Proof Theory of a Multi-Lane Spatial Logic*

Sven Linker, Martin Hilscher

Department of Computing Science, University of Oldenburg, Germany
{linker, hilscher}@informatik.uni-oldenburg.de

Abstract. We extend the Multi-lane Spatial Logic MLSL, introduced in previous work for proving the safety (collision freedom) of traffic maneuvers on a multi-lane highway, by length measurement and dynamic modalities. We investigate the proof theory of this extension, called EMLSL. To this end, we prove the undecidability of EMLSL but nevertheless present a sound proof system which allows for reasoning about the safety of traffic situations. We illustrate the latter by giving a formal proof for a lemma we could only prove informally before.

Keywords. Spatial logic, undecidability, labelled natural deduction.

1 Introduction

In our previous work [1] we proposed a multi-dimensional spatial logic MLSL inspired by Moszkowski's interval temporal logic (ITL) [2], Zhou, Hoare and Ravn's Duration Calculus (DC) [3] and Schäfer's Shape Calculus [4] for formulating the purely spatial aspects of safety of traffic maneuvers on highways. In MLSL we modeled the highway as one continuous dimension, i.e., in the direction along the lanes and one discrete dimension, the different lanes. We illustrated MLSL's usefulness by proving safety of two variants of lane change maneuvers on highways. The safety proof establishes that the braking distances of no two cars intersecting is an inductive invariant of a transition system capturing the dynamics of cars and controllers.

In this paper we introduce EMLSL which extends MLSL by length measurement and dynamic modalities. In comparison to MLSL, where we are only able to reason about qualitative spatial properties, i.e., topological relations between cars, EMLSL also allows for quantitative reasoning, e.g., on braking distances. To further the practicality of EMLSL, we define a proof system based on ideas of Basin et al. [5], who presented systems of labelled natural deduction for a vast class of typical modal logics. Rasmussen [6] refined their work to interval logics with binary chopping modalities. Since EMLSL incorporates both unary as well as chopping modalities, our proof system is strongly related to both approaches.

Besides providing a higher expressiveness, extending MLSL enables us to formulate and prove the invariance of the spatial safety property *inside* EMLSL and its deductive proof system. We demonstrate this by conducting a formal

* This research was partially supported by the German Research Council (DFG) in the Transregional Collaborative Research Center SFB/TR 14 AVACS.

proof of the so called *reservation lemma* [1], which informally states that no car changes lanes without having set the turn signal beforehand.

Further on, we show undecidability of a subset of EMLSL. We adapt the proof of Zhou et al. [7] for DC and reduce the halting problem of two counter machines to satisfiability of EMLSL formulas. Due to the restricted set of predicates EMLSL provides, this is non-trivial.

The *contributions* of this paper are as follows:

- we extend MLSL with lengths measurements and dynamic modalities (Sec. 2);
- we show the spatial fragment of EMLSL to be undecidable (Sec. 3);
- we present a suited proof system and derive the reservation lemma (Sec. 4).

2 Extended MLSL Syntax and Semantics

The purpose of EMLSL is to reason about highway situations. To this end, we first present the formal model of a *traffic snapshot* capturing the position and speed of every car on the highway at a given point in time. In addition a traffic snapshot comprises the lane a given car is driving on, which we call a *reservation*. Every car usually holds one reservation, i.e., drives on one lane, but may, during lane change maneuvers, hold up to two reservations on adjacent lanes. Furthermore, we capture the indication that a given car wants to change to a adjacent lane by the notion of a *claim* which is an abstraction of setting the turn signal. Every car may only hold claims while not engaged in a lane change.

To formally define a traffic snapshot, we assume a countably infinite set of globally unique *car identifiers* $\mathbb{I}$ and an arbitrary but fixed set of lanes $\mathbb{L} = \{0, \ldots, N\}$, for some $N \geq 1$. Throughout this paper we will furthermore make use of the notation $\mathcal{P}(X)$ for the powerset of X, and the override notation $\oplus$ from Z for function updates [8], i.e., $f \oplus \{x \mapsto y\}(z) = y$ if $x = z$ and $f(z)$ otherwise.

Definition 1 (Traffic snapshot). *A* traffic snapshot $\mathcal{TS}$ *is a structure* $\mathcal{TS} = (res, clm, pos, spd, acc)$, *where* res, clm, pos, spd, acc *are functions*

- $res : \mathbb{I} \to \mathcal{P}(\mathbb{L})$ *such that* $res(C)$ *is the set of lanes the car* C *reserves,*
- $clm : \mathbb{I} \to \mathcal{P}(\mathbb{L})$ *such that* $clm(C)$ *is the set of lanes the car* C *claims,*
- $pos : \mathbb{I} \to \mathbb{R}$ *such that* $pos(C)$ *is the position of the car* C *along the lanes,*
- $spd : \mathbb{I} \to \mathbb{R}$ *such that* $spd(C)$ *is the current speed of the car* C*,*
- $acc : \mathbb{I} \to \mathbb{R}$ *such that* $acc(C)$ *is the current acceleration of the car* C*.*

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

1. $res(C) \cap clm(C) = \emptyset$
2. $1 \leq |res(C)| \leq 2$
3. $0 \leq |clm(C)| \leq 1$
4. $1 \leq |res(C)| + |clm(C)| \leq 2$
5. $clm(C) \neq \emptyset$ *implies* $\exists n \in \mathbb{L} \bullet res(C) \cup clm(C) = \{n, n+1\}$
6. $|res(C)| = 2$ *or* $|clm(C)| = 1$ *holds only for finitely many* $C \in \mathbb{I}$*.*

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

The kinds of transitions are twofold. First, we have discrete transitions defining the possibilities to create, mutate and remove claims and reservations. The other type of transitions handles abstractions of the dynamics of cars, i.e., they allow for instantaneous changes of accelerations and for the passing of time, during which the cars move according to a simple model of motion. For the results presented subsequently, we only require the changes of positions to be continuous.

Definition 2 (Transitions). *The following* transitions *describe the changes that may occur at a traffic snapshot* $\mathcal{TS} = (res, clm, pos, spd, acc)$.

$$\begin{aligned}
\mathcal{TS} \xrightarrow{\mathsf{c}(C,n)} \mathcal{TS}' \quad \Leftrightarrow \quad & \mathcal{TS}' = (res, clm', pos, spd, acc) \\
& \wedge\, |clm(C)| = 0 \wedge\ |res(C)| = 1 \\
& \wedge\, res(C) \cap \{n+1, n-1\} \neq \emptyset \\
& \wedge\, clm' = clm \oplus \{C \mapsto \{n\}\} && (1)
\end{aligned}$$

$$\begin{aligned}
\mathcal{TS} \xrightarrow{\mathsf{wd}\ \mathsf{c}(C)} \mathcal{TS}' \quad \Leftrightarrow \quad & \mathcal{TS}' = (res, clm', pos, spd, acc) \\
& \wedge\, clm' = clm \oplus \{C \mapsto \emptyset\} && (2)
\end{aligned}$$

$$\begin{aligned}
\mathcal{TS} \xrightarrow{\mathsf{r}(C)} \mathcal{TS}' \quad \Leftrightarrow \quad & \mathcal{TS}' = (res', clm', pos, spd, acc) \\
& \wedge\, clm' = clm \oplus \{C \mapsto \emptyset\} \\
& \wedge\, res' = res \oplus \{C \mapsto res(C) \cup clm(C)\} && (3)
\end{aligned}$$

$$\begin{aligned}
\mathcal{TS} \xrightarrow{\mathsf{wd}\ \mathsf{r}(C,n)} \mathcal{TS}' \quad \Leftrightarrow \quad & \mathcal{TS}' = (res', clm, pos, spd, acc) \\
& \wedge\, res' = res \oplus \{C \mapsto \{n\}\} \\
& \wedge\, n \in res(C) \wedge\ |res(C)| = 2 && (4)
\end{aligned}$$

$$\begin{aligned}
\mathcal{TS} \xrightarrow{t} \mathcal{TS}' \quad \Leftrightarrow \quad & \mathcal{TS}' = (res, clm, pos', spd', acc) \\
& \wedge\, \forall C \in \mathbb{I}\colon\ pos'(C) = pos(C) + spd(C) \cdot t + \tfrac{1}{2} acc(C) \cdot t^2 \\
& \wedge\, \forall C \in \mathbb{I}\colon\ spd'(C) = spd(C) + acc(C) \cdot t && (5)
\end{aligned}$$

$$\begin{aligned}
\mathcal{TS} \xrightarrow{\mathsf{acc}(C,a)} \mathcal{TS}' \quad \Leftrightarrow \quad & \mathcal{TS}' = (res, clm, pos, spd, acc') \\
& \wedge\, acc' = acc \oplus \{C \mapsto a\} && (6)
\end{aligned}$$

We also combine passing of time and changes of accelerations to *evolutions*.

$$\mathcal{TS} \overset{t}{\Rightarrow} \mathcal{TS}' \Leftrightarrow \mathcal{TS} = \mathcal{TS}_0 \xrightarrow{t_0} \mathcal{TS}_1 \xrightarrow{\mathsf{acc}(C_0,a_0)} \dots \xrightarrow{t_n} \mathcal{TS}_{2n-1} \xrightarrow{\mathsf{acc}(C_n,a_n)} \mathcal{TS}_{2n} = \mathcal{TS}',$$

where $t = \sum_{i=0}^{n} t_i$, $a_i \in \mathbb{R}$ and $C_i \in \mathbb{I}$ for all $0 \leq i \leq n$.

The transitions preserve the sanity conditions in Def. 1.

Lemma 1 (Preservation of Sanity). *Let* $\mathcal{TS}$ *be a snapshot satisfying the constraints given in Def. 1. Then, each structure* $\mathcal{TS}'$ *reachable by a transition is again a traffic snapshot satisfying Def. 1.*

EMLSL restricts the parts of the motorway perceived by each car to so called *views*. Each view comprises a set of lanes and a real-valued interval, its length.

Definition 3 (View). *For a given traffic snapshot $\mathcal{TS}$ with a set of lanes $\mathbb{L}$, a* view *V is defined as 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 that is visible in the view,*
- *$E \in \mathbb{I}$ is the identifier of the car under consideration.*

A subview *of V is obtained by restricting the lanes and extension we observe. For this we use sub- and superscript notation: $V^{L'} = (L', X, E)$ and $V_{X'} = (L, X', E)$, where L' and X' are subintervals of L and X, respectively.*

Sensor Function. Subsequently we will use a car dependent sensor function $\Omega_E : \mathbb{I} \times \mathbb{TS} \to \mathbb{R}_+$ which, given a car identifier and a traffic snapshot, provides the length of the corresponding car, as perceived by E.

Abbreviations For a given view $V = (L, X, E)$ and a traffic snapshot $\mathcal{TS} = (res, clm, pos, spd, acc)$ we use the following abbreviations:

$$res_V : \mathbb{I} \to \mathcal{P}(L) \text{ with } C \mapsto res(C) \cap L \tag{7}$$

$$clm_V : \mathbb{I} \to \mathcal{P}(L) \text{ with } C \mapsto clm(C) \cap L \tag{8}$$

$$len_V : \mathbb{I} \to \mathcal{P}(X) \text{ with } C \mapsto [pos(C), pos(C) + \Omega_E(C, \mathcal{TS})] \cap X \tag{9}$$

The functions (7) and (8) are restrictions of their counterparts in $\mathcal{TS}$ to the sets of lanes considered in this view. The function (9) gives us the part of the view occupied by a car C.[1]

Definition 4 formalizes the partitioning of discrete intervals. We need this slightly intricate notion to have a clearly defined chopping operation, even on the empty set of lanes.

Definition 4 (Chopping discrete intervals). *Let I_D be a discrete interval, i.e., $I_D = [l, n]$ for some $l, n \in \mathbb{L}$ or $I_D = \emptyset$. Then $I_D = I_D^1 \ominus I_D^2$ if and only if $I_D^1 \cup I_D^2 = I_D$, $I_D^1 \cap I_D^2 = \emptyset$, and both I_D^1 and I_D^2 are discrete convex intervals, which implies $\max(I_D^1) + 1 = \min(I_D^2)$ or $I_D^1 = \emptyset$ or $I_D^2 = \emptyset$.*

We define the following relations on views to have a consistent description of vertical and horizontal chopping operations.

Definition 5 (Relations of Views). *Let V_1, V_2 and V be views of a snapshot $\mathcal{TS}$. Then $V = V_1 \ominus V_2$ if and only if $V = (L, X, E)$, $L = L_1 \ominus L_2$, $V_1 = V^{L_1}$ and $V_2 = V^{L_2}$. Furthermore, $V = V_1 \oslash V_2$ if and only if $V = (L, [r, t], E)$ and there is an $s \in [r, t]$ such that $V_1 = V_{[r,s]}$ and $V_2 = V_{[s,t]}$.*

[1] This presentation differs slightly from our previous work in two ways. First, we do not restrict the set of identifiers anymore to the cars "visible" to E. Since the functions for the reservations, claims or length return the empty set for cars outside of V, such cars cannot satisfy the corresponding atomic formulas. The definition of res_V and clm_V was altered due to a technical mistake in the previous form.

To abstract from the borders of intervals during the definition of the semantics, we define the following norm giving the length of an interval. This notion coincides with the length measurement of DC [3].

Definition 6 (Measure of a real-valued interval). *Let $I_R = [r, t]$ be a real-valued interval, i.e. $r, t \in \mathbb{R}$. The* measure *of I_R is the norm $\|I_R\| = t - r$.*

We employ three sorts of variables. The set of variables ranging over car identifiers is denoted by CVar, with typical elements c and d. For referring to lengths and quantities of lanes, we use the sorts RVar and LVar ranging over real numbers and elements of the set of lanes $\mathbb{L}$, respectively. The set of all variables is denoted by Var. To refer to the car owning the current view, we use the special constant ego. Furthermore we use the syntax ℓ for the length of a view, i.e., the length of the extension of the view and ω for the width, i.e., the number of lanes. For simplicity, we only allow for addition between correctly sorted terms. However, it is straightforward to augment the definition with further arithmetic operations.

Definition 7 (Syntax). *We use the following definition of* terms*.*

$$\theta ::= n \mid r \mid \text{ego} \mid u \mid \ell \mid \omega \mid \theta_1 + \theta_2,$$

where $n \in \mathbb{L}$, $r \in \mathbb{R}$ and $u \in \text{Var}$ and θ_i are both of the same sort, and not elements of $\text{CVar} \cup \{\text{ego}\}$*. We denote the set of terms with Θ. The syntax of the* extended multi-lane spatial logic EMLSL *is given as follows.*

$$\phi ::= \bot \mid \theta_1 = \theta_2 \mid re(c) \mid cl(c) \mid \phi_1 \to \phi_2 \mid \forall z \bullet \phi_1 \mid \phi_1 \frown \phi_2 \mid \begin{matrix}\phi_2 \\ \phi_1\end{matrix} \mid M\phi$$

where $M \in \{\Box_{\mathsf{r}(c)}, \Box_{\mathsf{c}(c)}, \Box_{\mathsf{wd\ c}(c)}, \Box_{\mathsf{wd\ r}(c)}, \Box_\tau\}$, $c \in \text{CVar} \cup \{\text{ego}\}$, $z \in \text{Var}$, and $\theta_1, \theta_2 \in \Theta$ are of the same sort. We denote the set of all EMLSL formulas by Φ.

Definition 8 (Valuation and Modification). *A* valuation *is a function $\nu \colon \text{Var} \cup \{\text{ego}\} \to \mathbb{I} \cup \mathbb{R} \cup \mathbb{L}$. We silently assume valuations and their modifications to respect the sorts of variables. For a view $V = (L, X, E)$, we lift ν to a function ν_V evaluating terms, where variables and* ego *are interpreted as in ν, and $\nu_V(\ell) = \|X\|$ and $\nu_V(\omega) = |L|$. The function $+$ is interpreted as addition.*

Definition 9 (Semantics). *In the following, let θ_i be terms of the same sort, $c \in \text{CVar} \cup \{\text{ego}\}$ and $z \in \text{Var}$. The* satisfaction *of formulas with respect to a traffic snapshot $\mathcal{TS}$, a view $V = (L, X, E)$ and a valuation ν with $\nu(\text{ego}) = E$ is defined inductively as follows:*

$$\begin{aligned}
&\mathcal{TS}, V, \nu \not\models \bot && \text{for all } \mathcal{TS}, V, \nu \\
&\mathcal{TS}, V, \nu \models \theta_1 = \theta_2 && \Leftrightarrow\ \nu_V(\theta_1) = \nu_V(\theta_2) \\
&\mathcal{TS}, V, \nu \models re(c) && \Leftrightarrow\ |L| = 1 \text{ and } \|X\| > 0 \text{ and} \\
& && \quad res_V(\nu(c)) = L \text{ and } X = len_V(\nu(c)) \\
&\mathcal{TS}, V, \nu \models cl(c) && \Leftrightarrow\ |L| = 1 \text{ and } \|X\| > 0 \text{ and}
\end{aligned}$$

$$
\begin{array}{lcl}
 & & clm_V(\nu(c)) = L \text{ and } X = len_V(\nu(c)) \\
\mathcal{TS}, V, \nu \models \phi_1 \to \phi_2 & \Leftrightarrow & \mathcal{TS}, V, \nu \models \phi_1 \text{ implies } \mathcal{TS}, V, \nu \models \phi_2 \\
\mathcal{TS}, V, \nu \models \forall z \bullet \phi & \Leftrightarrow & \forall \alpha \in \mathbb{I} \cup \mathbb{R} \cup \mathbb{L} \bullet \mathcal{TS}, V, \nu \oplus \{z \mapsto \alpha\} \models \phi \\
\mathcal{TS}, V, \nu \models \phi_1 \frown \phi_2 & \Leftrightarrow & \exists V_1, V_2 \bullet V = V_1 \oplus V_2 \text{ and} \\
 & & \mathcal{TS}, V_1, \nu \models \phi_1 \text{ and } \mathcal{TS}, V_2, \nu \models \phi_2 \\
\mathcal{TS}, V, \nu \models \begin{smallmatrix}\phi_2\\ \phi_1\end{smallmatrix} & \Leftrightarrow & \exists V_1, V_2 \bullet V = V_1 \ominus V_2 \text{ and} \\
 & & \mathcal{TS}, V_1, \nu \models \phi_1 \text{ and } \mathcal{TS}, V_2, \nu \models \phi_2 \\
\mathcal{TS}, V, \nu \models \Box_{\mathsf{r}(c)} \phi & \Leftrightarrow & \forall \mathcal{TS}' \bullet \mathcal{TS} \xrightarrow{\mathsf{r}(\nu(c))} \mathcal{TS}' \text{ implies } \mathcal{TS}', V, \nu \models \phi \\
\mathcal{TS}, V, \nu \models \Box_{\mathsf{c}(c)} \phi & \Leftrightarrow & \forall \mathcal{TS}', n \bullet \mathcal{TS} \xrightarrow{\mathsf{c}(\nu(c), n)} \mathcal{TS}' \text{ implies } \mathcal{TS}', V, \nu \models \phi \\
\mathcal{TS}, V, \nu \models \Box_{\mathsf{wd\ c}(c)} \phi & \Leftrightarrow & \forall \mathcal{TS}' \bullet \mathcal{TS} \xrightarrow{\mathsf{wd\ c}(\nu(c))} \mathcal{TS}' \text{ implies } \mathcal{TS}', V, \nu \models \phi \\
\mathcal{TS}, V, \nu \models \Box_{\mathsf{wd\ r}(c)} \phi & \Leftrightarrow & \forall \mathcal{TS}', n \bullet \mathcal{TS} \xrightarrow{\mathsf{wd\ r}(\nu(c), n)} \mathcal{TS}' \text{ implies } \mathcal{TS}', V, \nu \models \phi \\
\mathcal{TS}, V, \nu \models \Box_{\tau} \phi & \Leftrightarrow & \forall \mathcal{TS}', t \bullet \mathcal{TS} \overset{t}{\Rightarrow} \mathcal{TS}' \text{ implies } \mathcal{TS}', V, \nu \models \phi
\end{array}
$$

In addition to the standard abbreviations of the remaining Boolean operators and the existential quantifier, we use $\top \equiv \neg\bot$. An important derived modality of our previous work [1] is the *somewhere* modality

$$\langle\phi\rangle \equiv \top \frown \begin{pmatrix}\top\\ \phi\\ \top\end{pmatrix} \frown \top.$$

Further, we use its dual operator *everywhere*. We abbreviate the modality *somewhere along the extension of the view* with the operator $\Diamond_\ell$, similar to the *on some subinterval* modality of DC.

$$[\phi] \equiv \neg \langle\neg\phi\rangle \qquad \Diamond_\ell \phi \equiv \top \frown \phi \frown \top \qquad \Box_\ell \phi \equiv \neg\Diamond_\ell \neg\phi$$

Likewise, abbreviations can be defined to express the modality *on some lane*. Furthermore, we define the diamond modalities for the transitions as usual, i.e., $\Diamond_* \phi \equiv \neg\Box_* \neg\phi$, where $* \in \{\mathsf{r}(c), \mathsf{c}(c), \mathsf{wd\ r}(c), \mathsf{wd\ c}(c), \tau\}$.

In the first definition of MLSL, we included the atom *free* to denote free space on the road, i.e., space which is neither occupied by a reservation nor by a claim. It was not possible to derive this atom from the others, since we were unable to express the existence of exactly one lane and a non-zero extension in the view. However, in the current presentation, *free* can be defined within EMLSL. Observe that a view of non-zero extension can be characterized by $\ell > 0 \equiv \neg(\ell = 0)$.

$$free \equiv \ell > 0 \wedge \omega = 1 \wedge \forall c \bullet \Box_\ell(\neg cl(c) \wedge \neg re(c))$$

Furthermore, we can define $\ell < r \equiv \neg(\ell = r \frown \top)$ and use the superscript φ^r to abbreviate the schema $\varphi \wedge \ell = r$. For reasons of clarity, we will not always use this abbreviation and write out the formula instead, to emphasize the restriction.

As an example, the following formula defines the behavior of a safe distance controller, i.e., as long as the car starts in a situation with free space in front of it, the formula demands that after an arbitrary time, there is still free space left.

$$\forall x, y \bullet \Diamond_\ell \left(\begin{array}{c} \omega = x \\ re(\text{ego}) \frown free \\ \omega = y \end{array} \right) \rightarrow \Box_\tau \left(\Diamond_\ell \left(\begin{array}{c} \omega = x \\ re(\text{ego}) \frown free \\ \omega = y \end{array} \right) \right)$$

We have to relate the lane in both the antecedent and the conclusion by the atoms $\omega = x$ and $\omega = y$ respectively. If we simply used $\langle re(\text{ego}) \frown free \rangle$, it would be possible for the reservations to be on different lanes, and hence, we would not ensure that free space is in front of each of ego's reservations at every point in time. However, the formula does not constrain how the situations may change, whenever reservations or claims are created or withdrawn.

Observe that it is crucial to combine acceleration and time transitions into a single modality $\Box_\tau$. Let ego drive on lane m with a velocity of v. If we only allowed for the passing of time, this formula would require all cars on m in front of ego to have a velocity $v_f \geq v$, while all cars behind ego had to drive with $v_b \leq v$. Hence the evolutions allow for more complex behavior in the underlying model.

Like for ITL [2] or DC [3], we call a formula *flexible* whenever its satisfaction is dependent on the current traffic snapshot and view. Otherwise the formula is *rigid*. However, since the spatial dimensions of EMLSL are not directly interrelated, we also distinguish *horizontally rigid* and *vertically rigid* formulas. The satisfaction of the former is independent of the extension of views, while for the latter, the amount of lanes in a view is of no influence. If a formula is only independent of the current traffic snapshot, we call it *dynamically rigid*.

Definition 10 (Types of Rigidity). *Let ϕ be a formula of EMLSL. We call ϕ* dynamically rigid*, if it does not contain any spatial atom, i.e., $re(c)$ or $cl(c)$ as a subformula. Furthermore, we call ϕ* horizontally rigid*, if it is dynamically rigid and in addition does not contain ℓ as a term. Similarly, ϕ is* vertically rigid*, if it is dynamically rigid and does not contain ω as a term. If ϕ is both vertically and horizontally rigid, it is simply* rigid.

Lemma 2. *Let ϕ by dynamically rigid and ϕ_H (ϕ_V) be horizontally (vertically) rigid. Then for all traffic snapshots $\mathcal{TS}$, $\mathcal{TS}'$, views V, V_1, V_2 and valuations ν,*

1. *$\mathcal{TS}, V, \nu \models \phi$ iff $\mathcal{TS}', V, \nu \models \phi$*
2. *Let $V = V_1 \oslash V_2$. Then $\mathcal{TS}, V, \nu \models \phi_H$ iff $\mathcal{TS}, V_i, \nu \models \phi_H$ (for $i \in \{1, 2\}$).*
3. *Let $V = V_1 \ominus V_2$. Then $\mathcal{TS}, V, \nu \models \phi_V$ iff $\mathcal{TS}, V_i, \nu \models \phi_V$ (for $i \in \{1, 2\}$).*

Proof. By induction on the structure of EMLSL formulas.

3 Undecidability of pure MLSL

In this section we give an undecidability result for the spatial fragment of EMLSL, i.e., we do not need the modalities for the discrete state changes of the

model or the evolutions. We will call this fragment *spatial MLSL*, subsequently. We reduce the halting problem of two-counter machines, which is known to be undecidable [9], to satisfaction of spatial MLSL formulas.

Intuitively, a two counter machine executes a branching program which manipulates a (control) state and increments and decrements two different counters c_1 and c_2. Formally, two counter machines consist of a set of states $Q = \{q_0, \ldots, q_m\}$, distinguished initial and final states $q_0, q_{fin} \in Q$ and a set of instructions I of the form shown in Tab. 1 (the instructions for the counter c_2 are analogous). The instructions mutate configurations of the form $s = (q_i, c_1, c_2)$, where $q_i \in Q$ and $c_1, c_2 \in \mathbb{N}$ into new configurations:

Table 1. Instructions for counter c_1 of a two-counter machine

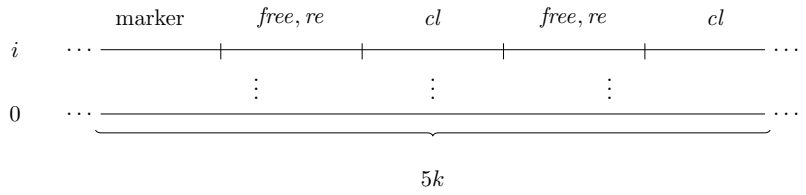
s	Instruction	s'
(q, c_1, c_2)	$q \xrightarrow{c_1^+} q_j$	$(q_j, c_1 + 1, c_2)$
$(q, 0, c_2)$	$q \xrightarrow{c_1^-} q_j, q_n$	$(q_j, 0, c_2)$
$(q, c + 1, c_2)$	$q \xrightarrow{c_1^-} q_j, q_n$	(q_n, c, c_2)

An *run from the initial configuration* of a two-counter machine (Q, q_0, q_{fin}, I) is a sequence of configurations $(q_0, 0, 0) \xrightarrow{i_0} \ldots \xrightarrow{i_p} (q_{p+1}, c_{p+1}, c'_{p+1})$, where each i_j is an instance of an instruction within I. If $q_{p+1} = q_{fin}$, the run is *halting*.

We follow the approach of Zhou et al. [7] for DC. They encode the configurations in recurring patterns of length $4k$, where the first part constitutes the current state, followed by the contents of the first counter. The third part is filled with a marker to distinguish the counters, and is finally followed by the contents of the second counter. Each of these parts is exactly of length k.

Zhou et al. could use distinct observables for the state of the machine, counters and separating delimiters, since DC allows for the definition of arbitrary many observable variables. We have to modify this encoding since within spatial MLSL we are restricted to two predicates for reservations and claims, and the derived predicate for free space, respectively. Furthermore, due to the constraints on EMLSL models in Def. 1, we cannot use multiple occurrences of reservations of a unique car to stand, e.g., for the values of one counter. Hence we have to existentially quantify all mentions of reservations and claims. We will never reach an upper limit of existing cars, since we assume $\mathbb{I}$ to be countably infinite.

The current state of the machine q_i is encoded by the number of lanes below the current configuration, the states of the counters is described by a sequence of reservations, separated by a single claim. To safely refer to the start of a configuration, we also use an additional marker consisting of a claim, an adjacent reservation and again a claim. Each part of the configurations is assumed to have length k. Free space separates the reservations within one counter from each other and from the delimiters. Intuitively, a configuration is encoded as follows:



To enhance the readability of our encoding, we use the abbreviation $\text{marker} \equiv \exists c \bullet cl(c) \frown \exists c \bullet re(c) \frown \exists c \bullet cl(c)$ to denote the start of a configuration.

Like Zhou et al., we ensure that reservations and claims are mutually exclusive. We do not have to consider *free*, since it is already defined as the absence of both reservations and claims. Observe that we use the square brackets to denote the *everywhere* modality (cf. Section 2).

$$\text{mutex} = \forall c, d \bullet [cl(c) \to \neg re(d)) \wedge (re(c) \to \neg cl(d)]\,.$$

The initial marking $(q_0, 0, 0)$ is then defined by the following formula.

$$\text{init} = \left(\begin{array}{c} [\neg \exists c \bullet cl(c)] \\ \text{marker}^k \frown \mathit{free}^k \frown (\exists c \bullet cl(c))^k \frown \mathit{free}^k \frown (\exists c \bullet cl(c))^k \\ \omega = 0 \end{array} \right) \frown \top$$

We have to ensure that the configurations occur periodically after every $5k$ spatial units. Therefore, we use the following schema $Per(\mathcal{D})$. Observe that we only require that the lanes surrounding the formula $\mathcal{D}$ do not contain claims. This ensures on the one hand that no configuration lies in parallel with the formula $\mathcal{D}$, since well-defined configurations have to include claims. On the other hand, it allows for satisfiability of the formula, since we do not forbid the occurrence of reservations, which are needed for the claims within the configurations.

$$Per(\mathcal{D}) = \left[\left(\begin{array}{c} [\neg \exists c \bullet cl(c)] \\ \mathcal{D} \\ [\neg \exists c \bullet cl(c)] \end{array} \frown \ell = 5k \right) \to \left(\ell = 5k \frown \begin{array}{c} [\neg \exists c \bullet cl(c)] \\ \mathcal{D} \\ [\neg \exists c \bullet cl(c)] \end{array} \right) \right]$$

Note that we did not constrain on which lane the periodic behavior occurs. This will be defined by the encoding of the operations.

Now we may define the periodicity of the delimiters and the counters. Here we also have to slightly deviate from Zhou et al.: we are not able to express the statement "almost everywhere *free* or $re(c)$ holds," directly. We have to encode it by ensuring that on every subinterval with a length greater than zero, we can find another subinterval which satisfies *free* or $re(c)$. This expresses in particular, that no claim may occur, due to the mutual exclusion property.

$$\begin{aligned} \text{periodic} = {} & Per((\Box_\ell(\ell > 0 \to \top \frown (\mathit{free} \vee \exists c \bullet re(c)) \frown \top) \wedge \omega = 1)^k) \\ & \wedge Per((\exists c \bullet cl(c))^k) \wedge Per(\text{marker}^k) \end{aligned}$$

We turn to the encoding of the operation $q_i \xrightarrow{c_1^+} q_j$, i.e., the machine goes from q_i to q_j and increments the first counter by one. Similar to Zhou et al.,

we use encodings of the form $\neg(\mathcal{D}_1 \frown \neg\mathcal{D}_2)$, meaning "whenever the beginning of the view satisfies $\mathcal{D}_1$, the next part satisfies $\mathcal{D}_2$."

The formula F_1 copies the reservations of counter one of state q_i to the corresponding places in counter one in state q_j.

$$F_1 = \neg\left(\left(\begin{array}{c}\top\\ \mathrm{marker}^k \frown \ell < k \frown \exists c \bullet re(c) \frown ((\exists c \bullet re(c) \frown \top) \wedge \ell = 5k)\\ \omega = i\end{array}\right) \frown \neg\left(\begin{array}{c}\top\\ \ell = 0 \vee (\exists c \bullet re(c) \frown \top)\\ \omega = j\end{array}\right)\right)$$

We use a similar formula F_{free} to copy the free space before the reservations.

The formulas F_2 and F_3 handle the addition of another reservation to the counter. We have to distinguish between an empty counter and one already containing reservations.

$$F_2 = \left(\begin{array}{c}\top\\ \mathrm{marker}^k \frown \mathit{free}^k \frown \ell = 5k\\ \omega = i\end{array}\right) \rightarrow \left(\begin{array}{c}\top\\ \top \frown (\mathit{free} \frown \exists c \bullet re(c) \frown \mathit{free})^k\\ \omega = j\end{array}\right)$$

$$F_3 = \left(\begin{array}{c}\top\\ \mathrm{marker}^k c \frown \ell < k \frown \exists c \bullet re(c) \frown (\mathit{free} \frown \exists c \bullet cl(c) \frown \top) \wedge \ell = 6k\\ \omega = i\end{array}\right) \rightarrow \left(\begin{array}{c}\top\\ \top \frown (\mathit{free} \frown \exists c \bullet re(c) \frown \mathit{free} \frown \exists c \bullet cl(c))^k\\ \omega = j\end{array}\right)$$

In addition, we need formulas which copy of contents of the second counter to the new configuration, similar to F_1.

Let I_C be the set of the machine's instructions and $F(i)$ be the conjunction of the formulas encoding operation i and q_{fin} its final state. Then

$$\mathrm{halt}(C) = \mathrm{init} \wedge \mathrm{periodic} \wedge \mathrm{mutex} \wedge \bigwedge_{i \in I_C} \Box_\ell F(i) \wedge \Diamond_\ell \left(\begin{array}{c}\top\\ \exists c \bullet cl(c)\\ \omega = \mathit{fin}\end{array}\right).$$

If and only if $\mathrm{halt}(C)$ is satisfiable, the machine contains a halting run. This holds since only configurations may contain claims (as defined in the formalization of periodicity), and whenever the machine reaches its final state, it halts. Hence the halting problem of two counter machines with empty initial configuration reduces to satisfiability of spatial MLSL formulas.

Proposition 1. *Let C be a two counter machine. Then C has a halting run if and only if* $\mathrm{halt}(C)$ *is satisfiable.*

The main theorem of this section is a corollary of Prop. 1.

Theorem 1. *The satisfiability problem of spatial MLSL is undecidable.*

Even though we used the full power of spatial MLSL in the proof, i.e., we used both ℓ and ω, the proof would be possible without using the latter. For that, we would not be able to encode the state of the configuration in the lanes, but by a similar way to the markers in the formulas. For example, the formula $(\exists c \bullet cl(c) \frown \exists c \bullet re(c) \frown \exists c \bullet cl(c))^k$ would denote the state q_0, and with another iteration of $re(c)$, it would denote q_1 and so on. If we remove the references to more than one lane in each of the formulas above, the reservations and claims would already imply that only one lane exists, and hence, the use of ω within the abbreviation *free* could be omitted. This shows that spatial MLSL is already undecidable even if we only use ℓ.

4 Labelled Natural Deduction for EMLSL

Despite the negative decidability result of the previous section, we define a system of labelled natural deduction [10,5,11] for the full logic EMLSL. That is, the rules of the deduction system do not operate on formulas ϕ, but on *labelled formulas* $w\colon \phi$, where w is a term of a *labelling algebra* and ϕ is a formula of EMLSL. They may connect the derivations of formulas and relations between the terms w to allow for a tighter relationship between both. The labelling algebra is more involved than for standard modal logics, since EMLSL is in essence a multi-dimensional logic, where the modalities are not interdefinable. Obviously, the spatial modalities can not be defined by the dynamic modalities and vice versa. Furthermore, neither can the dynamic modalities be defined by each other in general. Consider, e.g., the modalities $\Box_{\mathsf{r}(c)}$ and $\Box_{\mathsf{c}(c)}$. Both of these modalities rely on different transitions between the models, which are only indirectly related.

The labels of the algebra consist of tuples $\mathcal{TS}, V$, where similar to the semantics, $\mathcal{TS}$ is the name of a traffic snapshot and V a view. The algebra is then twofold. The relations of the form $V = V_1 \oplus V_2$ and $V = V_1 \ominus V_2$ define ternary reachability relations between views for the spatial modalities. Relations between snapshots, e.g., $\mathcal{TS} \xrightarrow{\mathsf{r}(C)} \mathcal{TS}'$ describe the behavior of transitions. The relations within the labelling algebra for traffic snapshots directly correspond to the dynamic modalities. For example, we have $\mathcal{TS} \xrightarrow{\mathsf{c}(C)} \mathcal{TS}'$, whenever there exists an $n \in \mathbb{N}$ such that $\mathcal{TS} \xrightarrow{\mathsf{c}(C,n)} \mathcal{TS}'$.

We do not give a deduction system for the transitions between snapshots, since the conditions needed to hold between them are of a very complex nature, i.e., they are definable only with the power of full first-order logic with functions, identity and arithmetic. Hence we would not achieve a system with a nice distinction between the relational deductions and the deductions of labelled formulas [5,11]. Instead we simply assume the existence of the relations between snapshots whenever needed. That is, we will often have, e.g., the existence of a transition in our set of assumptions. However, we give simple rules defining that chopping of a view into two subviews is always possible.

Definition 11 (Labelled Formulas and Relational Formulas). *Let $\mathcal{TS}$ be a name for a traffic snapshot, V a name for a view and ϕ a formula according to Definition 7. Then $\mathcal{TS}, V\colon \phi$ is a* labelled formula *of EMLSL. Furthermore, we call $\mathcal{TS} \xrightarrow{*} \mathcal{TS}'$, $V = V_1 \oplus V_2$ and $V = V_1 \ominus V_2$* relational formulas*, where $\xrightarrow{*}$ is a relation of the labelling algebra.*

To have a meaningful soundness result of the calculus, we give the relation of the semantics of labelled formulas and normal formulas. Observe that we do not define a completely independent notion of models, but only use a valuation for this purpose. This is due to the semantic information which is still comprised within the views and traffic snapshots.

Definition 12 (Satisfaction of Labelled Formulas). *We say that a valuation ν* satisfies *a labelled formula $\mathcal{TS}, V\colon \phi$, written $\nu \models \mathcal{TS}, V\colon \phi$ if and only if $\mathcal{TS}, V, \nu \models \phi$. Furthermore,*

$$
\begin{aligned}
\nu \models \mathcal{TS}_1 \xrightarrow{\mathsf{r}(c)} \mathcal{TS}_2 \quad &\Leftrightarrow \quad \mathcal{TS}_1 \xrightarrow{\mathsf{r}(\nu(c))} \mathcal{TS}_2, \\
\nu \models \mathcal{TS}_1 \xrightarrow{\mathsf{wd}\ \mathsf{r}(c)} \mathcal{TS}_2 \quad &\Leftrightarrow \quad \exists n \bullet \mathcal{TS}_1 \xrightarrow{\mathsf{wd}\ \mathsf{r}(\nu(c),n)} \mathcal{TS}_2, \\
\nu \models \mathcal{TS}_1 \xrightarrow{\mathsf{c}(c)} \mathcal{TS}_2 \quad &\Leftrightarrow \quad \exists n \bullet \mathcal{TS}_1 \xrightarrow{\mathsf{c}(\nu(c),n)} \mathcal{TS}_2 \\
\nu \models \mathcal{TS}_1 \xrightarrow{\mathsf{wd}\ \mathsf{c}(c)} \mathcal{TS}_2 \quad &\Leftrightarrow \quad \mathcal{TS}_1 \xrightarrow{\mathsf{wd}\ \mathsf{c}(\nu(c))} \mathcal{TS}_2 \\
\nu \models \mathcal{TS}_1 \xrightarrow{\tau} \mathcal{TS}_2 \quad &\Leftrightarrow \quad \exists t \bullet \mathcal{TS}_1 \overset{t}{\Rightarrow} \mathcal{TS}_2
\end{aligned}
$$

The relational formulas $V = V_1 \oplus V_2$ and $V = V_1 \ominus V_2$ are independent of the valuation at hand, and hence are satisfied whenever V_1 and V_2 combined according to Definition 5 result in V.

Definition 13 (Derivation). *A* derivation *of a labelled formula $\mathcal{TS}, V\colon \phi$ from a set of labelled formulas Γ and a set of relational formulas Δ is a tree, where the root is $\mathcal{TS}, V\colon \phi$, each leaf is an element of Γ or Δ and each node within the tree is a result of an application of one of the rules defined subsequently. We denote the existence of such a derivation by $\Gamma, \Delta \vdash \mathcal{TS}, V\colon \phi$.*

Following Rasmussen [6], we define predicates for chop-freeness of formulas and rigidity of terms and formulas. To increase the deducible theorems, we differentiate between *vertical* and *horizontal* chop-freeness and rigidity. These properties are especially important for the correct instantiation of terms, i.e., for the elimination of universal quantifiers.

Example 1. Consider the formula

$$\forall x \bullet \begin{pmatrix} \ell = x \\ \ell = x \end{pmatrix} \rightarrow \ell = x \Big),$$

which is a theorem of MLSL, since the length of a view is not changed by chopping vertically. If we use classical universal quantifier instantiation and substitute the vertically flexible term ω for x, then we would get

$$\begin{array}{l}\ell = \omega \\ \ell = \omega\end{array} \rightarrow \ell = \omega. \tag{10}$$

Now let V be a view satisfying the antecedent of (10). Then V can be vertically chopped such that its length equals its width on both subviews. Now let $\ell = c$. Then also $\omega = c$ for both subviews. Since V consists of both these subviews, V satisfies $\omega = 2c$. But the conclusion of (10) states that V should satisfy $\omega = \ell = c$. However, we could of course substitute x by the vertically rigid term ℓ.

We denote vertical (horizontal) chop-freeness by the predicate vcf (hcf) and vertical (horizontal) rigidity by vri (hri). The rules for the definition of all four predicates are straightforward, since both rigidity and chop-freeness are syntactic properties. All atomic formulas are vertically and horizontally chop-free. For $\oslash$ being a Boolean operator or the horizontal chop $\frown$, the following rules give vertical chop-freeness.

$$\frac{\mathrm{vcf}(\phi) \quad \mathrm{vcf}(\psi)}{\mathrm{vcf}(\phi \oslash \psi)}\,\mathrm{vcf} \oslash \mathrm{I} \qquad \frac{\mathrm{vcf}(\phi \oslash \psi)}{\mathrm{vcf}(\phi)}\,\mathrm{vcf} \oslash \mathrm{E} \qquad \frac{\mathrm{vcf}(\phi \oslash \psi)}{\mathrm{vcf}(\psi)}\,\mathrm{vcf} \oslash \mathrm{E}$$

The rules for quantifiers and the horizontal rules are defined similarly.

For terms, ℓ is vertically rigid and ω is horizontally rigid. The spatial atoms $re(c)$, $cl(c)$ and *free* are neither horizontally nor vertically rigid, since they require the view to possess an extension greater than zero and exactly one lane. Equality is both vertically and horizontally rigid, as long as both compared terms are rigid. Below, we show some exemplary rules, where $\otimes$ is an arbitrary binary operator.

$$\frac{\mathrm{hri}(\phi) \quad \mathrm{hri}(\psi)}{\mathrm{hri}(\phi \otimes \psi)}\,\mathrm{hri} \otimes \mathrm{I} \qquad \frac{\mathrm{hri}(\phi \otimes \psi)}{\mathrm{hri}(\phi)}\,\mathrm{hri} \otimes \mathrm{E} \qquad \frac{\mathrm{hri}(\phi \otimes \psi)}{\mathrm{hri}(\psi)}\,\mathrm{hri} \otimes \mathrm{E}$$

$$\frac{\dfrac{}{\mathbb{E}V', V''(V = V' \ominus V'')}\,\mathrm{VDec} \qquad \begin{array}{c}[V = V_2 \ominus V_2] \\ \vdots \\ \mathcal{TS}, V_3 \colon \phi\end{array}}{\mathcal{TS}, V_3 \colon \phi}\,\mathbb{E}\mathrm{E}$$

We have only two simple rules for the relations between views. First, we state that each view V is decomposable into two subviews. This is true, since we allow for the empty view, i.e., the view without lanes or with a point-like extension. We use $\mathbb{E}$ to denote existential quantification over views. To use the relations between views, we have to be able to instantiate views, i.e., we have to introduce a rule for *elimination of existential quantifiers over views*. As a side condition for this elimination rule, we require that $\mathcal{TS}, V_3 \colon \phi$ is not dependent on any assumption including V_1 or V_2 as a label, except for $V = V_1 \ominus V_2$. The rule itself is a straightforward adaptation of the classical rule. Again, we only show the case for the vertical relations.

The intuition of rigidity is formalized in the following rules. Whenever a formula is horizontally rigid, the formula holds on all views horizontally reachable from the current view. Observe that the traffic snapshot may change arbitrarily, since horizontally rigid formulas are also dynamically rigid. The rules for vertically rigidity are similar.

$$\frac{\mathcal{TS}, V\colon \phi \qquad \mathrm{hri}(\phi) \qquad V = V_1 \oslash V_2}{\mathcal{TS}', V_1\colon \phi}\, R_H \qquad \frac{\mathcal{TS}, V\colon \phi \qquad \mathrm{hri}(\phi) \qquad V = V_1 \oslash V_2}{\mathcal{TS}', V_2\colon \phi}\, R_H$$

$$\frac{\mathcal{TS}, V_1\colon \phi \qquad \mathrm{hri}(\phi) \qquad V = V_1 \oslash V_2}{\mathcal{TS}', V\colon \phi}\, R_H \qquad \frac{\mathcal{TS}, V_2\colon \phi \qquad \mathrm{hri}(\phi) \qquad V = V_1 \oslash V_2}{\mathcal{TS}', V\colon \phi}\, R_H$$

For the first-order operators, we use the typical definitions of labelled natural deduction rules [5]. The only difference lies in the rules for quantification. We may instantiate an universally quantified variable with a horizontally (vertically) rigid, if the formula is vertically (horizontally) chop-free. If the formula is completely chop-free, we may instantiate the variable with an arbitrary term. Similarly, rigid terms may instantiate x in arbitrary formulas. In all cases, a side condition for the instantiation is that s respects the sort of x.

$$\frac{\mathcal{TS}, V\colon \forall x \bullet \phi \qquad \mathrm{hcf}(\phi) \quad \mathrm{vri}(s)}{\mathcal{TS}, V\colon \phi[x \mapsto s]}\, \forall \mathrm{E} \qquad \frac{\mathcal{TS}, V\colon \forall x \bullet \phi \qquad \mathrm{vcf}(\phi) \quad \mathrm{hri}(s)}{\mathcal{TS}, V\colon \phi[x \mapsto s]}\, \forall \mathrm{E}$$

$$\frac{\mathcal{TS}, V\colon \forall x \bullet \phi \qquad \mathrm{hcf}(\phi) \quad \mathrm{vcf}(\phi)}{\mathcal{TS}, V\colon \phi[x \mapsto s]}\, \forall \mathrm{E} \qquad \frac{\mathcal{TS}, V\colon \forall x \bullet \phi \qquad \mathrm{hri}(s) \quad \mathrm{vri}(s)}{\mathcal{TS}, V\colon \phi[x \mapsto s]}\, \forall \mathrm{E}$$

$$\frac{\mathcal{TS}, V_1\colon \phi \qquad \mathcal{TS}, V_2\colon \psi \qquad V = V_1 \oslash V_2}{\mathcal{TS}, V\colon \phi \frown \psi}\, \frown \mathrm{I}$$

$$\frac{\mathcal{TS}, V\colon \phi \frown \psi \qquad \begin{array}{c}[\mathcal{TS}, V_1\colon \phi] \\ [\mathcal{TS}, V_2\colon \psi] \\ [V = V_1 \oslash V_2] \\ \vdots \\ \mathcal{TS}', V'\colon \chi\end{array}}{\mathcal{TS}', V'\colon \chi}\, \frown \mathrm{E}$$

The elimination and introduction rules for the chop modalities are adopted from Rasmussen [6], and resemble the rules for existential quantification. We only show the case for the horizontal chop, the rules for vertical chopping are obtained straightforwardly, by replacing horizontal modalities and relations by the vertical ones.

The chopping of intervals is not ambiguous, i.e., there is a unique view of a certain length at the beginning of a view. This is the *single decomposition property* [12] of interval logics and captured in the following rules. Hence when there are two vertical chops of a view, and the upper parts are of equal width, we can derive that the same formulas hold on the lower parts. Even though we only show the vertical set of rules, similar rules hold for the horizontal chopping of views.

$$\frac{\mathcal{TS}, V_1\colon \phi \qquad \mathcal{TS}, V_2\colon \omega = s \quad \mathcal{TS}, V_2'\colon \omega = s \qquad \mathrm{vri}(s) \quad V = V_1 \ominus V_2 \quad V = V_1' \ominus V_2'}{\mathcal{TS}, V_1'\colon \phi}\, VD$$

$$\frac{\mathcal{TS}, V_2\colon \phi \qquad \mathcal{TS}, V_1\colon \omega = s \quad \mathcal{TS}, V_1'\colon \omega = s \qquad \mathrm{vri}(s) \quad V = V_1 \ominus V_2 \quad V = V_1' \ominus V_2'}{\mathcal{TS}, V_2'\colon \phi}\, VD$$

The additivity of length and width can be formalized by the following rules.

$$\frac{\mathcal{TS}, V_1 : \omega = s \qquad \mathcal{TS}, V_2 : \omega = t \qquad \mathrm{vri}(s) \quad \mathrm{vri}(t) \quad V = V_1 \ominus V_2}{\mathcal{TS}, V : \omega = s + t}\, V+\mathrm{I}$$

$$\frac{\mathcal{TS}, V : \omega = s + t \qquad \mathrm{vri}(s) \quad \mathrm{vri}(t) \qquad \begin{array}{c}[\mathcal{TS}, V_1 : \omega = s] \\ [\mathcal{TS}, V_2 : \omega = t] \\ [V = V_1 \ominus V_2] \\ \vdots \\ \mathcal{TS}', V' : \phi\end{array}}{\mathcal{TS}', V' : \phi}\, V+\mathrm{E}$$

$$\frac{\mathcal{TS} \xrightarrow{*} \mathcal{TS}' \qquad \mathcal{TS}, V : \Box_* \phi}{\mathcal{TS}', V : \phi}\, \Box_*\mathrm{E}$$

$$\frac{\begin{array}{c}[\mathcal{TS} \xrightarrow{*} \mathcal{TS}'] \\ \vdots \\ \mathcal{TS}', V : \phi\end{array}}{\mathcal{TS}, V : \Box_* \phi}\, \Box_*\mathrm{I}$$

The dynamic modalities are defined along the lines of Basin et al. [5]. If a transition from the current snapshot is possible, the box modalities may be eliminated and if we can prove that under the assumption of a transition $*$, ϕ holds on the now reachable snapshot, $\Box_* \phi$ holds.

Finally, we have to define how the spatial atoms behave with respect to occurring transitions. There are two types of rules in general, *stability rules* and *activity rules*. Stability rules define which atoms stay true after a snapshot changes according to a certain transition. The truth of all reservation and claims of cars not involved in the transition are unchanged. Only one stability rule for creating reservations includes the car which is the source of the transition. We will show this rule and one example for typical stability. The *activity rules* state how the reservations and claims of cars will change according to the transitions.

The following stability rules show that whenever a car creates a new claim, the reservations and claims of other cars are unchanged. We have similar stability rules for the other types of transitions.

$$\frac{\mathcal{TS}, V : cl(c) \qquad \mathcal{TS} \xrightarrow{\mathsf{c}(d)} \mathcal{TS}' \qquad \mathcal{TS}, V : c \neq d}{\mathcal{TS}', V : cl(c)}\, \xrightarrow{\mathsf{c}(c)}\mathrm{S}$$

$$\frac{\mathcal{TS}, V : re(c) \qquad \mathcal{TS} \xrightarrow{\mathsf{c}(d)} \mathcal{TS}' \qquad \mathcal{TS}, V : c \neq d}{\mathcal{TS}', V : re(c)}\, \xrightarrow{\mathsf{c}(c)}\mathrm{S}$$

The activity rule for $\mathsf{c}(c)$ implies two properties. First, a claim may only be created when only one reservation exists. Second, the newly created claim resides on one side of the existing reservation. Observe that the negations in the antecedent would allow for empty views on both sides of the reservation, but this case is prohibited by the antecedent that the view V is two lanes wide.

$$\frac{\mathcal{TS}, V : \begin{array}{c}\neg(re(c) \vee cl(c)) \\ re(c) \\ \neg(re(c) \vee cl(c))\end{array} \qquad \mathcal{TS} \xrightarrow{\mathsf{c}(d)} \mathcal{TS}' \qquad \mathcal{TS}, V : c = d \quad \mathcal{TS}, V : \omega = 2}{\mathcal{TS}', V : \begin{array}{c} re(c) \\ cl(c)\end{array} \vee \begin{array}{c} cl(c) \\ re(c)\end{array}}\, \xrightarrow{\mathsf{c}(c)}\mathrm{A}$$

Rules for the creation of reservations in between traffic snapshots are:

$$\frac{\mathcal{TS}, V : cl(c) \qquad \mathcal{TS} \xrightarrow{\mathsf{r}(d)} \mathcal{TS}' \qquad \mathcal{TS}, V : c = d}{\mathcal{TS}', V : re(c)} \xrightarrow{\mathsf{r}(c)} \mathrm{A}$$

$$\frac{\mathcal{TS}, V : re(c) \qquad \mathcal{TS} \xrightarrow{\mathsf{r}(d)} \mathcal{TS}' \qquad \mathcal{TS}, V : c = d}{\mathcal{TS}', V : re(c)} \xrightarrow{\mathsf{r}(c)} \mathrm{S}$$

The following activity rules define the withdrawal of reservations and claims.

$$\frac{\mathcal{TS}, V : \begin{matrix} re(c) \\ re(c) \end{matrix} \qquad \mathcal{TS} \xrightarrow{\mathsf{wd}\ \mathsf{r}(d)} \mathcal{TS}' \qquad \mathcal{TS}, V : c = d}{\mathcal{TS}', V : \begin{matrix} re(c) \\ \neg re(c) \end{matrix} \vee \begin{matrix} \neg re(c) \\ re(c) \end{matrix}} \xrightarrow{\mathsf{wd}\ \mathsf{r}(c)} \mathrm{A}$$

$$\frac{\mathcal{TS}, V : cl(c) \qquad \mathcal{TS} \xrightarrow{\mathsf{wd}\ \mathsf{c}(d)} \mathcal{TS}' \qquad \mathcal{TS}, V : c = d}{\mathcal{TS}', V : \neg cl(c)} \xrightarrow{\mathsf{wd}\ \mathsf{c}(c)} \mathrm{A}$$

We also have rules for "backwards" reasoning, i.e., if our current snapshot is reachable from another, we may draw conclusions about the originating snapshot. Again, we differentiate between activity and stability rules (omitted here).

$$\frac{\mathcal{TS}', V : re(c) \qquad \mathcal{TS} \xrightarrow{\mathsf{r}(d)} \mathcal{TS}' \qquad \mathcal{TS}, V : c = d}{\mathcal{TS}, V : re(c) \vee cl(c)} \xleftarrow{\mathsf{r}} \mathrm{A}$$

$$\frac{\mathcal{TS}', V : cl(c) \qquad \mathcal{TS} \xrightarrow{\mathsf{c}(d)} \mathcal{TS}' \qquad \mathcal{TS}, V : c = d}{\mathcal{TS}, V : \neg cl(c)} \xleftarrow{\mathsf{c}} \mathrm{A}$$

Observe that we can not reason backwards along withdrawal transitions, since these may be taken without changing any reservations and claims (cf. Def. 2).

Theorem 2. *The calculus of labelled natural deduction for EMLSL is sound.*

As an example, we derive a variant of the *reservation lemma*, which we proved informally in our previous work [1].

Lemma 3 (Reservation). *A reservation of a car c observed directly after c created it, was either already present or is due to a previously existing claim. I.e., assuming $\mathcal{TS} \xrightarrow{\mathsf{r}(c)} \mathcal{TS}'$, the formula $(re(c) \vee cl(c)) \leftrightarrow \Box_{\mathsf{r}(c)} re(c)$ holds. Hence*

$$\{\mathcal{TS} \xrightarrow{\mathsf{r}(c)} \mathcal{TS}'\} \vdash \mathcal{TS}, V : (re(c) \vee cl(c)) \leftrightarrow \Box_{\mathsf{r}(c)} re(c).$$

Proof. The existence of the transition is of major importance for the elimination of the box modality in the proof using the backwards reasoning rule. For reasons of simplicity, we use a variant of the stability rules and activity rules, where d in the transition has been replaced by c, and hence we do not need the extra assumption of $\mathcal{TS}, V : c = d$. We use two auxiliary derivations, which allow us to infer the existence of a reservation on the snapshot after taking a transition.

$$\Pi_{\mathrm{S}}\colon\ \frac{[\mathcal{TS}, V : re(c)]_1 \qquad [\mathcal{TS} \xrightarrow{\mathsf{r}(c)} \mathcal{TS}']_2}{\mathcal{TS}', V : re(c)} \qquad \Pi_{\mathrm{A}}\colon\ \frac{[\mathcal{TS}, V : cl(c)]_1 \qquad [\mathcal{TS} \xrightarrow{\mathsf{r}(c)} \mathcal{TS}']_2}{\mathcal{TS}', V : re(c)}$$

Derivation of $\vdash \mathcal{TS}, V : (re(c) \vee cl(c)) \to \Box_{\mathsf{r}(c)} re(c)$.

$$\vee\mathrm{E}_1\ \frac{\dfrac{\Pi_{\mathrm{S}}}{\mathcal{TS}', V : re(c)} \qquad \dfrac{\Pi_{\mathrm{A}}}{\mathcal{TS}', V : re(c)} \qquad [\mathcal{TS}, V : re(c) \vee cl(c)]_3}{\dfrac{\dfrac{\mathcal{TS}', V : re(c)}{\mathcal{TS}, V : \Box_{\mathsf{r}(c)} re(c)}\,\Box_{\mathsf{r}(c)}\mathrm{I}_2}{\mathcal{TS}, V : (re(c) \vee cl(c)) \to \Box_{\mathsf{r}(c)} re(c)} \to \mathrm{I}_3}$$

Derivation of $\{\mathcal{TS} \xrightarrow{\mathsf{r}(c)} \mathcal{TS}'\} \vdash \mathcal{TS}, V : \Box_{\mathsf{r}(c)} re(c) \to (re(c) \vee cl(c))$.

$$\frac{\dfrac{\dfrac{[\mathcal{TS}, V : \Box_{\mathsf{r}(c)} re(c)]_1 \qquad \mathcal{TS} \xrightarrow{\mathsf{r}(c)} \mathcal{TS}'}{\mathcal{TS}', V : re(c)}\,\Box_{\mathsf{r}(c)}\mathrm{E} \qquad \mathcal{TS} \xrightarrow{\mathsf{r}(c)} \mathcal{TS}'}{\mathcal{TS}, V : re(c) \vee cl(c)}\,\xleftarrow{\mathsf{r}(c)}}{\mathcal{TS}, V : \Box_{\mathsf{r}(c)} re(c) \to (re(c) \vee cl(c))} \to \mathrm{I}_1$$

□

Since models of EMLSL are based on the real numbers, we cannot hope for a complete deduction system.

5 Related and Future Work

Most related work on spatial logics is focused on purely qualitative spatial reasoning [13], e.g., the expressible properties concern topological relations [14]. Logics expressing quantitative spatial properties are rare, an example is Schäfer's Shape Calculus (SC) [4], which is a very general extension of DC. Contrasting SC, the focus of EMLSL lies on a restricted field of application, i.e., highway traffic. EMLSL is an instance of a multi-dimensional and multi-modal logic [15], since it consists of various different modal operators, which are not interdefinable. It is also a combination of binary modalities, i.e., the chopping operations, and unary box-like modalities, i.e., the dynamic modal operators. Labelled natural deduction for (multi-)modal logics has been studied intensely recently. E.g., when the rules for relational formulas can be defined with horn clauses as antecedents, nice meta-theoretical properties like normalization of proofs can be established [5,11]. In intuitionistic modal logic, similar results are obtained, when the relational theory is defined using only geometric sequents [16]. Unfortunately, even with our restricted set of rules for view relations, these results do not carry over to our setting, since we made use of existential quantification on views. Still we would like to explore how rules for the manipulation of traffic snapshots could blend in. However, due to the complex internal structure of traffic snapshots, we do not expect such rules to be definable by horn clauses. Rasga et al. investigated the fibring [17] of labelled deductive systems [18]. We assume that the deduction system of Sec. 4 is an instance of such a fibring, where the Boolean operators are shared between all deduction systems involved. A further classification of

EMLSL (or a suitable subset) and its proof system within the framework of fibring and multi-dimensional logics would be of interest in order to use preservation results concerning, e.g., decidability. Finally, an implementation within a general theorem prover like Isabelle [19] similar to implementations for modal or interval logics [5,11,6] would increase the usefulness of the proof system.

References

1. Hilscher, M., Linker, S., Olderog, E., Ravn, A.: An abstract model for proving safety of multi-lane traffic manoeuvres. In: ICFEM, Springer (2011) 404–419
2. Moszkowski, B.: A temporal logic for multilevel reasoning about hardware. Computer **18** (1985) 10–19
3. Zhou Chaochen, Hoare, C.A.R., Ravn, A.P.: A calculus of durations. Information Processing Letters **40** (1991) 269 – 276
4. Schäfer, A.: A calculus for shapes in time and space. In Liu, Z., Araki, K., eds.: ICTAC 2004. Volume 3407 of LNCS., Springer (2005) 463–478
5. Basin, D., Matthews, S., Viganò, L.: Natural deduction for non-classical logics. Studia Logica **60** (1998) 119–160
6. Rasmussen, T.M.: Labelled natural deduction for interval logics. In Fribourg, L., ed.: CSL. Volume 2142 of LNCS. Springer (2001) 308–323
7. Zhou Chaochen, Hansen, M.R., Sestoft, P.: Decidability and undecidability results for duration calculus. In Enjalbert, P., Finkel, A., Wagner, K., eds.: STACS 93. Volume 665 of LNCS. Springer (1993) 58–68
8. Woodcock, J., Davies, J.: Using Z – Specification, Refinement, and Proof. Prentice Hall (1996)
9. Minsky, M.L.: Computation: finite and infinite machines. Prentice-Hall, Inc. (1967)
10. Gabbay, D.M.: Labelled deductive systems. Volume 1. Oxford University Press (1996)
11. Viganò, L.: Labelled Non-Classical Logics. Kluwer Academic Publishers (2000)
12. Dutertre, B.: Complete proof systems for first order interval temporal logic. In: Proceedings of the 10th Annual IEEE Symposium on Logic in Computer Science. LICS '95, Washington, DC, USA, IEEE Computer Society (1995) 36–
13. van Benthem, J., Bezhanishvili, G.: Modal logics of space. In Aiello, M., Pratt-Hartmann, I., Benthem, J., eds.: Handbook of Spatial Logics. Springer (2007) 217–298
14. Randell, D.A., Cui, Z., Cohn, A.G.: A Spatial Logic based on Regions and Connection. In: Proc. 3rd Int'l Conf. on Knowledge Representation and Reasoning. (1992)
15. Gabbay, D., Kurucz, A., Wolter, F., Zakharyaschev, M.: Many-dimensional modal logics: theory and applications. Volume 148 of Studies in Logic and the Foundations of Mathematics. Elsevier (2003)
16. Simpson, A.K.: The Proof Theory and Semantics of Intuitionistic Modal Logic. PhD thesis, University of Edinburgh (1994)
17. Caleiro, C., Sernadas, A., Sernadas, C.: Fibring logics: Past, present and future. In: We Will Show Them! Essays in Honour of Dov Gabbay, Volume 1. (2005) 363–388
18. Rasga, J., Sernadas, A., Sernadas, C., Viganò, L.: Fibring labelled deduction systems. Journal of Logic and Computation **12** (2002) 443–473
19. Paulson, L.: Isabelle: A Generic Theorem Prover. Springer (1994)