No Need Knowing Numerous Neighbours
Towards a realizable interpretation of MLSL

Martin Fränzle[1⋆], Michael R. Hansen[2⋆⋆], and Heinrich Ody[1⋆⋆⋆]

[1] Department of Computing Science, University of Oldenburg, Germany
fraenzle@informatik.uni-oldenburg.de, heinrich.ody@uni-oldenburg.de
[2] DTU Compute, Technical University of Denmark mire@dtu.dk

Abstract. The Multi-Lane Spatial Logic MLSL introduced by Hilscher et al. in [4] is a two-dimensional spatial logic geared towards modelling and analysis of traffic situations, where the two dimensions are interpreted as the lanes of a road and the distance travelled down that road, respectively. The intended use of MLSL is for capturing (and reasoning about) guards and invariants in supervisory control schemes for highly automated driving [11]. Unfortunately, the logic turns out to be undecidable [6, 7, 10], rendering implementability and thus the actual use of such guard conditions in supervisory control questionable in general.
We here show that under a reasonable model of technical observation of the traffic situation, the actual decidability and implementability issues take a much more pleasing form: given that a real autonomous car can only sample state information of a finite set of environmental cars in real-time, we show that it is decidable whether truth of an arbitrary MLSL formula can be safely determined on a given sample size. For such feasible formulas, we furthermore state a procedure for determining its truth value based on such a sample.

Keywords. Highly automated driving, supervisory control schemes, spatial logic, decidability.

1 Introduction

To show collision freedom of cars of a controller for traffic manoeuvres on motorways we have to show that our controller does not cause collisions with obstacles or other traffic participants, e.g. cars. This is a verification problem of hybrid systems, where the discrete part models the control layer and the continuous part models the car dynamics.

To simplify this problem Hilscher et al. [4] suggested to consider collision freedom separated from the car dynamics and developed an abstract model to

⋆ Work of the author was partially supported by Deutsche Forschungsgemeinschaft within the Transregional Collaborative Research Center SFB/TR 14 AVACS.
⋆⋆ Work of the author was partially supported by the Danish Research Foundation for Basic Research within the IDEA4CPS project.
⋆⋆⋆ Work of the author was partially supported by Deutsche Forschungsgemeinschaft within the Research Training Group DFG GRK 1765 SCARE.

represent the positions of cars on the motorway and Multi-Lane Spatial Logic (MLSL) to specify and reason about properties of such models. In [5] this work was extended to address safety properties for traffic on country roads.

The logic MLSL is a multi-dimensional interval logic with a discrete dimension of lanes and a continuous dimension of travel distance. It is tailored towards reasoning about traffic manoeuvres, a very specific use case, and the design of MLSL is primarily inspired by Duration Calculus [2], Propositional Interval Temporal Logic [9] and Shape Calculus [12]. The first two logics are usually used to describe temporal properties, while Shape Calculus is considered as a spatio-temporal logic. Other interval logics having some similarities with MLSL are CDT, a modal logic for chopping intervals [13] and Halpern-Shoham-logic [3], a logic based on Allen's interval relations [1]. While Duration Calculus and Shape Calculus allow quantitative reasoning, the other logics only permit qualitative reasoning.

Being inspired by interval logics, MLSL has similar strengths and weaknesses as such logics, that is, MLSL is expressive and satisfiability problems for MLSL are typically undecidable [6,7,10]. The satisfiability (and model-checking) problem for a fragment of MLSL is shown decidable in [10] when a fixed maximal bound on the number of cars in traffic situation in imposed.

An application of MLSL has been in connection with the definition of lane change controllers for motorways [4,7]. These controllers are defined as extended timed automata which, additionally to clock constraints, may have MLSL formulas as transition guards. While MLSL reasons about a countable infinite set of car identifiers, technical surveillance of the traffic situation by car can in real-time only harvest information about a finite set (also called a sample) of neighbouring cars. Evaluation of guard or invariant conditions employed in supervisory control can consequently only resort to such a finite sample. Due to disturbances and/or imperfect equipment, different samples may be drawn from the same traffic situation. This immediately causes two questions:

1. Is the evaluation of a guard (a MLSL formula) independent from the particular sample drawn, which may vary within reasonable bounds?
2. Will the evaluation of the guard on a sample provide reliable information on its validity over all cars, including the hidden cars, that is, cars that are not observed by the equipment?

Extending MLSL with a so-called scope formula $\{c_1, c_2, \ldots, c_n\} : \phi$ which restricts the variation of cars in the evaluation of ϕ to $c_1, c_2, \ldots, c_n$, we arrive at a logic in which the above two questions can be formalized and studied. The aim of this work is to get results for the above questions under reasonable assumptions.

In Section 2 we introduce our extension of MLSL called Multi-Lane Spatial Logic with Scope (MLSLS), and in Section 3 we formalize the above questions. In Section 4, it is shown that the satisfiability problem for so-called well-scoped MLSLS formulas is decidable. The technique for showing this decidability result is strongly based on [10], where the satisfiability problem for MLSL is reduced to the satisfiability problem for quantified linear integer-real arithmetic

(QLIRA). A difference is that while the decidability result of [10] is based on a bound on the number of cars considered, the decidability for well-scoped MLSLS formulas is based on a syntactical restriction on formulas, that is, every existentially quantified formula $\exists c.\psi$ occurs within the context of a scoped formula $\{c_1, c_2, \ldots, c_n\}:\phi$, thereby restricting the range of c to a finite set given by $\{c_1, c_2, \ldots, c_n\}$. In Section 5 we show how this decision procedure can be used in connections with reasonable model and formula assumptions to decide the questions above. Section 6 contains a brief summary.

2 Multi-Lane Spatial Logic with Scope

In this section we introduce Multi-Lane Spatial Logic (MLSL) together with an extension called Multi-Lane Spatial Logic with Scope (MLSLS). In this extension it is possible to narrow the *scope* for the cars considered in a given traffic situation. MLSLS is a conservative extension of MLSL.

The definition of MLSLS is based on [4, 7]. It is simpler in the sense that we only consider spatial properties of static traffic configurations in this paper, and more complex because we introduce a scope component.

2.1 The Model

Only motorway traffic is considered here and a motorway is modelled as a two dimensional world; the vertical discrete dimension represents the different *lanes* and the horizontal dense dimension represents the *extension* of the lanes. A *traffic snapshot* contains for every car information about the current lane of the car, which we call *reservation* and the position along the lane. Usually, a car only has a reservation for one lane, but when it is changing lanes it has reservations on two adjacent lanes. Additionally, when a car would like to change to another lane it has a *claim* for that lane.

We assume a countably infinite set of *car identifiers* $\mathbb{I}$ and an arbitrary but fixed set of lanes $\mathbb{L} = \{0, \ldots, k\}$, for some $k \in \mathbb{N}_{\geq 1}$ to be given. Let $\mathcal{P}(\mathbb{L})$ denote the powerset of $\mathbb{L}$.

Definition 1 (Traffic snapshot [4, 7]). *A* traffic snapshot $\mathcal{TS}$ *is a structure* $\mathcal{TS} = (res, clm, pos)$*, where*

- $res : \mathbb{I} \to \mathcal{P}(\mathbb{L})$ *maps cars to their reserved lanes,*
- $clm : \mathbb{I} \to \mathcal{P}(\mathbb{L})$ *maps cars to their claimed lanes and*
- $pos : \mathbb{I} \to \mathbb{R}$ *maps cars to the position of their rear along the lanes.*

Furthermore, we require the following sanity conditions *to hold for all* $C \in \mathbb{I}$*.*

1. *Car* C *cannot both reserve and claim the same lane:* $res(C) \cap clm(C) = \emptyset$
2. *Car* C *can reserve at most two lanes:* $1 \leq |res(C)| \leq 2$
3. *Car* C *can claim at most one lane:* $0 \leq |clm(C)| \leq 1$
4. *Car* C *can reserve or claim at most two lanes:* $1 \leq |res(C)| + |clm(C)| \leq 2$

5. *A claimed lane must be next to a reserved lane for car C:*

$$clm(C) \neq \emptyset \text{ implies } \exists n \in \mathbb{L}.res(C) \cup clm(C) = \{n, n+1\}$$

6. *Only finitely may cars participate or initiate in lane changing manoeuvres:*

$$|res(C)| = 2 \text{ or } |clm(C)| = 1 \text{ holds only for finitely many } C \in \mathbb{I}$$

7. *As cars have a geoemtric extent, they have a minimum spacing of, say, 3.5 m. This applies on at least one of the lanes a car currently holds a reservation for*

$$\exists l \in res(C).\forall C' \in \mathbb{I} \setminus \{C\}. \begin{pmatrix} |res(C')| > 1 \vee l \notin res(C') \\ \vee\, |pos(C) - pos(C')| \geq 3.5 \end{pmatrix}$$

We denote the set of all traffic snapshots by $\mathbb{TS}$.

To address the safety of given traffic situation, a notion *safety envelope* of a car is introduced in [4] to capture the necessary space for a safe stop of the car. No car should interfere with the safety envelope of another car. The safety envelope of C in the traffic snapshot $\mathcal{TS}$ is:

$$se(C, \mathcal{TS}) = [pos(C), pos(C) + br\text{-}dis_C]\,,$$

where $br\text{-}dis_C$ is the current given breaking distance of C.

In MLSL, properties from the perspective of a specific car called *ego* are considered. The notion *view* captures this perspective, where a view has information about the lanes, their extension and the identity of *ego*. Intuitively, a view is a window through which *ego* perceives a traffic snapshot.

Definition 2 (View). *A* view *is a structure $V = (L, X, E)$, where*

- *$L = [l, n] \subseteq \mathbb{L}$ is an interval of lanes that are visible in the view,*
- *$X = [r, t] \subseteq \mathbb{R}$ is the extension of the lanes that is visible in the view and*
- *$E \in \mathbb{I}$ is the identifier of the car under consideration, that is,ego.*

A subview *of V is obtained by restricting the lanes and extension we observe. Let L', X' be subintervals of L and X, then we define*

$$V^{L'} = (L', X, E) \qquad \text{and} \qquad V_{X'} = (L, X', E).$$

If $l > n$ or $r = t$ we say that the view is empty.

Let *CVar* be a set of variables ranging over car identifiers. In the logic we use a special constant *ego* to refer to the owner of the current view. A valuation maps variables and *ego* to car identifiers, i.e. a valuation is a function $\nu : CVar \cup \{ego\} \to \mathbb{I}$. Additionally we define valuation updates with the override notation $\oplus$ from Z [15] as $\nu \oplus \{c \mapsto C\}(c') = C$ if $c = c'$ and $\nu(c')$ otherwise.

A view narrows down the spatial part of the motorway to a possibly restricted set of lanes with a possibly restricted extend. We introduce the notion *scope* to the model to be able to narrow down the considered cars in a given situation. This leads to the following definition of a *model with scope*.

Definition 3 (Model with Scope). *Let $CS \subseteq \mathbb{I}$ be a set of cars, $\mathcal{TS}$ be a traffic snapshot, V be a view and ν be a valuation. Then we call $\mathcal{M} = (CS, \mathcal{TS}, V, \nu)$ a model of MLSLS with scope CS.*

Notice that a model $\mathcal{M}$ with scope $\mathbb{I}$ is a model of MLSL in the sense of [4,7].

2.2 The Logic: MLSLS

MLSL is a multi-modal, first-order logic with interval-logic inspired modalities, that is, a *vertical chop* modality for partitioning a view into an upper and a lower subview, and a *horizontal chop* modality for partitioning a view into a left and a right subview. MLSLS extends MLSL with formulas of the form:

$$cs : \phi \,, \text{for } cs \subseteq CVar,$$

where the cars considered when determining the truth value of ϕ is narrowed down to cars denoted by variables in cs.

Definition 4 (Syntax). *The set of MLSLS formulas $\phi \in \Phi$ is generated by the grammar:*

$$\phi ::= \gamma = \gamma' \mid \mathit{free} \mid \mathit{re}(\gamma) \mid \mathit{cl}(\gamma) \mid \ell = s \mid \neg\phi \mid \phi \wedge \phi \mid \exists c.\phi \mid \phi \frown \phi \mid \begin{matrix}\phi\\ \phi\end{matrix} \mid cs : \phi \,,$$

where $c \in CVar$, $s \in \mathbb{R}$, $\gamma, \gamma' \in CVar \cup \{ego\}$ and $cs \subseteq CVar$.

The formula *free* is true for one-lane views containing no cars, $re(c)$ and $cl(c)$ are true for one-lane views that are fully covered by the safety envelope of a reservation or claim, respectively, by c. $\phi \frown \psi$ denotes horizontal partitioning of a view and $\begin{matrix}\phi\\ \psi\end{matrix}$ vertical chop of a view.

Let $\mathrm{freeVar}(\phi)$ denote the set of free variables occurring in a MLSLS formula ϕ. The definition of this function is standard for the first-order fragment, so we just give the parts for *ego*, chopped and scoped formulas:

$$\mathrm{freeVar}(ego) = \{ego\}$$
$$\mathrm{freeVar}(\phi \frown \psi) = \mathrm{freeVar}\begin{pmatrix}\phi\\ \psi\end{pmatrix} = \mathrm{freeVar}(\phi) \cup \mathrm{freeVar}(\psi)$$
$$\mathrm{freeVar}(xs : \phi) = xs \cup \mathrm{freeVar}(\phi)$$

Definition 5 (Semantics). *Let $c \in CVar$, $s \in \mathbb{R}$ and $\gamma, \gamma' \in CVar \cup \{ego\}$. Given a scope $CS \subseteq \mathbb{I}$, a traffic snapshot $\mathcal{TS}$, a view $V = ([l, n], [r, t], E)$ and a valuation ν with $\nu(ego) = E$ we define the* satisfaction *of a formula by a model $\mathcal{M} = (CS, \mathcal{TS}, V, \nu)$ as follows:*

$$\mathcal{M} \models \gamma = \gamma' \quad \Leftrightarrow \quad \nu(\gamma) = \nu(\gamma')$$
$$\mathcal{M} \models \mathit{free} \quad \Leftrightarrow \quad (l \notin \mathit{res}(C) \cup \mathit{clm}(C) \text{ or } \mathit{se}(C, \mathcal{TS}) \cap (r, t) = \emptyset)$$

$$
\begin{array}{lcl}
 & & \textit{for every } C \in CS,\ \textit{and } l = n \textit{ and } r < t \\
\mathcal{M} \models re(\gamma) & \Leftrightarrow & l \in res(\nu(\gamma))\ \textit{and}\ [r,t] \subseteq se(\nu(\gamma), \mathcal{TS})\ \textit{and}\ l = n\ \textit{and}\ r < t \\
\mathcal{M} \models cl(\gamma) & \Leftrightarrow & l \in clm(\nu(\gamma))\ \textit{and}\ [r,t] \subseteq se(\nu(\gamma), \mathcal{TS})\ \textit{and}\ l = n\ \textit{and}\ r < t \\
\mathcal{M} \models \ell = s & \Leftrightarrow & t - r = s \\
\mathcal{M} \models cs : \phi & \Leftrightarrow & (\{\nu(c) \mid c \in cs\}, \mathcal{TS}, V, \nu) \models \phi \\
\mathcal{M} \models \neg\phi & \Leftrightarrow & \mathcal{M} \not\models \phi \\
\mathcal{M} \models \phi_0 \wedge \phi_1 & \Leftrightarrow & \mathcal{M} \models \phi_0\ \textit{and}\ \mathcal{M} \models \phi_1 \\
\mathcal{M} \models \exists c.\phi & \Leftrightarrow & (CS, \mathcal{TS}, V, \nu \oplus \{c \mapsto C\}) \models \phi,\ \textit{for some } C\ \textit{in } CS \\
\mathcal{M} \models \phi_0 \frown \phi_1 & \Leftrightarrow & (CS, \mathcal{TS}, V_{[r,s]}, \nu \models \phi_0)\ \textit{and}\ (CS, \mathcal{TS}, V_{[s,t]}, \nu \models \phi_1, \\
 & & \textit{for some } s \textit{ where } r \leq s \leq t \\
\mathcal{M} \models \begin{smallmatrix}\phi_1\\ \phi_0\end{smallmatrix} & \Leftrightarrow & l \leq n\ \textit{implies} \\
 & & \quad (CS, \mathcal{TS}, V^{[l,m]}, \nu) \models \phi_0\ \textit{and}\ (CS, \mathcal{TS}, V^{[m+1,n]}, \nu) \models \phi_1) \\
 & & \quad \textit{for some } m, \textit{ where } l - 1 \leq m \leq n,\ \textit{and} \\
 & & l > n\ \textit{implies}\ ((CS, \mathcal{TS}, V, \nu) \models \phi_0\ \textit{and}\ (CS, \mathcal{TS}, V, \nu) \models \phi_1)
\end{array}
$$

In the semantics of the vertical chop operator $\begin{smallmatrix}\phi_1\\ \phi_0\end{smallmatrix}$ we deviate from the classical semantics and distinguish two cases. If the current view contains at least one lane we split the view into a lower and an upper subview and evaluate ϕ_0 on the lower subview and ϕ_1 on the upper subview. Otherwise, i.e. when the view is empty, we do not chop the view and instead evaluate both formulas on the same view. The intuition here is that all subviews of an empty view are empty and we can not distinguish different empty views with MLSLS. This special handling is necessary, because if we chop along a lane into a lower and an upper subview the lanes of the two subviews should be disjoint. However, for horizontal chops the endpoint of left subview and the startpoint of the right subview are shared.

The scope component CS of a model $(CS, \mathcal{TS}, V, \nu)$ is used in the semantics for the formulas *free* and $\exists c.\phi$. The formula *free* holds if no car from the scope CS occupies a part of the lane under consideration, and $\exists c.\phi$ holds if ϕ holds for some car C in the scope CS.

Definition 6 (Satisfiability and Validity).

- *An MLSLS formula ϕ is satisfiable iff for $(CS, \mathcal{TS}, V, \nu) \models \phi$, for some scope $CS \subseteq \mathbb{I}$, traffic snapshot $\mathcal{TS}$, view V and valuation ν.*
- *An MLSLS formula ϕ is valid iff $(\mathbb{I}, \mathcal{TS}, V, \nu) \models \phi$ for every traffic snapshot $\mathcal{TS}$, view V and valuation ν.*

If we disregard formulas of the form $cs : \phi$ and use $\mathbb{I}$ as scope component in models, then the above semantics coincides with that for MLSL.

We make use of the standard first order abbreviations such as $\mathit{true}, \mathit{false}, \vee, \forall$. In addition we define

$$\ell \geq r \equiv \ell = r \frown \mathit{true}$$

that is, the extension is longer than or equal to r. It is now easy to define $\ell < r, \ell \leq r$, and $\ell > r$.

To derive a similar constraint for the lane dimension: $\mathcal{L} = 1$, i.e. the number of lanes in the current view is one, we use the formula *free*:

$$\mathcal{L} = 1 \equiv \{\} : \mathit{free}$$

together with the empty scope. Inspecting the semantics we see that this formula is true for a model with view $([l, n], [r, t], E)$ iff $l = n$ and $r < t$, i.e. it is required that a lane has a positive extent. Further relations on the number of lanes can be derived using vertical chop, for example:

$$\mathcal{L} = 2 \equiv \begin{pmatrix} \mathcal{L} = 1 \\ \mathcal{L} = 1 \end{pmatrix}$$

The following formula expresses that two cars on the same lane should always be at least 4 m apart:

$$\neg \langle \exists c, d. \mathit{re}(c) \frown \ell < 4 \frown \mathit{re}(d) \rangle$$

3 Technical Observability and Stable Models

While MLSL reasons about a countable infinite set $\mathbb{I}$ of (unique) car identifiers, technical surveillance of the traffic situation by the ego car can in real-time and thus in situ only harvest information about a finite set $\mathbb{S} \subset \mathbb{I}$ of neighbouring cars. Evaluation of guard or invariant conditions employed in supervisory control can consequently only resort to state information concerning the perceived set $\mathbb{S}$ of cars. Let us assume that the particular sample $\mathbb{S} \subset \mathbb{I}$ drawn satisfies some reasonable constraints. Then this leads to the question whether

1. evaluating such a condition is independent from the particular sample $\mathbb{S} \subset \mathbb{I}$ drawn and
2. evaluating a condition on a sample $\mathbb{S} \subset \mathbb{I}$ provides reliable information on its validity over $\mathbb{I}$ itself, including the hidden states of cars in $\mathbb{I} \setminus \mathbb{S}$.

Note that the constraints do not determine a single sample, but rather exclude useless samples. In the rest of this section we first formalize these properties and then give an example.

We assume that $\mathbb{S}$ has a fixed maximal size $|\mathbb{S}| \leq N \in \mathbb{N}$ imposed by the real-time constraints on harvesting environmental information via own measurements by the ego car and via car2x communication. We furthermore assume that we know which sample sets $\mathbb{S}$ may arise in a given situation, which in turn is represented by an omniscient traffic snapshot $\mathcal{TS}$. That is, we assume a relation $\mathit{consistent} \subset \mathbb{TS} \times \mathcal{P}(\mathbb{I})$, where $\mathit{consistent}(\mathcal{TS}, \mathbb{S})$ captures the relation between overall traffic situations $\mathcal{TS}$ and samples $\mathbb{S} \subset \mathbb{I}$ that may arise due to technical surveillance within the particular situation. We use MLSLS to express the consistency relation.

Definition 7 (Consistency Constraint). *Let $\bar{c} \equiv ego, c_2, \ldots, c_N$ be a vector of car variables. A consistency constraint is a MLSLS formula consistent$(\bar{c})$ which has $c_2, \ldots, c_N$ has free variables. In any satisfying model of consistent$(\bar{c})$ the assignments to $\bar{c}$ constitute a consistent sample for the traffic snapshot.*

Note that a sample may contain less than N identifiers. In this case some of the variables from $\bar{c}$ are mapped to the same car. This consistency formula can be considered a requirements specification for the equipment sensing cars in the neighbourhood.

Now we can formalize properties 1. and 2. from above in MLSLS. A formula is *stable* iff on all models, the evaluation of the truth value does not depend on the particular consistent samples drawn. Further, a formula is *strongly stable* iff it always evaluates to the same value on a consistent sample and the omniscient traffic snapshot.

Definition 8 (Stability under sampling). *Let $\bar{c} \equiv ego, c_2, \ldots, c_N$ and $\bar{c}' \equiv ego, c'_2, \ldots, c'_N$ be two vectors of car variables, where $c_2, \ldots, c_N$ and $c'_2, \ldots, c'_N$ are mutually distinct and let consistent$(\bar{c})$ be a consistency constraint. Then a MLSLS formula ϕ is stable iff*

$$(\mathit{consistent}(\bar{c}) \wedge \mathit{consistent}(\bar{c}')) \implies (\{\bar{c}\}{:}\phi \iff \{\bar{c}'\}{:}\phi) \text{ is valid,} \quad (1)$$

and it is strongly stable iff

$$\mathit{consistent}(\bar{c}) \implies (\{\bar{c}\}{:}\phi \iff \phi) \text{ is valid.} \quad (2)$$

3.1 An Example of Stability under Sampling

In the following we give an example of a consistency constraint and a formula and show that the formula is stable under sampling. As abbreviations we introduce

$$\begin{aligned} \mathrm{notObs}(\bar{c}, c) &\equiv \textstyle\bigwedge_{c' \in \bar{c}} c \neq c' \wedge (\mathit{re}(c) \vee \mathit{cl}(c))\,, \\ \mathrm{someObs}(\bar{c}) &\equiv \textstyle\bigvee_{c' \in \bar{c}} (\mathit{re}(c') \vee \mathit{cl}(c'))\,, \end{aligned}$$

where $\mathrm{notObs}(\bar{c}, c)$ holds if the extension $[r, t]$ of the considered lane is covered by a reservation or claim of a car c that does not belong to the sample, and $\mathrm{someObs}(\bar{c})$ holds if the extension $[r, t]$ of the considered lane is covered by a reservation or claim of some sampled car (possibly *ego*). Consider the following consistency constraints:

1. Cars beyond a certain distance, e.g. 500 m or two lanes, are never observed, be it due to physical limits of sensors or to filtering mechanisms in car2x communication aiming at confining communication bandwidth.

$$\mathrm{Con}_1(\bar{c}) \equiv \neg \langle \mathit{re}(ego) \rangle \frown \ell \geq 500 \frown \langle \mathrm{someObs}(\bar{c}) \rangle \wedge \neg \begin{pmatrix} \langle \mathit{re}(ego) \rangle \\ \mathcal{L} \geq 2 \\ \langle \mathrm{someObs}(\bar{c}) \rangle \end{pmatrix}$$

2. Within a distance of 250 m from *ego*, it is not the case that a car on my own lane is not sampled while another car significantly further away from *ego* is sampled. By significantly further away we mean more than 5 m.

$$\mathrm{Con}_2(\bar{c}) \equiv \neg(\langle re(ego) \frown \ell \leq 250 \frown \exists c.\mathrm{notObs}(c)\rangle \frown \ell \geq 5 \frown \langle \mathrm{someObs}(\bar{c})\rangle)$$

3. All cars on neighboring lanes within 100 m of *ego* belong to the sample.

$$\mathrm{Con}_3(\bar{c}) \equiv \neg \left\langle \begin{pmatrix} \mathcal{L} = 1 \\ re(ego) \\ \mathcal{L} = 1 \end{pmatrix} \frown \ell \leq 100 \frown \langle \exists c.\mathrm{notObs}(\bar{c}, c)\rangle \right\rangle$$

These constraints are not exhaustive, as we have not considered, e.g. the area behind *ego* at all. We define the consistency constraint as

$$\mathrm{consistent}(\bar{c}) \equiv \bigwedge_{i=1}^{3} \mathrm{Con}_i(\bar{c}) \tag{3}$$

Consider next a guard on a transition of a control automaton that should be taken when an overtake manoeuvre is initiated. This guard should ensure that an overtake manoeuvre is meaningful and safe. It is meaningful when there is a car in front of ego (on the same lane) within 35 m, and it is safe when the lane left of *ego* is free for at least 100 m. This leads to the following specification of the guard:

$$\mathrm{guard} \equiv \neg \left\langle \begin{matrix} (free \wedge \ell \geq 100) \frown true \\ re(ego) \frown (free \wedge \ell < 35) \frown \exists c.re(c) \frown true \end{matrix} \right\rangle$$

This guard is evaluated on a given sample for a given traffic snapshot.

To check whether guard is stable for samples up to size $N = 20$ let $\bar{c} \equiv ego, c_2, \ldots, c_{20}, \bar{c}' \equiv ego, c'_2, \ldots, c'_{20}$ be two vectors of car variables. We instantiate the stability formula from Definition 8 as

$$(\mathrm{consistent}(\bar{c}) \wedge \mathrm{consistent}(\bar{c}')) \implies (\{\bar{c}\}\!:\!\mathrm{guard} \iff \{\bar{c}'\}\!:\!\mathrm{guard})$$

and check whether it is satisfied by all models of the form $(\mathbb{I}, \mathcal{TS}, V, \nu)$. To see that guard is stable w.r.t. the consistency constraint from Formula 3 observe that when $(\mathbb{I}, \mathcal{TS}, V, \nu) \models \mathrm{consistent}(\bar{c}) \wedge \mathrm{consistent}(\bar{c}')$ holds, then ν contains an assignments to $\bar{c}, \bar{c}'$, which induce two consistent samples $\mathbb{S}, \mathbb{S}'$. Further, we know that Con_3 ensures that $\mathbb{S}, \mathbb{S}'$ both contain all cars on adjacent lanes within 100 m in front of *ego*. As guard reasons about space at most 100 m from *ego*, we know that $(\mathbb{I}, \mathcal{TS}, V, \nu) \models \{\bar{c}\}\!:\!\mathrm{guard} \iff \{\bar{c}'\}\!:\!\mathrm{guard}$ is satisfied.

4 Satisfiability of MLSLS

In this section we give a decision procedure for deciding a subset of MLSLS. To do so we transform formulas from this subset to constraints belonging to

quantified linear integer-real arithmetic (QLIRA), for which the satisfiability problem is decidable [8, 14]. In the considered fragment, scoped formulas are used to enforce that there is a fixed bound on the number of cars that need consideration. In particular, it is required that the formulas *free* and $\exists c.\phi$ occur only inside a scoped formula. Such formulas are called well-scoped formulas.

Definition 9 (Well-scoped MLSLS formulas). *The set of well-scoped MLSLS formulas $\phi \in \Phi_W$ is generated by the following grammar:*

$$\begin{aligned}
\phi &::= A \mid \neg\phi \mid \phi \wedge \phi \mid \phi \frown \phi \mid \begin{matrix}\phi\\ \phi\end{matrix} \mid cs{:}\phi' \,, \\
A &::= \ell = s \mid \gamma = \gamma' \mid re(\gamma) \mid cl(\gamma) \\
\phi' &::= \mathit{free} \mid \exists c.\phi' \mid A \mid \neg\phi' \mid \phi' \wedge \phi' \mid \phi' \frown \phi' \mid \begin{matrix}\phi'\\ \phi'\end{matrix} \mid cs{:}\phi' \,,
\end{aligned}$$

where $c \in CVar$, $cs \subseteq CVar$, $s \in \mathbb{R}$ and $\gamma, \gamma' \in CVar \cup \{ego\}$.

In QLIRA we use variables ranging over the real numbers, linear arithmetic and rounding to the next smaller integer.

Definition 10 (Formulas of QLIRA). *The set of QLIRA formulas $\psi \in \Psi$ is generated by the following grammar:*

$$\begin{aligned}
\psi &::= \exists x \in \mathbb{R}.\psi \mid term \leq term \mid \neg\psi \mid \psi \wedge \psi \,, \\
term &::= s \mid x \mid \lfloor x \rfloor \mid term + term \,,
\end{aligned}$$

where RVar us a set of variables and $s \in \mathbb{R}$. With NVar we denote the set of variables where each element $x \in NVar$ has the constraint $x \leq \lfloor x \rfloor \wedge 0 \leq x$.

We use the remaining propositional connectives and $=, <, \geq$ and $>$ as abbreviations. Furthermore, $\exists i \in \mathbb{N}.\psi$ is an abbreviation for $\exists i \in \mathbb{R}.i \leq \lfloor i \rfloor \wedge 0 \leq i \wedge \psi$.

For terms $term_j, j \in [0, 3]$, and terms $\{term_0, \ldots, term_k\}$, $k \in \mathbb{N}_{\geq 1}$, we define

$$\begin{aligned}
[term_0, term_1] \subseteq [term_2, term_3] &\equiv term_2 \leq term_0 \wedge term_1 \leq term_3 \,, \\
term_0 \in \{term_1, \ldots, term_k\} &\equiv \bigvee_{term_j \in \{term_1, \ldots, term_k\}} term_j = term_0 \,, \\
[term_0, term_1] \cap (term_2, term_3) = \emptyset &\equiv term_1 \leq term_2 \vee term_3 \leq term_0 \,.
\end{aligned}$$

Note that both intervals in the subset definition are closed, but for the intersection definition the interval $(term_2, term_3)$ is open.

4.1 A QLIRA representation of a traffic snapshot $\mathcal{TS}$

It is now described how the satisfiability problem for well-scoped formulas ϕ is reduced to satisfiability of QLIRA formulas. Variables of QLIRA are introduced so that the various components of a model $(CS, \mathcal{TS}, V, \nu)$, for $V = ([l, n], [r, t], E)$, can be represented in QLIRA, and so that the translation function "mimics" the

definition of the semantics relation $\models$ in Definition 5. A key issue is the QLIRA representation of a traffic snapshot.

Let ϕ be a well-scoped formula with free variables $cs = \{c_0, c_1, \ldots, c_{n-1}\} = \text{freeVar}(\phi)$. Then, due to the structure of well-scoped formulas, the number of free variables n is a bound on the number of cars necessary to consider when checking for the satisfiability of ϕ. Hence, only a finite traffic snapshot needs to be represented when checking for satisfiability. To do so, we introduce n natural number variables of QLIRA $C^0, C^1, \ldots, C^{n-1}$ representing n cars. Furthermore, let $f_{init} : cs \to NVar$ be defined by $f_{init}(c_i) = C^i$.

The spatial information for each of these cars, say $C^i \in NVar$[3], is represented by five QLIRA variables: C^i_{pos}, C^i_{res}, $C^i_{res'}$, C^i_{clm}, and $C^i_{br\text{-}dis}$, for the position, lane reserved, alternative lane reserved, lane claimed, and size of the safety envelope. Hence, prior to the translation of ϕ a table is created containing n entries $(C^i, (C^i_{pos}, C^i_{res}, C^i_{res'}, C^i_{clm}, C^i_{br\text{-}dis}))$ with QLIRA variables for n cars.

Variables C^i_{pos} and $C^i_{br\text{-}dis}$ range over the reals and variables C^i_{res} range over natural numbers. Variables $C^i_{res'}, C^i_{clm}$ range over natural numbers denoting a lane or may take a special value, say $\text{nil} = -2$ denoting no reservation or no claim. Technically this is enforced by associating a constraint of the form $(x \leq \lfloor x \rfloor \wedge 0 \leq x) \vee x = -2$ with each such variable.

To meaningfully represent a traffic snapshot, these variables must satisfy properties such as $C^i_{br\text{-}dis} > 0$ and if two distinct variables C^i and C^j denotes the same car, then the characterizing variables for C^i and C^j must agree. Such properties can be formulated in QLIRA, for example:

$$C^i = C^j \implies \begin{pmatrix} C^i_{pos} = C^j_{pos} \wedge C^i_{br\text{-}dis} = C^j_{br\text{-}dis} \\ \wedge\ C^i_{res} = C^j_{res} \wedge C^i_{res'} = C^j_{res'} \wedge C^i_{clm} = C^j_{clm} \\ \wedge\ (C^i_{res'} = \text{nil} \vee C^i_{clm} = \text{nil}) \end{pmatrix}$$

Notice that also the sanity constraints of Definition 1 can be expressed in QLIRA using the introduced variables. We will not give further details here but just assume the existence of a QLIRA formula, named "sanity", capturing the sanity constraints for the QLIRA representation (like the formula above) as well as the sanity constraints for traffic snapshot.

4.2 Translating well-scoped formulas to QLIRA

The translation function from well-scoped formulas to QLIRA should "mimic" the definition of the semantic relation $\models$ in Definition 5. Inspecting this semantics, it is observed that the traffic snapshot $\mathcal{TS}$ and the ego part E of a view $V = ([l, n], [r, t], E)$ remain constants throughout the recursive definition of $(CS, \mathcal{TS}, V, \nu) \models \phi$.

Hence, the translation function must keep track of the

- scope part CS, i.e. a subset of $\{C^0, C^1, \ldots, C^{n-1}\}$,

[3] Please note that C in some context denotes a car identifier and in another context a QLIRA variable

- the lane part $[l, n]$ of V, i.e. two natural number variables of QLIRA,
- the extent part $[r, t]$ of V, i.e. two real number variables of QLIRA, and
- the valuation part ν, i.e. a function with type: $f : cs \to \{C^0, C^1, \ldots, C^{n-1}\}$.

The part that many change during translation is modelled by the type T:

$$\mathrm{T} = \mathcal{P}(NVar) \times NVar \times NVar \times RVar \times RVar \times ((CVar \cup \{ego\}) \to NVar)$$

Definition 11 (Transformation). *The transformation is given by a function*

$$tr : \mathcal{T} \times \Phi_W \to \Psi\,.$$

Let $\Upsilon = (CS, i, i', x, x', f) \in \mathrm{T}$. *Then the transformation is given as:*

$$\begin{aligned}
tr(\Upsilon, re(\gamma)) :=\ & x' > x \wedge [x, x'] \subseteq [C_{pos}, C_{pos} + C_{br\text{-}dis}] \wedge \\
& i = i' \wedge i = C_{res} \vee i = C_{res'},\ \textit{where } C = f(\gamma) \\
tr(\Upsilon, cl(\gamma)) :=\ & x' > x \wedge [x, x'] \subseteq [C_{pos}, C_{pos} + C_{br\text{-}dis}] \wedge \\
& i = i' \wedge i = C_{clm},\ \textit{where } C = f(\gamma) \\
tr(\Upsilon, free) :=\ & i = i' \wedge x' > x \wedge \bigwedge_{C \in CS} \big(i \notin \{C_{res}, C_{res'}, C_{clm}\} \vee \\
& [x, x'] \cap [C_{pos}, C_{pos} + C_{br\text{-}dis}] = \emptyset\big) \\
tr(\Upsilon, \gamma = \gamma') :=\ & f(\gamma) = f(\gamma') \\
tr(\Upsilon, cs : \phi) :=\ & tr((\{f(c) \mid c \in cs\}, i, i', x, x', f), \phi) \\
tr(\Upsilon, \phi_0 \wedge \phi_1) :=\ & tr(\Upsilon, \phi_0) \wedge tr(\Upsilon, \phi_1) \\
tr(\Upsilon, \neg\phi) :=\ & \neg tr(\Upsilon, \phi) \\
tr(\Upsilon, \exists c.\phi) :=\ & \bigvee_{C \in CS} tr((CS, i, i', x, x', f \oplus \{c \mapsto C\}), \phi) \\
tr(\Upsilon, \phi_0 \frown \phi_1) :=\ & \exists x'' \in \mathbb{R}. x \le x'' \le x' \wedge \\
& \qquad tr((CS, i, i', x, x'', f), \phi_0) \wedge tr((CS, i, i', x'', x', f), \phi_1)
\end{aligned}$$

$$tr(\Upsilon, \begin{smallmatrix}\phi_1\\ \phi_0\end{smallmatrix}) := \left(\begin{array}{l}
i \le i' \implies \exists i'' \in \mathbb{N}. \left(\begin{array}{r} i - 1 \le i'' \le i' \\ \wedge\ tr((CS, i, i'', x, x', f), \phi_0) \\ \wedge\ tr((CS, i'' + 1, i', x, x', f), \phi_1) \end{array}\right) \\
\wedge \\
i > i' \implies \left(\begin{array}{r} tr((CS, i, i', x, x', f), \phi_0) \\ \wedge\ tr((CS, i, i', x, x', f), \phi_1) \end{array}\right)
\end{array}\right)$$

Notice that the translation function is a direct reflection of the definition of semantic relation $\models$.

We define

$$F(\phi) \equiv \text{sanity} \wedge tr((\emptyset, i, i', x, x', f_{init}), \phi)\,,$$

where the formula "sanity" is the QLIRA formula mentioned above that expresses the sanity constraints on traffic snapshots and sanity constraints on the encoding. The QLIRA constraint $F(\phi)$ is equisatisfiable to the well-scoped MLSLS formula ϕ, which is stated in the following theorem.

Theorem 1. *Given a well-scoped MLSLS formula ϕ we can create QLIRA constraints $F(\phi)$ such that*

$$\phi \text{ is satisfiable} \iff F(\phi) \text{ is satisfiable}.$$

A direct consequence is decidability of well-scoped MLSLS:

Corollary 1 (Decidability of well-scoped MLSLS). *Satsfiability and validity of well-scoped MLSLS are decidable.*

Proof. Well-scoped MLSLS is closed under negation. Hence, both satisfiability and validity can be reduced to satisfiability problems. Given a well-scoped MLSLS formula ϕ, Theorem 1 permits generating a QLIRA constraint $F(\phi)$ which is equisatisfiable to ϕ. Satisfiabilty of $F(\phi)$, and thus equivalently satisfiability of ϕ, is decidable due to decidability of QLIRA. □

5 Deciding Stability

We will now turn to the problem of deciding whether an MLSLS formula ϕ is stable, i.e. whether

$$(\text{consistent}(\bar{c}) \wedge \text{consistent}(\bar{c'})) \implies (\{\bar{c}\}{:}\phi \iff \{\bar{c'}\}{:}\phi) \tag{4}$$

holds.

Observing that the satisfiablity problem of scoped MLSLS is decidable, we address the above problem by adequately closing formula (4) by successive introduction of scope operators. To this end, we note two properties of the scope operator which permit its introduction:

Lemma 1 (Scope introduction). *Let ϕ be an MLSLS formula containing a positive (or negative, respectively) occurrence of some subformula $\psi = \exists c.\xi$. Furthermore assume that ψ does occur outside any scope operator in ϕ. Let ϕ' be the formula that is obtained by replacing ψ in ϕ by $\{d_1, \ldots, d_n\} : \exists c.\xi$, where $d_1, \ldots, d_n$ are arbitrary variable names. Then validity of ϕ' is a sufficient (necessary, resp.) condition for validity of ϕ.*

Proof. The freshly introduced scope operator confines the range of the existential quantifier to a subrange of the car identifiers, which strengthens or weakens, respectively, the overall formula depending on polarity of the quantifier occurrence. □

Due to decidability of scoped MLSLS (Corollary 1), we thus obtain a safe, yet incomplete method for checking stability under sampling: in our exemplary consistency predicates Con_1 to Con_3, we do only encounter universal statements

over objects outside the sample[4], which we consider to be the general form.[5] For such consistency formulas, we can proceed as follows:

1. Build formula (4). Within this formula, existential quantifiers outside scope operators do only occur positively.
2. Scope these quantifiers by arbitrarily large scopes according to Lemma 1.
3. Decide validity of the resulting scoped formula using the procedure of Corollary 1.
4. If the formula is valid then report "stable" and stop. This is sound as validity of the scoped formula is a sufficient condition for validity of the original criterion (4) according to Lemma 1.
5. Else go back to step 2 and repeat with larger scopes.

In most cases, we can however do better: the range of perception tends to be bounded. This means that there is a fixed range around the ego car outside which we do not expect cars to show up in the sample. In such cases, a lossless closure of the consistency predicate by introduction of (generally very large) scope operators is possible, which in turn will be exploited for deciding whether the actual (generally much smaller) sample still is large enough.

To get there, we first note some semantic properties of scope operators characterizing situations where introduction of scope operators does not affect satisfaction.

Lemma 2. *Let $\mathcal{TS}$ be a traffic snapshot, $V = ([l,n],[r,t],E)$ be a view and ν be a valuation, and denote by*

$$I_V = \{C \mid C \in \mathbb{I} \wedge [r,t] \cap se(C,\mathcal{TS}) \neq \emptyset \wedge [l,n] \cap (res(C) \cup clm(C)) \neq \emptyset\}$$

the set of cars visible in the view. Let ϕ be an MLSLS formula. Then

$$(\mathbb{I},\mathcal{TS},V,\nu) \models \phi \iff \exists CS = I_V \uplus S \subset \mathbb{I}.|S| = m \wedge (CS,\mathcal{TS},V,\nu) \models \phi\,,$$

where m is the number of quantifiers in ϕ and $\uplus$ denotes disjoint union. That is ϕ holds over all cars iff there is a sample containing all cars in the view plus as many extra cars as there are quantifiers which satisfies ϕ.

Proof. By induction on the structure of ϕ. The only atomic formula that is influenced by scoping is *free*. It is easy to check that the theorem holds when ϕ is *free*. The only interesting case left is quantification, as the semantics of the remaining constructs is not influenced by scoping. Therefore assume in the remainder that $\phi = \exists c.\psi$.

[4] The only quantifiers that are not over samples are existential quantifiers in negative context.

[5] To this end please note that there is no need to express that a sampled object actually exists in the outside world, as this has been built into the semantics. Existential statements about objects outside the sample therefore seem to be of no practical value.

To show the implication from left to right, assume that $(\mathbb{I}, \mathcal{TS}, V, \nu) \models \exists c.\psi$. Then there exists $C \in \mathbb{I}$ such that $(\mathbb{I}, \mathcal{TS}, V, \nu \oplus \{c \to C\}) \models \psi$. By induction hypothesis, as ψ has one quantifier less, there is $CS' = I_V \uplus S' \subset \mathbb{I}$ with $|S'| = m-1$ such that $(CS', \mathcal{TS}, V, \nu \oplus \{c \to C\}) \models \psi$. If $C \in CS'$ then $C \in CS' \cup \{D\}$ for an arbitrary $D \in \mathbb{I} \setminus CS'$ and consequently $(CS' \cup \{D\}, \mathcal{TS}, V, \nu) \models \exists c.\psi$. If $C \notin CS'$ then especially $C \notin I_V$ and consequently all atomic predicates except equations between car identifiers evaluate to false when c is bound to C. For the equations observe that C is as distinct from all the other car identifiers in $\mathbb{I}$ as it is from all the identifiers in CS'. Consequently $(CS' \cup \{C\}, \mathcal{TS}, V, \nu) \models \exists c.\psi$ holds again.

Let us now assume that $(\mathbb{I}, \mathcal{TS}, V, \nu) \not\models \exists c.\psi$. Then for all $C \in \mathbb{I}$ we have $(\mathbb{I}, \mathcal{TS}, V, \nu \oplus \{c \to C\}) \not\models \psi$. By induction hypothesis, as ψ has one quantifier less, $(CS', \mathcal{TS}, V, \nu \oplus \{c \to C\}) \not\models \psi$ holds for all $CS' = I_V \uplus S' \subset \mathbb{I}$ with $|S'| = m-1$. As this does in particular apply for all such CS' with $C \notin CS'$, we can conclude $(CS, \mathcal{TS}, V, \nu) \not\models \exists c.\psi$ for all $CS = I_V \uplus S$ with $|S| = m$.

Consequently the bi-implication holds. □

Assumption 1. *We assume that a part of a motorway of length s with m lanes can contain at most n different cars, where n depends on s and m.*

A direct consequence of this assumption is the following.

Corollary 2. *Let $\mathcal{TS}$ be a traffic snapshot, V be a view, ν be a valuation and n be the maximal number of cars fitting into the view V. Let ϕ be an MLSLS formula. Then $(\mathbb{I}, \mathcal{TS}, V, \nu) \models \phi$ iff there is a valuation ν' extending ν such that $(\mathbb{I}, \mathcal{TS}, V, \nu') \models \{c_1, \ldots, c_n, d_1, \ldots, d_m\}{:}\phi$, where m is the number of quantifiers in ϕ.*

We say that a quantifier (either from $\exists c.\phi$ or *free*) is unscoped if it ranges over the full range of car identifiers rather than just a finite sample. The crucial point of the previous corollary is to evaluate unscoped quantifiers only on a fixed area. We fix this area to *ego* and then we can introduce scopes to a large subset of MLSLS without losing completeness for this subset.

Corollary 3 (exact scope introduction). *Let ϕ be a MLSLS formula where each unscoped quantifier occurs within a positively occurring context of the form: $(\eta \wedge \ell \sim_1 k_1 \wedge \mathcal{L} \sim_2 k_2) \odot \psi$, where $\eta \in \{re(ego), cl(ego), \langle re(ego)\rangle, \langle cl(ego)\rangle\}$, k_i is constant and $\sim_i \in \{<, \leq, =\}$, for $i \in \{1, 2\}$, and $\odot \in \{\implies, \wedge\}$. Let K_1 and K_2 be the largest such constants occurring in the formula. Then*

$$\phi \text{ is equisatisfiable to } \{c_1, \ldots, c_n, d_1, \ldots, d_m\}{:}\phi$$

where n is the maximum number of cars fitting into a view V of length $2K_1$ and width $2K_2 - 1$ according to Assumption 1 and m is the number of quantifiers in ϕ.

Proof. Note that all subviews satisfying the "guarding" conditions $(\eta \wedge \ell \sim_1 k_1 \wedge \mathcal{L} \sim_2 k_2)$ have to contain the ego car due to η and are thus within a range of $[-K_1, K_1]$ around the ego car position and within the range of $[-K_2+1, K_2-1]$ around the ego car lane. The statement then follows from the previous. □

Note that this corollary permits to convert a partially scoped formula into an equisatisfiable fully scoped formula. Due to decidability of scoped MLSLS, this gives rise to the following decidability result.

Theorem 2 (Decidability of stability). *Let ϕ be an arbitrary MLSLS formula and let* consistent$(\bar{c})$ *be a consistency predicate defined in MLSLS expressing when a sample $\bar{c}$ is consistent. Further, let* consistent$(\bar{c})$ *be a MLSLS formula where each unscoped quantifier occurs within a positively occurring context of the form: $(\eta \wedge \ell \sim_1 k_1 \wedge \mathcal{L} \sim_2 k_2) \odot \psi$, where $\eta \in \{re(ego), cl(ego), \langle re(ego)\rangle, \langle cl(ego)\rangle\}$, k_i is constant and $\sim_i \in \{<, \leq, =\}$, for $i \in \{1, 2\}$, and $\odot \in \{\implies, \wedge\}$. Then it is decidable whether ϕ is stable under sampling with the consistency predicate* consistent$(\bar{c})$.

Proof. Stability under sampling is logically characterized by validity of formula (4). Under the preconditions of the theorem, the negation of formula (4) can be rewritten to an equisatisfiable MLSLS formula in scoped form according to Corollary 3. As satisfiablity of scoped MLSLS formulas is decidable due to Corollary 1, the claim follows. □

It should be noted that the exemplary consistency conditions from Section 3 fall into the above fragment of MLSLS such that stability under sampling can be decided for various sample sizes.

6 Conclusion

Considering Multi-Lane Spatial Logic (MLSL) in the context of supervisory control schemes for highly automated driving, we consider issues arising when formulas are checked on the basis of finite samples obtained in a given traffic situation rather than on the basis of complete information.

To study this in a formal framework, a conservative extension of MLSL is defined, called Multi-Lane Spatial Logic with Scope (MLSLS). In this extended logic one can, by use of a special *scope formula* $\{c_1, \ldots c_n\} : \phi$, narrow down the cars considered when determining the truth of ϕ to those denoted by $\{c_1, \ldots c_n\}$. Questions, such as

- Is the evaluation of a guard (a MLSLS formula) independent from the particular sample drawn (which may vary within reasonable bounds)?

can be formalized in MLSLS.

To tackle such questions, a decision procedure is presented for the satisfiability of a fragment of MLSLS. Furthermore, formal techniques are developed on top of this decision procedure allowing questions like the above on to be decided under reasonable assumptions on traffic models and the structure of formulas used as guards in control automata.

References

1. Allen, J.F.: Maintaining knowledge about temporal intervals. Communications of the ACM 26(11), 832–843 (1983)
2. Chaochen, Z., Hoare, C.A.R., Ravn, A.P.: A calculus of durations. Information processing letters 40(5), 269–276 (1991)
3. Halpern, J.Y., Shoham, Y.: A propositional modal logic of time intervals. Journal of the ACM (JACM) 38(4), 935–962 (1991)
4. Hilscher, M., Linker, S., Olderog, E.R., Ravn, A.P.: An Abstract Model for Proving Safety of Multi-Lane Traffic Manoeuvres. In: Qin, S., Qiu, Z. (eds.) ICFEM. LNCS, vol. 6991, pp. 404–419. Springer (2011)
5. Hilscher, M., Linker, S., Olderog, E.: Proving safety of traffic manoeuvres on country roads. In: Theories of Programming and Formal Methods, LNCS, vol. 8051, pp. 96–212. Springer (2013)
6. Linker, S.: Proofs for traffic safety : combining diagrams and logic. Ph.D. thesis, Carl von Ossietzky University of Oldenburg (2015)
7. Linker, S., Hilscher, M.: Proof theory of a multi-lane spatial logic. In: Theoretical Aspects of Computing–ICTAC 2013. pp. 231–248. Springer (2013)
8. Monniaux, D.: A Quantifier Elimination Algorithm for Linear Real Arithmetic. In: Cervesato, I., Veith, H., Voronkov, A. (eds.) Logic for Programming, Artificial Intelligence, and Reasoning, LNCS, vol. 5330, pp. 243–257. Springer (2008), `http://dx.doi.org/10.1007/978-3-540-89439-1_18`
9. Moszkowski, B.: A temporal logic for multi-level reasoning about hardware. IEEE Computer 18(2), 10–19 (1985)
10. Ody, H.: Analysing decision problems of multi-lane spatial logic (2015), manuscript
11. Olderog, E.R., Ravn, A.P., Wisniewski, R.: Linking Discrete and Continuous Models (2014), manuscript
12. Schäfer, A.: A Calculus for Shapes in Time and Space. In: Liu, Z., Araki, K. (eds.) Theoretical Aspects of Computing - ICTAC 2004, LNCS, vol. 3407, pp. 463–477. Springer (2005), `http://dx.doi.org/10.1007/978-3-540-31862-0_33`
13. Venema, Y.: A modal logic for chopping intervals. Journal of Logic and Computation 1(4), 453–476 (1991)
14. Weispfenning, V.: Mixed real-integer linear quantifier elimination. In: Proceedings of the 1999 international symposium on Symbolic and algebraic computation. pp. 129–136. ACM (1999)
15. Woodcock, J., Davies, J.: Using Z – Specification, Refinement, and Proof. Prentice Hall (1996)