设定冲突解决策略（如优先权覆盖、时间优先等）。

7 交通推理引擎运行机制

7.1 交通推理引擎的运行机理

7.1.1 TSRL 语义推理机制

TSRL推理构建在交通语义本体模型之上，语义推理机制包括以下步骤：

- a) 一致性检查执行
 - 1) 对当前交通知识库进行扫描，验证每个交通个体是否满足类的定义约束；
 - 2) 若发现冲突，推理器生成冲突报告并标记不一致实体。
- b) 交通类型推断流程
 - 1) 遍历所有新加入或更新的交通实体；

2) 根据类层级和属性限制，推断其是否属于更具体的子类，并在交通知识库中自动更新实体类型。

c) 交通属性继承与关系推导

对交通实体属性执行逻辑扩展：

- 1) 若交通属性为传递性（如“在同一路线上”），则自动递推出间接关系；
- 2) 若交通属性为对称性（如“相邻”），则在反向方向补全关系；
- 3) 若存在交通属性链（如“车-路-区段”），则推导出更高层次的关系（如“车属于区段”）。

d) 更新与持久化

- 1) 将推理产生的新类型或关系写入交通知识库；
- 2) 若推理结果影响后续规则执行，则触发相应的交通规则引擎。

7.1.2 TSRL 规则推理机制

TSRL提供逻辑规则扩展，其运行机制具体分为以下执行步骤：

a) 前向推理

- 1) 当传感器、事件或外部系统输入新交通事实数据时，系统将其加入交通知识库；
- 2) 触发依赖该交通事实的交通规则前提条件，逐条检查是否满足；
- 3) 对于满足条件的交通规则，执行结论部分，将新推导出的事实加入交通知识库；
- 4) 将新增交通事实递交回调度器，可能继续触发链式推理。

b) 反向推理

- 1) 当用户或系统提出目标查询（如“是否可以开放某条进路”），交通推理引擎将目标作为待证明结论；
- 2) 在交通规则库中查找结论部分与目标匹配的规则；
- 3) 将交通规则的前提条件转化为新的子目标，并检查交通知识库是否满足；若未满足，则递归查找能够证明该条件的其他交通规则；
- 4) 若所有前提被证明，最终确认目标结论成立；若存在前提无法证明，则返回失败或部分成立。

c) 增量推理

- 1) 系统维护依赖关系图谱（Dependency Graph），记录每条交通推理结果与输入数据、规则的依赖关系；
- 2) 当新交通数据加入或旧数据被修改/删除时，系统首先定位受影响的交通规则与推理链；

- 3) 仅重新计算受影响的局部规则，而非全量运行整个推理库；
- 4) 将更新后的结果即时替换旧结论，确保交通知识库保持一致性与实时性。

d) 运行调度与优化

- 1) 交通推理引擎在运行时根据任务优先级选择前向或反向模式，必要时混合使用；
- 2) 若系统负载过高，增量推理机制会限制全局推理，仅执行必要子集，以保证实时响应。

7.1.3 推理编排机制

为了高效、可靠地管理和调度底层交通推理引擎的执行过程，需要高层的交通推理编排策略：

a) 策略控制：核心问题是如何集成与应用推理结果。常用的两种策略是：

1) 物化推理：预先执行所有可能的推导，并将推理结果作为显式的事实存储到知识库中。后续检索查询直接从事实库中获取，响应速度极快，但缺点是存储开销大，且当原始数据变更时，需重新进行物化。

2) 查询重写：在用户或系统发送查询时，即时分析查询语句，递归地将其中涉及的类和属性限制、规则等“重写”扩展为一个庞大的、包含所有需要推导步骤的逻辑表达式的复杂查询，然后由基础推理来完成查询应答。此方式节省存储空间，但复杂查询的响应时间可能较长且可能难以实现。

b) 冲突解决

在实践中，不同的规则或推理路径可能推导出相互矛盾或有优先权冲突的结论，系统需要有效的冲突检测和消解机制：

- 1) 规则优先级：为规则显式指定优先级数值或层级，高优先级规则覆盖低优先级规则；
- 2) 时间最近规则（或事实）：采用时间戳更新的规则或数据来源推导出的结论优先；
- 3) 特例覆盖原则（Specificity）：处理更具体情况的规则优先于处理一般情况的规则；
- 4) 溯源与解释：为了让系统的决策具有透明性和可追溯性，特别是对于产生重要影响的推理结果至关重要；
- 5) 规则轨迹：明确显示得出该结论所经过的精确规则序列；
- 6) 证据链：清晰列出支撑每次规则触发或本体推导的原始输入数据源。这不仅便于问题排查和结果验证，也有助于提升用户对系统决策的信任度。

7.2 交通推理引擎的结构设计

7.2.1 交通推理引擎结构

交通语义推理框架通过分层化设计实现从数据采集到语义推理再到行动反馈的全流程支撑。其总体

目标是在多源异构交通数据的基础上，经由语义集成、知识图谱与推理引擎处理，形成可查询、可解释、可交互的语义化知识体系，并最终通过标准化的协议接口实现与外部系统、交通主体的互操作，推理引擎结构如图3所示。

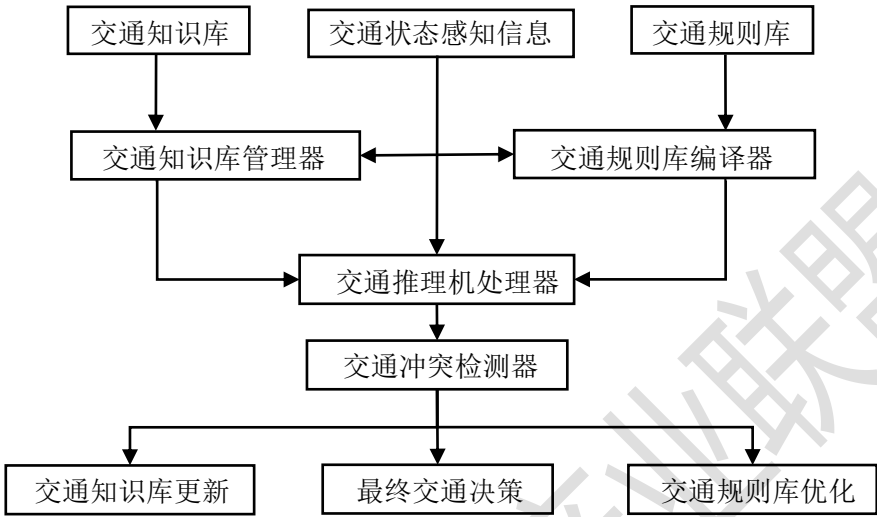


图3 推理引擎结构

7.2.2 交通推理引擎流程

推理引擎推理流程如图4所示，交通推理步骤如下：

a) 接收查询并初始化

用户通过ASK接口提交查询，将该初始目标推送到交通执行栈，同时初始化本次推理的交通动态表条目与交通绑定表。

b) 检索候选规则

交通执行栈根据目标向交通规则库发起“请求匹配规则”，规则库返回候选规则集合，可按优先级或启发式排序。

c) 逐条规则处理

对每条候选规则，执行栈保存当前绑定/动态表快照以便回溯，然后读取该规则的前提。

d) 读取事实候选

对每个前提，执行栈向交通知识库请求可能匹配的事实，知识库返回匹配事实集合。

e) 前提匹配与变量合一

对每个候选事实与前提做匹配合一：在匹配过程中读取/更新交通绑定表，将成功的变量约束写入绑定表。若前提本身是子目标，则把该子目标递归推入执行栈。

f) 记录临时状态与中间结果

每次匹配尝试都写入交通动态表，以保留中间证据与约束；需要时，交通动态表可向执行栈或外部模块提供中间结果。

g) 回溯策略

若某前提匹配失败，使用动态表/绑定表中的快照恢复变量绑定状态，然后选择下一个候选事实或回退到上一层选择下一个候选规则继续尝试。

h) 规则成立与结论返回

若某规则的所有前提均成功匹配，则该规则成立：执行栈生成/返回匹配的交通事实，并将结果逐层返回到初始目标处。

i) 目标失败判定

若所有候选规则与事实均尝试完仍无解，则该目标判为失败，执行栈返回失败信息。

j) 清理与输出

本次推理结束后，统一释放/清空交通绑定表中的临时变量绑定和交通动态表的临时状态，保持知识库一致性；最后把最终结论与必要的中间证据返回给ASK接口。

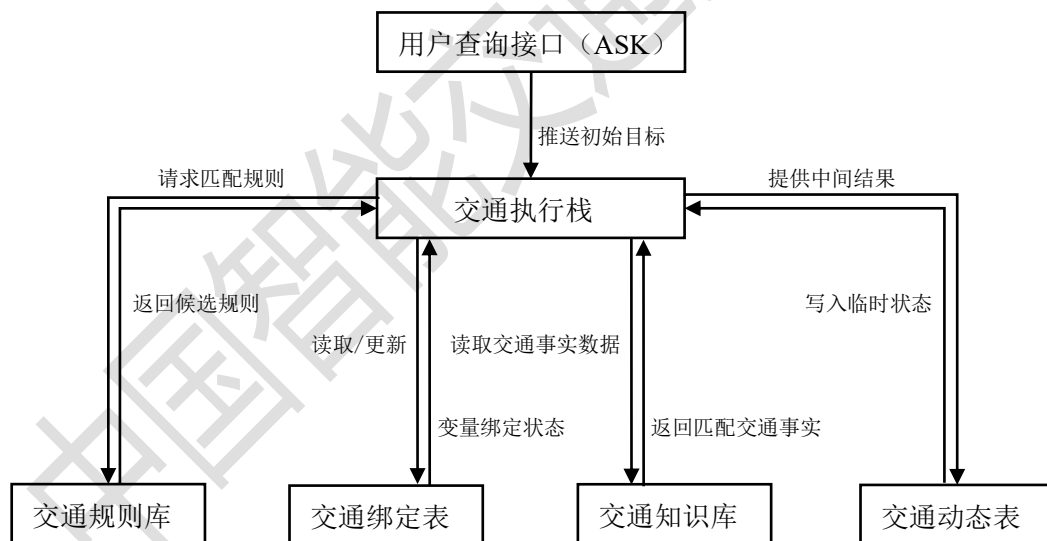


图4 推理流程

通过上述机制，交通语义逻辑推理机可有效支撑多交通方式、多交通主体、多场景的智能交通推理任务。

附录 A

(资料性附录)

交通语义推理规则示例

A.1 概述

本附录给出了部分最经典的交通语义推理规则。

A.2 交通语义推理规则

(1) 道路/高速公路规则

Rule1 <状态识别规则>

$$\text{OverspeedVehicle}(v,t) \text{ :- } \text{Vehicle}(v) \wedge \text{HasSpeed}(v,t,s) \wedge \text{GreaterThan}(s,30);$$

识别速度超过30km/h的城市道路车辆为超速状态。

Rule2 <行为触发规则>

$$\text{MustStop}(v,t) \text{ :- } \text{Approaching}(v,\text{SignalizedIntersection},t) \wedge \text{SignalColor}(t,\text{red}) ;$$

车辆接近信号灯路口且红灯亮时必须停车。

Rule3 <优先级推理规则>

$$\text{HasRightOfWay}(v1,v2) \text{ :- } \text{IsEmergencyVehicle}(v1) \wedge \neg \text{IsEmergencyVehicle}(v2) \wedge$$

$$\text{ConflictAtJunction}(v1,v2,t);$$

紧急车辆与非紧急车辆在路口冲突时拥有优先通行权。

Rule4 <语义关系扩展规则>

$$\text{SchoolBus}(v) \text{ :- } \text{CanStopInMovingTraffic}(v) \wedge \text{RequiresAllStop}(v);$$

校车语义隐含可路上停车并触发其他车辆完全停止。

Rule5 <时序推理规则>

$$\text{HasPriority}(v1,v2) \text{ :- } \text{ArrivalAtMerge}(v1,t1) \wedge \text{ArrivalAtMerge}(v2,t2) \wedge \text{LessThan}(t1,t2);$$

在合流点先到达车辆拥有优先通行权。

Rule6 <空间逻辑规则>

$$\text{TailgatingWarning}(v2,t) \text{ :- } \text{IsBehind}(v2,v1) \wedge \text{DistanceBetween}(v1,v2,d,t) \wedge$$

$$\text{LessThan}(d,\text{SafeDistance}(v1,s,t));$$

后车与前车间距小于安全跟车距离构成追尾警告。

Rule7 <协同推理规则>

ShouldCreateGap(LeftVehicle,v,t):- RequestLaneChange(v,Left,t) $\wedge$ HasSpace(LeftVehicle,t) $\wedge$ FlashLight(v,3,t);

车辆请求左变道并闪烁灯光时，邻居车辆应主动让出空间。

Rule8 <异常检测规则>

IllegalShoulderUseWarning(v,t):- Occupied(v,HardShoulder,t) $\wedge$ $\neg$ IsEmergency(v) $\wedge$ $\neg$ IsBreakdown(v);

非紧急状态占用应急车道触发违规行为警告。

(2) 轨道系统规则

Rule1 <状态识别规则>

CriticalFaultStatus(SegmentA,t):- SignalFailure(SegmentA,t) $\wedge$ TrackOccupied(SegmentA,t);

轨道区段信号故障且被占用时识别为严重故障状态。

Rule2 <行为触发规则>

ActivateBarrier(LevelCrossing,t):- ApproachingTrain(v,LevelCrossing,500m,t);

列车距道口500米时触发栏杆降落。

Rule3 <优先级推理规则>

YieldPriority(v2,v1):- HighSpeedRail(v1) $\wedge$ FreightRail(v2) $\wedge$ SharedTrack(v1,v2,t);

客货混行线路中货运列车须让行高速列车。

Rule4 <语义关系扩展规则>

MaintenanceActivity(x):- ImposesSpeedRestriction(x,40km/h) $\wedge$ RequiresSafetyZone(x,50m);

维修活动相关轨道区段触发40km/h限速及50米安全区。

Rule5 <时序推理规则>

DelayedStatus(v1,t):- DepartureTime(v1,StationA,t1) $\wedge$ ArrivalTime(v1,StationB,t2) $\wedge$ GreaterThan(t2-t1,ScheduleTolerance);

实际运行时间超计划容忍值时标记为晚点。

Rule6 <空间逻辑规则>

CollisionRisk(Block,t):- TrainInBlock(v1,Block,t) $\wedge$ TrainInBlock(v2,Block,t) $\wedge$ $\neg$ IsCoupled(v1,v2);

同一闭塞区内两列非连接列车构成碰撞风险。

Rule7 <协同推理规则>

ApproveRerouting(v,Platform2,t):- RequestPlatformChange(v,Platform2,t) $\wedge$ AvailableCapacity(Platform2,t);

列车请求变更站台且容量充足时应批准路径调整。

Rule8 <异常检测规则>

OverspeedWarning(v,t):- DetectedSpeed(v,s,t) $\wedge$ MaxAllowed(section,Smax) $\wedge$ GreaterThan(s,Smax+10%);

检测到速度超过区段限速10%时触发超速警告。

(3) 水运航行规则

Rule1 <状态识别规则>

LowClearanceStatus(v,t):- Vessel(v) $\wedge$ Draft(v,d,t) $\wedge$ ChannelDepth(c,cd,t) $\wedge$ GreaterThan(d,cd*0.8);

船舶吃水大于航道深度80%时为低富余水深状态。

Rule2 <行为触发规则>

InitiateCPAProtocol(v1,v2):- DetectedCollisionCourse(v1,v2,t,15min) $\wedge$ RangeClosingRate(r) $\wedge$ GreaterThan(r,0);

检测到15分钟内碰撞航向且距离缩减时启动避碰程序。

Rule3 <优先级推理规则>

GiveWay(v1,v2):- PoweredVessel(v1) $\wedge$ SailingVessel(v2) $\wedge$ CrossingSituation(v1,v2,t);

机动船与帆船交叉相遇时机动船须避让。

Rule4 <语义关系扩展规则>

VLCC(x) :- RequiresTugEscort(x) $\wedge$ ImposesTrafficSeparation(x);

超级油轮(VLCC)隐含需拖轮护航及触发分道通航。

Rule5 <时序推理规则>

MustExecuteHoldingPattern(v,t):- ETA(v,PortEntry,t1) $\wedge$ TidalWindow(start,end,t2) $\wedge$ $\neg$ WithinTime(t1,start,end);

预计到港时间不在潮汐窗口内须执行盘旋待命。

Rule6 <空间逻辑规则>

SinglePassageRequired(c,t):- InNarrowChannel(v,t) $\wedge$ Width(v,w) $\wedge$ ChannelWidth(c,cw) $\wedge$ L

essThan(cw-w, SafeMargin);

船舶宽度使航道富余小于安全值时触发单向通行。

Rule7 <协同推理规则>

ConfirmBerthAssignment(Berth,v,30min):- RequestBerthChange(v,Berth,t) $\wedge$ TugAvailable(tug,t)

$\wedge$ PilotAvailable(pilot,t);

船舶请求变更泊位且拖轮、引航员就绪时分配新泊位。

Rule8 <异常检测规则>

LostContactEvent(v,t):- AISTransmissionGap(v,t, Δt) $\wedge$ greaterThan(Δt ,10min) $\wedge$ InTrafficSeparationScheme(v,t);

在分道通航区内AIS信号丢失超10分钟触发失联警告。

附录 B

(资料性附录)

交通语义推理流程示例

B.1 交通语言推理场景

一列客运列车PassengerTrain即将驶入车站范围，其图定运行有严格的时效要求，享有最高优先级（priority=1）。而此时，一列货运列车FreightTrain1正在占用主轨道TrackA，该轨道同时位于PassengerTrain的计划通行路径上。另一列货运列车FreightTrain2则处于侧线TrackB待避状态。站调系统需在PassengerTrain到达前，指挥FreightTrain1安全、及时地转离主轨道（转至空闲的TrackC），为PassengerTrain让出通行空间，同时确保FreightTrain2保持待避状态，避免误入冲突区域。整个过程需要在严格的时空约束和安全规则下进行。

B.2 交通语义推理

(1) 初始状态检测

轨旁设备实时不间断将当前线路占用情况和列车运行计划实时上传至调度中心。

Send by FreightTrain1&FreightTrain2&PassengerTrain: OnTrack(FreightTrain1, TrackA), OnTrack(FreightTrain2, TrackB), PlannedRoute(PassengerTrain, TrackA) Priority(PassengerTrain, 1);

Received by ControlCentreTrafficDispatcher: OnTrack(FreightTrain1, TrackA), OnTrack(FreightTrain2, TrackB), PlannedRoute(PassengerTrain, TrackA) Priority(PassengerTrain, 1);

含义：各辆列车分别向调度中心报告：FreightTrain1 正在占用主轨道 TrackA，FreightTrain2 处于侧线 TrackB，PassengerTrain 的计划运行路径是 TrackA，PassengerTrain 的运行优先级最高（priority=1）。

(2) 冲突检测与调度决策

Inferred by ControlCentreTrafficDispatcher: Conflict(PassengerTrain, FreightTrain1, TrackA), Control(FreightTrain1, TransferTo(TrackC));

Send by ControlCentreTrafficDispatcher: Control(FreightTrain1, TransferTo(TrackC));

Received by TracksideEquipment: Control(FreightTrain1, TransferTo(TrackC));

含义：调度中心通过逻辑推理识别到：PassengerTrain与FreightTrain1在TrackA上存在运行冲突；据优先级规则（PassengerTrain优先），调度中心生成决策并发送给轨旁设备：指令FreightTrain1 转移至

空闲轨道TrackC，避免冲突。

（3）路径可用性反馈

Send by TracksideEquipment: TrackAvailable(TrackC), NoConflict(TrackC), RightSwitchPosition(TrackC);

Received by ControlCentreTraicDispaicher: TrackAvailable(TrackC), NoConflict(TrackC), RightSwitchPosition(TrackC);

含义：轨旁设备实时不间断向调度中心反馈TrackC的状态：TrackC空闲可用；当前无其他列车占用或冲突；相关道岔处于正确的定位状态，允许进入。这为调度中心确认转线方案的可行性提供了安全保障。

（4）转线授权

Inferred by ControlCentreTrafficDispatcher: LetGo(FreightTrain1, TrackC);

Send by ControlCentreTrafficDispatcher: LetGo(FreightTrain1, TrackC);

Received by FreightTrain1: LetGo(FreightTrain1, TrackC);

含义：调度中心向FreightTrain1下达控制指令：授权FreightTrain1驶入TrackC，完成FreightTrain1的转线动作，从而释放TrackA。这是从“调度决策”到“实际执行”的关键环节。

（5）主轨道状态更新

Send by TracksideEquipment: TrainPosition(PassengerTrain), TrackAvailable(TrackA), NoConflict(TrackA), RightSwitchPosition(TrackA);

Received by ControlCentreTraicDispaicher: TrainPosition(PassengerTrain), TrackAvailable(TrackA), NoConflict(TrackA), RightSwitchPosition(TrackA);

含义：轨旁设备实时不间断向调度中心传输PassengerTrain的位置信息和轨道状态，直道确认PassengerTrain已经通过该区段，TrackA空闲无冲突，道岔已经正确定位，允许FreightTrain1按计划进入。这为后续下达通行许可提供条件。

（6）进路许可下达

Inferred by ControlCentreTrafficDispatcher: TrackAPassable(FreightTrain1, TrackA);

Send by ControlCentreTrafficDispatcher: TrackAPassable(FreightTrain1, TrackA);

Received by FreightTrain1: TrackAPassable(FreightTrain1, TrackA);

含义：调度中心基于轨道状态和优先级规则，向FreightTrain1下达“进路许可”，即FreightTrain1可以进入并通过TrackA；PassengerTrain已经驶离区间，FreightTrain2保持待避，整个进路安全。

T/ITS 0293.4-XXXX

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1部分：通用术语

T/ITS 0293.4-XXXX

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X月第一版 2025年X月第一次印刷