



HAL
open science

Symbolic domains and reachability for nets with trajectories

Loïc Hélouët, Prerak Contractor

► **To cite this version:**

Loïc Hélouët, Prerak Contractor. Symbolic domains and reachability for nets with trajectories. 2023. hal-04330097

HAL Id: hal-04330097

<https://inria.hal.science/hal-04330097v1>

Preprint submitted on 7 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Symbolic domains and reachability for nets with trajectories.

Loïc Hélouët  

Univ. Rennes, IRISA, CNRS & INRIA, Rennes, France

Prerak Contractor 

IIT Bombay, Mumbai, India

Abstract

This paper addresses the problem of reachability for timed models handling additional quantities progressing linearly such as distance of moving objects to a target. We first introduce a variant of Petri nets called *trajectory nets* where some places are standard control places containing tokens, and other places contain a simplified representation of trajectories of objects instead of tokens. We give a semantics for this model, and propose an abstraction of sets of equivalent configurations into symbolic domains. We show how to compute in polynomial time a symbolic representation of successors of configurations appearing in a domain, and prove that domains are closed under calculus of successors. Further, the set of domains for a fixed trajectory net is finite. When the control part of the model is bounded, reachability, coverability and safety properties involving distances can be checked in PSPACE on a sound, complete, and finite abstraction called a state class graph.

2012 ACM Subject Classification Timed and hybrid systems

Keywords and phrases Reachability; Concurrency; symbolic domains

Digital Object Identifier 10.4230/LIPIcs...

1 Introduction

Some properties of cyber-physical systems such as transport networks call for the verification of quantitative properties addressing time, but also continuous values such as distances. A typical example is safety of metro networks, where one wants to guarantee safety headways, or bound the number of trains in tunnels to guarantee safe evacuation of passengers in case of power failure. Models such as timed automata [2] or time Petri nets [12] only address time, and cannot be used to handle such problems. Models that can address both time and continuous values such as distances rapidly have the expressive power of hybrid automata [1], for which most problems become undecidable.

This paper introduces *trajectory nets*, a model tailored for the analysis of safety properties of systems involving both time and distances such as metro networks. Trajectory nets are a variant of time Petri nets, where some places are dedicated to control, and other places depict object movements with simplified representations called *trajectories*. Configurations assign an integral number of tokens to control places, and a trajectory, representing the remaining time and distance to the next stop to a subset of trajectory places. Dealing with trajectories allows to define properties that address both distances and time. Verification of a safety property of the form "At each instant, less than K trains are in a tunnel" amounts to a reachability question for sets of configurations depicting forbidden positions of trains.

As a first contribution of this paper, we define trajectory nets and give their semantics in terms of configurations, discrete events (end of progress of a trajectory, creation of a new one) and timed moves. As in many continuous models, the set of possible configurations is infinite. Non-finiteness of the configuration space of a net comes on one hand from the unboundedness of the discrete contents of control places, and on the other hand from the continuous representation of trajectories. We show that in their full generality, trajectory



© L. Hélouët, P. Contractor;

licensed under Creative Commons License CC-BY 4.0

Leibniz International Proceedings in Informatics



LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

45 nets can simulate a two-counters machine, and are hence Turing Powerful. As a consequence,
 46 safety properties relying on coverability or reachability of a configuration are undecidable.

47 As a second contribution of the paper, we show that the continuous part of configurations
 48 can be represented symbolically by sets of linear inequalities called *domains*. Abstracting
 49 time in variants of Petri nets is now a standard approach [3, 11, 10], that can be compared
 50 to regions in timed automata [2]. However, for trajectory nets, domains have to abstract
 51 away two types of continuous values, namely time and distance. We show that we can
 52 compute a successor relation on domains, that domains are closed under this relation, and
 53 that the set of reachable domains of a given trajectory net is finite. A consequence is that,
 54 for trajectory nets with bounded control places, one can compute a sound and complete
 55 symbolic abstraction of the timed behaviour called a *state class graph*, and use it to verify
 56 properties of the original model. A direct consequence of this result is that coverability and
 57 reachability of bounded trajectory nets is PSPACE-complete. Further the safety properties
 58 motivating this work can be decided, in PSPACE.

59 2 Preliminaries

60 In the rest of the paper, we will denote respectively by $\mathbb{R}, \mathbb{Q}, \mathbb{N}$ the sets of reals, rationals, and
 61 non-negative integers. We will denote by $\mathbb{R}^{\geq 0}, \mathbb{Q}^{\geq 0}$ the sets of positive reals and rationals, and
 62 by $\mathcal{I}_{\mathbb{Q}}$ the set of intervals of the form $[a, b]$ or $[a, \infty)$ where $a, b \in \mathbb{Q}$. Let $X = \{x_1, \dots, x_n\}$ be
 63 a set of variables. A *linear constraint* over X with rational coefficients (or simply constraint
 64 for short) is an expression of the form $a_1.x_1 + a_2.x_2 + \dots + a_n.x_n \leq b$, where b is a rational value
 65 and a_i 's are rational coefficients (which can have value 0). A constraint is two-dimensional if
 66 it has at most two variables with non-zero coefficients.

67 A *valuation* for a set of variables X is a map $\mu : X \rightarrow \mathbb{R}$. We will say that a valuation
 68 satisfies a linear constraint $C(X) ::= \sum a_i.x_i \leq b$ iff replacing every x_i by its valuation in
 69 $C(X)$ yields a tautology.

70 A system of linear constraints is a set of linear inequalities. It is two-dimensional iff all
 71 its constraints are two-dimensional. A valuation satisfies a system S iff it satisfies all linear
 72 inequalities in S (i.e. systems are conjunctions of constraints over X). A valuation that
 73 satisfies S is called a *solution* for S . we will denote by $\llbracket S \rrbracket$ the set of solutions for S and say
 74 that S is satisfiable iff $\llbracket S \rrbracket \neq \emptyset$. Slightly abusing our definition, we will sometimes adopt a
 75 more compact representation and write $a \leq expr \leq b$ instead of a conjunction of constraints
 76 of the form $-expr \leq -a$ and $expr \leq b$.

77 Two-dimensional systems can be encoded using Difference Bound Matrices [8] or constraint
 78 graphs, and allow for efficient polynomial algorithms to check satisfiability, compute canonical
 79 forms, intersections... (see [4] for a survey).

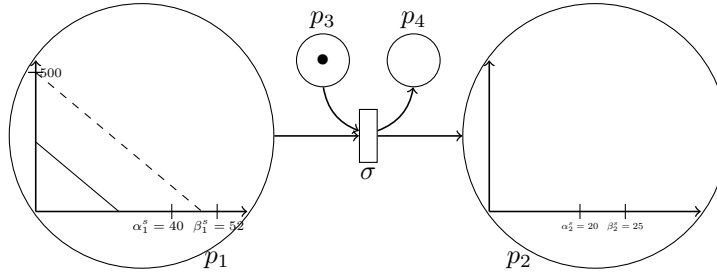
80 3 Trajectory nets

81 ► **Definition 1.** A *Trajectory net* is a tuple $\mathcal{N} = (P, T, F, I, H)$ where $P = P_T \uplus P_C$ is a set of
 82 places. We distinguish a set P_T of trajectory places, and a set P_C of control places. Trajectory
 83 places are holders for train movements, and control places are standard places used to allow
 84 or forbid firing of transitions. $T = \{\sigma_1, \dots, \sigma_{|T|}\}$ is a set of transitions, $F \subseteq P \times T \cup T \times P$
 85 is a flow relation. The function $I : P_T \rightarrow \mathcal{I}_{\mathbb{Q}}$ associates a rational interval $[\alpha_p^s, \beta_p^s]$ to every
 86 trajectory place $p \in P_T$, and the function $H : P_T \rightarrow \mathbb{Q}$ associates a rational distance to every
 87 trajectory place.

88 The flow relation of a trajectory net follows the usual terminology of Petri nets. A relation

89 (p, σ) from a place $p \in P_C$ to a transition σ means that σ needs a token in place p to fire.
 90 Similarly a pair (p, σ) with $p \in P_T$ means that σ can fire only if some trajectory in place
 91 p has reached its final position. On the other hand, a pair $(\sigma, p) \in F$ indicates that firing
 92 of transition σ will produce a fresh token (or a fresh trajectory) in place p . We denote by
 93 $\bullet(\sigma) = \{p \in P \mid (p, \sigma) \in F\}$ the *preset* of σ , i.e. the set of places from which σ consumes
 94 a token or a trajectory when firing, and by $(\sigma)^\bullet = \{p \in P \mid (\sigma, p) \in F\}$ the *postset* of σ ,
 95 i.e., the set of places where tokens or trajectories are added when firing σ . In the rest of
 96 the paper, so simplify semantics, we will consider trajectory nets where $|\bullet(\sigma) \cap P_T| \leq 1$ and
 97 $|(\sigma)^\bullet \cap P_T| \leq 1$.

98 Figure 1 represents the basic ingredient of trajectory net: two trajectory places p_1, p_2 ,
 99 two control places p_3, p_4 , a transition σ . On this example, we have $\bullet(\sigma) = \{p_1, p_3\}$ and
 100 $(\sigma)^\bullet = \{p_2, p_4\}$. Place p_3 contains a token, and place p_1 a trajectory. Trajectories are space-
 101 time diagrams, where abscissae represents time and ordinates distance. In this example,
 102 the object with trajectory in place p_1 started originally at a position $H(p_1) = 500m$ with
 103 a planned trip duration $T_p = 46s$ (this original trajectory is represented by a dashed line)
 104 sampled in interval $[40, 52]$. In the configuration represented in the figure, the object is at
 105 200m from it arrival, with a remaining trip duration of 25s, which is represented by a thick
 106 segment.



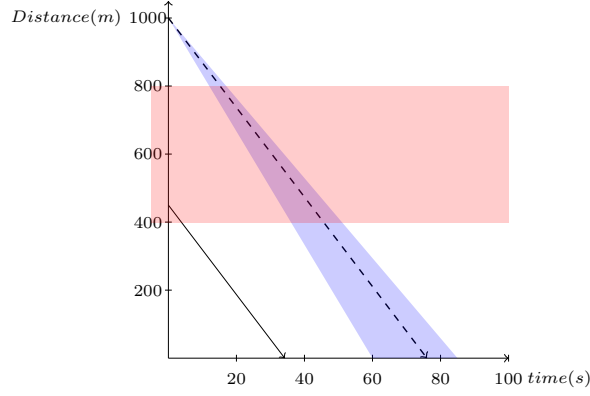
■ **Figure 1** Basic elements of a trajectory net: places, transitions, trajectories.

107 ► **Definition 2.** Let $p \in P_T$ be a trajectory place, $H(p) = H_p^s \in \mathbb{Q}^{\geq 0}$ be a rational value, called
 108 the initial distance of p , and $I(p) = [\alpha_p^s, \beta_p^s]$ be the interval depicting the possible duration of
 109 a movement in place p . A trajectory in p is a pair $tr_p = (T_p, t_p)$, where $T_p \in [\alpha_p^s, \beta_p^s]$ denotes
 110 the initial duration of a movement in the physical space represented by place p and $t_p < T_p$ is
 111 the current remaining trip time in that place. A trajectory tr_p is progressing if $t_p > 0$ and is
 112 blocked otherwise.

113 We denote by $\mathcal{T}r(p)$ the set of all trajectories that may appear in p , that is the set of
 114 all pairs $\mathcal{T}r(p) = \{(T_p, t_p) \mid T_p \in [\alpha_p^s, \beta_p^s] \wedge 0 < t_p < T_p\}$. We assume that represented
 115 objects have a constant speed $v = H(p)/T_p$ during their trip, which allows us to represent
 116 their trajectories as segments. The semantics of a trajectory net is defined in terms of
 117 configurations, that assign an integral number of tokens to control places in T_C , and a
 118 trajectory to a subset of places in P_T . We explicitly differentiate blocked and progressing
 119 trajectories.

120 More formally, a configuration is a pair $C = (M, B, \mathcal{T})$, where $M : P_C \rightarrow \mathbb{N}$ is a marking
 121 of control places, $B \subseteq P_T$ is a subset of trajectory places containing blocked trajectories,
 122 and $\mathcal{T} : P_T \rightarrow \bigcup \mathcal{T}r(p)$ associates a progressing trajectory to a subset of places in P_T .
 123 Given a marking, we will write $M \geq \bullet(\sigma)$ iff $M(p) \geq 1$ for every $p \in \bullet(\sigma) \cap P_C$. We will
 124 denote by $M - \bullet(\sigma)$ the marking M' such that $M'(p) = M(p)$ for every place $p \notin \bullet(\sigma) \cap P_C$

XX:4 Symbolic domains and reachability for nets with trajectories.



■ **Figure 2** A Trajectory (T_p, t_p) in some place p with $H(p) = 1000m$ and $I(p) = [60, 85]$. We have $T_p = 85$ and $t_p = 34$. The blue cone represents all initial trajectories in p for possible initial values for $T_p \in I(p)$. The red area represents a particular critical zone of space (e.g. a tunnel between distance 400 to 800) that needs to be considered for safety.

125 $M'(p) = M(p) - 1$ for every place $p \in \bullet(\sigma) \cap P_C$. We will denote by $M + (\sigma)^\bullet$ the marking
 126 M' such that $M'(p) = M(p)$ for every place $p \notin (\sigma)^\bullet \cap P_C$ and $M'(p) = M(p) + 1$ for every
 127 place $p \in (\sigma)^\bullet \cap P_C$.

128 Then the semantics of a trajectory net can be depicted in terms of timed and discrete
 129 moves from a configuration to the next one. The systems starts in an initial configuration
 130 $C_0 = (M_0, B_0, \mathcal{T}_0)$ such that $B_0 = \emptyset$, and for every p such that \mathcal{T}_0 is defined, $\mathcal{T}_0(p) = (T_p^0, t_p^0)$
 131 with $T_p^0 = t_p^0 \in [\alpha_p^s, \beta_p^s]$. The main idea of the semantics is that a transition can fire if the
 132 objects in the trajectory places of its preset have reached their final destination, and the
 133 control places in the preset allow firing. In other terms, all trajectories in the preset of a
 134 fired transition must be blocked.

135 Upon firing of a transition σ , control tokens are consumed, blocked trajectories are
 136 deleted, and new trajectories and new tokens in control places of $(\sigma)^\bullet$ are created. We adopt
 137 an exclusive semantics w.r.t. trajectory places, i.e. a transition σ can fire only if there
 138 if trajectory places is $(\sigma)^\bullet$ are empty. When modelling metro networks, this semantics is
 139 appropriate to represent a fixed block policy, where track are divided into exclusive blocks
 140 that can contain at most one train at any instant. Upon firing, for each place $p \in (\sigma)^\bullet \cap P_T$
 141 new trajectories are sampled: their initial duration T_p is a random value chosen in $[\alpha_p^s, \beta_p^s]$
 142 and we set $\mathcal{T}(p) = (T_p, T_p)$. On the other hand, elapsing time allows for the progress of
 143 existing trajectories. However, we consider an urgent semantics, that allows elapsing $\delta > 0$
 144 time units in a configuration C only if no discrete firing can occur in C .

145 Discrete moves

146 Discrete moves are either the blocking of a trajectory or the firing of a transition. Blocking
 147 a trajectory $tr_p = (T_p, t_p)$ in place p is possible only if $t_p = 0$, and consists in deleting tr_p , and
 148 adding p to the list of places containing a blocked trajectory. We will say that a transition
 149 σ is fireable in a configuration $C = (M, B, \mathcal{T})$ iff $M \geq \bullet(\sigma)$, the trajectory in $\bullet(\sigma) \cap P_T$ is
 150 blocked, and the place depicting the physical space needed to perform action σ is free, that is,
 151 $\forall p \in (\sigma)^\bullet \cap P_T, p \notin B$ and $\mathcal{T}(p)$ is undefined. The effect of a firing σ is the consumption of
 152 all tokens in $\bullet(\sigma) \cap P_C$, the production of a new token in each place of $(\sigma)^\bullet \cap P_C$, the deletion
 153 of the blocked trajectory in $\bullet(\sigma) \cap B$, and the creation of a new trajectory $\mathcal{T}(p') = (T_{p'}, T_{p'})$
 154 in place $p' \in (\sigma)^\bullet$ with $T_{p'} \in [\alpha_{p'}^s, \beta_{p'}^s]$. We will write $M[\sigma]M'$ when M' is the marking
 155 obtained after firing of σ from M , i.e. when $M' = M - \bullet(\sigma) + (\sigma)^\bullet$.

$$\begin{array}{c}
p \in P_T \\
\mathcal{T}(p) = (x, 0) \text{ for some } x \\
\mathcal{T}'(p_i) = \begin{cases} \mathcal{T}'(p_i) & \text{if } p_i \neq p \\ \text{is undefined} & \text{otherwise} \end{cases} \\
\frac{B' = B \cup \{p\}}{C = (M, B, \mathcal{T}) \xrightarrow{\text{block } p} C' = (M, B', \mathcal{T}')}
\end{array}
\qquad
\begin{array}{c}
M \geq \bullet(\sigma) \wedge M[\sigma]M' \\
\forall p \in (\sigma)^\bullet \cap P_T, p \notin B \wedge \mathcal{T}(p) \text{ is undefined} \\
\bullet(t) \cap P_T \subseteq B \\
B' = B \setminus \bullet(t) \\
\mathcal{T}'(p) = \begin{cases} (T_p, T_p), & \text{with } T_p \in [\alpha_p^s, \beta_p^s] \text{ if } p = (\sigma)^\bullet \\ \mathcal{T}(p) & \text{otherwise} \end{cases} \\
\frac{}{C = (M, B, \mathcal{T}) \xrightarrow{\sigma} C' = (M', B', \mathcal{T}')}
\end{array}$$

156 Timed moves

157 The effect of time elapsing is to reduce the remaining trip time of progressing transitions.
158 Elapsing δ time units is allowed if this duration does not exceed remaining trip time of a
159 progressing trajectory. As in Time Petri nets [12], we adopt an *urgent semantics*, that is we
160 forbid time progress if a discrete event can occur. Time progress of δ is hence forbidden if
161 some transition is firable less than δ time units after the current date, or if a trajectory gets
162 blocked. For a given description of trajectories \mathcal{T} , we denote by $\mathcal{T} + \delta$ the function that
163 associates the pair $(\mathcal{T} + \delta)(p) = (T_p, t_p - \delta)$ to place p if $\mathcal{T}(p) = (T_p, t_p)$.

$$\frac{0 < \delta \leq \min\{t_p \mid (\exists(T_p, t_p) \in \mathcal{T}(P_T))\} \\
\forall \sigma \in T, \sigma \text{ is not firable}}{C = (M, B, \mathcal{T}) \xrightarrow{\delta} C' = (M, B, \mathcal{T} + \delta)}$$

164 Obviously, our semantics rules enjoy time additivity, i.e. if $C_1 \xrightarrow{\delta_1} C_2$ and $C_2 \xrightarrow{\delta_2} C_3$,
165 then $C_1 \xrightarrow{\delta_1 + \delta_2} C_3$. Notice also that timed and discrete moves are exclusive. It is hence natural
166 to describe runs of a trajectory net as an alternation of timed and discrete moves. A *run* of a
167 trajectory net from a configuration $C_0 = (M_0, \mathcal{T}_0)$ is a sequence of timed and discrete moves
168 $\rho = (M_0, B_0, \mathcal{T}_0) \xrightarrow{\delta_0} (M_0, B_0, \mathcal{T}_0 + \delta_0) \xrightarrow{e_1} (M_1, B_1, \mathcal{T}_1) \dots$, where each move of the form
169 $(M_i, B_i, \mathcal{T}_i) \xrightarrow{\delta_i} (M_i, B_i, \mathcal{T}_i + \delta_i)$ is a legal timed move, and $(M_i, B_i, \mathcal{T}_i) \xrightarrow{e_i} (M_{i+1}, B_{i+1}, \mathcal{T}_{i+1})$
170 is a legal discrete move, that is a blocking of a trajectory, $(M_i, B_i, \mathcal{T}_i) \xrightarrow{\text{block } p} (M_{i+1}, B_{i+1}, \mathcal{T}_{i+1})$
171 or a firing of a transition $(M_i, B_i, \mathcal{T}_i) \xrightarrow{\sigma_i} (M_{i+1}, B_{i+1}, \mathcal{T}_{i+1})$.

172 We will write $(M, B, \mathcal{T}) \xrightarrow{*} (M', B', \mathcal{T}')$ if there exists a sequence of discrete and
173 timed moves leading from (M, B, \mathcal{T}) to (M', B', \mathcal{T}') . Without loss of generality, we assume
174 that a net starts in an initial configuration $C_0 = (M_0, B_0, \mathcal{T}_0)$ without blocked transitions,
175 and such that for every $p \in P_T$ where $\mathcal{T}_0(p)$ is defined, we have $\mathcal{T}_0(p) = (T_p^0, T_p^0)$ for
176 some $T_p^0 \in [\alpha_p^s, \beta_p^s]$. We will denote by $\text{Reach}(C_0)$ the set of reachable configurations, i.e.
177 $\text{Reach}(C_0) = \{(M, B, \mathcal{T}) \mid C_0 \xrightarrow{*} (M, B, \mathcal{T})\}$. We will say that a trajectory net is *bounded*
178 iff, there exists an integer K such that for every configuration (M, B, \mathcal{T}) in $\text{Reach}(C_0)$, and
179 for every place $p \in P_C$, $M(p) \leq K$.

180 We will address reachability problems, i.e. study whether a particular configuration
181 (M, B, \mathcal{T}) is reachable. Now, asking whether a configuration (M, B, \mathcal{T}) is reachable refers
182 to the exact position of objects, and is a too precise question. To cope with this problem,
183 we transform contents of trajectory places into markings as follows: given a configuration
184 $C = (M, B, \mathcal{T})$ we define a complete marking M_C that associated an integral number of
185 tokens to each place in P , and such that $M_C(p) = M(p)$ if $p \in P_C$, and $M(p) = 1$ if $p \in P_T$
186 and $\mathcal{T}(p)$ is defined or if $p \in B$. We then differentiate three decision problems:

- 187 ■ *exact reachability*: for a given configuration C , is C a configuration of $\text{Reach}(C_0)$?
- 188 ■ *boolean reachability*: for a given configuration C , is there a configuration $C' \in \text{Reach}(C_0)$
189 such that $M_C = M_{C'}$?

XX:6 Symbolic domains and reachability for nets with trajectories.

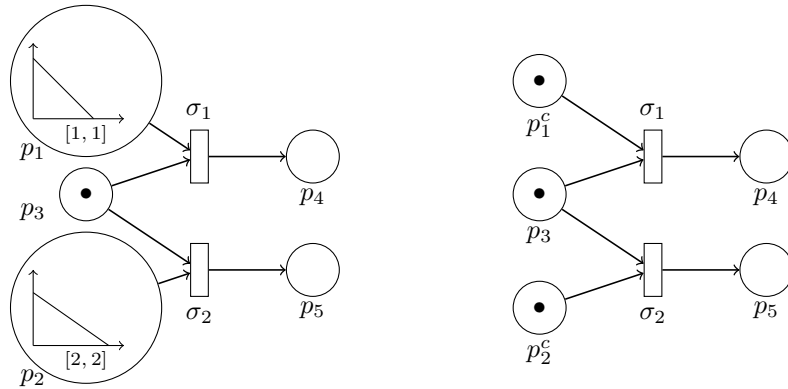
190 ■ *coverability*: for a given configuration C , is there a configuration $C' \in Reach(C_0)$ such
 191 that $M_C \geq M_{C'}$?

192 Coverability can be used to address properties of a subset of objects, and check that they
 193 cannot be at the same moment in critical zones defined by identified places. In section 7, we
 194 will show that we can address finer safety properties involving distances, consider dangerous
 195 zones for trajectories (such as the red zone on the diagram of Figure 2) and forbid situations
 196 where the number of objects positioned in these critical areas exceeds a certain threshold.
 197 Unfortunately, we can show that trajectory nets are powerful enough to model a two counters
 198 machine. We hence have the following result:

199 ► **Theorem 3.** *Reachability, boolean reachability and coverability are undecidable for trajectory*
 200 *nets*

201 **Proof Sketch.** We can simulate the behaviour of an unbounded two counters machine with
 202 an unbounded trajectory net. The encoding is provided in Appendix B. ◀

203 ► **Remark 4.** We can easily transform a trajectory net $\mathcal{N} = (P = P_C \cup P_T, T, F, I, H)$ and its
 204 initial configuration $C_0 = (M_0, B_0, \mathcal{T}_0)$ into a standard untimed Petri net $\mathcal{U}(\mathcal{N}) = (P', T, F)$
 205 where $P' = P_C \cup \{p^c \mid p \in P_T\}$ replaces every trajectory place by a standard place, with
 206 initial marking $M_0^{\mathcal{U}} = M_{C_0}$. It is well known that coverability and reachability are decidable
 207 for Petri nets. This does not contradict Theorem 3, as $\mathcal{U}(\mathcal{N})$ allows more markings and more
 208 runs than \mathcal{N} . Consider the example of Figure 3, with configuration $C_0 = (M_0, B_0 = \emptyset, \mathcal{T}_0)$
 209 with $M_0(p_3) = 1, M_0(p_4) = M_0(p_5) = 0, \mathcal{T}_0(p_1) = (1, 1), \mathcal{T}_0(p_2) = (2, 2)$, and its translation
 210 to a standard Petri net $\mathcal{U}(\mathcal{N})$ on the right of the Figure. One can see from this example that
 211 in the initial configuration C_0 , transition σ_1 is fireable, but transition σ_2 will never fire. On
 212 the other hand, in $\mathcal{U}(\mathcal{N})$, both transitions σ_1 and σ_2 can fire from M_{C_0} .



■ **Figure 3** A trajectory net and its untimed abstraction into a Petri net.

213 A standard way to recover decidability of reachability and coverability in timed extensions
 214 of Petri nets is to restrict to bounded nets, and find an appropriate abstraction of time. One
 215 cannot simply forget timing information (the example in Figure 3 is bounded, but erasing
 216 time makes some markings reachable). For time Petri nets, where time is measured by
 217 clocks attached to enabled transitions, [3] defines an abstraction called state classes, that
 218 are equivalence classes for sets of configurations with identical markings and equivalent
 219 constraints on the values of clocks called domains (and can be compared to regions of timed

220 automata [2]). In the next section, we consider sate classes and domains for trajectory nets
 221 as a sound abstraction of continuous values appearing in configurations of trajectory nets.

222 4 Domains

223 In this section we define domains for trajectory nets. Domains are a way to define symbolically
 224 the value of initial and remaining trajectory durations with positive real valued variables.
 225 For a given configuration $C = (M, B, \mathcal{T})$, we have two types of trajectories: the blocked
 226 trajectories, which remaining running time is know (it is 0), and which initial time is of no
 227 use to define the position of the moving train. The second type of trajectory is the set of
 228 progressing trajectories in places for which $\mathcal{T}(p)$ is defined. For such places, $\mathcal{T}p = (T_p, t_p)$,
 229 and we have two variables of interest: T_p and t_p . Indeed, by keeping T_p , one can compute
 230 the position of a train, which is an important information for safety properties considered in
 231 Section 7.

232 ► **Definition 5 (Domains).** *Let \mathcal{N} be a trajectory net, with set of trajectory places P_T . Let*
 233 *$P \subseteq P_T$ be a subset of P_T representing places with progressing trajectories. Then, a domain for*
 234 *\mathcal{N} with actives trajectories in P is a set of inequalities D over variables $V_D = \{T_i, t_i \mid i \in P\}$,*
 235 *of the form:*

$$236 \quad \alpha_i^1 \leq T_i \leq \beta_i^1 \text{ for all } i \in P \quad (1)$$

$$237 \quad \alpha_i^2 \leq t_i \leq \beta_i^2 \text{ for all } i \in P \quad (2)$$

$$238 \quad t_i - t_j \leq \gamma_{ij}^3 \text{ for all } i \neq j \quad (3)$$

$$239 \quad \alpha_i^4 \leq T_i - t_i \leq \beta_i^4 \text{ for all } i \in P \quad (4)$$

$$240 \quad \alpha_{ij}^5 \leq T_i - t_i + t_j \leq \beta_{ij}^5 \text{ for all } i \neq j \quad (5)$$

$$241 \quad -T_i + t_i + T_j - t_j \leq \gamma_{ij}^6 \text{ for all } i \neq j \quad (6)$$

243 where each inequality appears **exactly** once in D for each $i \in P$ and for each pair of distinct
 244 places $i, j \in P$, and $\alpha_i^1, \alpha_i^2, \alpha_i^4, \alpha_i^5, \beta_i^1, \beta_i^2, \beta_i^4, \beta_i^5, \gamma_{ij}^3, \beta_{ij}^5, \gamma_{ij}^6$ are constant values, or $-\infty, +\infty$.

245 Domains are systems of linear inequalities. We will hence say that a valuation for V_D is a
 246 *solution* for D iff replacing variables T_i, t_i by their value in V_D yields a tautology, and denote
 247 by $\llbracket D \rrbracket$ the set of all solutions for D . Slightly abusing our notation, we will write $\mathcal{T} \in \llbracket D \rrbracket$
 248 when the valuation $\mu_{\mathcal{T}}$ that associates to variables $\{T_i, t_i\}$ their respective values in \mathcal{T} is a
 249 solution of D . We will say that two domains D_1, D_2 are equivalent iff $\llbracket D_1 \rrbracket = \llbracket D_2 \rrbracket$. Even
 250 if two domains are equivalent, they may have different representations. Indeed, consider a
 251 single pair of variables $T_1 = t_1$ which values lay in the interval $[3, 4]$. We can represent the
 252 constraint on T_1, t_1 as $D_1 = \{3 \leq T_1 \leq 12; 0 \leq T_1 - t_1 \leq 0; 0 \leq t_1 \leq 4\}$. However the domain
 253 $D_2 = \{0 \leq T_1 \leq 4; 0 \leq T_1 - t_1 \leq 0; 3 \leq t_1 \leq 20\}$ represents the same set of solutions.

254 ► **Definition 6.** *Let $D = \{a_i \leq \text{expr}_i \leq b_i\}$ be a domain of the form given in Defn 5. Then,*
 255 *the canonical form for D is a domain $D^* = \{a_i^* \leq \text{expr}_i \leq b_i^*\}$, where a_i^* is the smallest*
 256 *value taken by expr_i in $\llbracket D \rrbracket$, and b_i^* is the largest value taken by expr_i in $\llbracket D \rrbracket$.*

257 ► **Proposition 7.** *The canonical form for a domain D is unique and preserves $\llbracket D \rrbracket$.*

258 **Proof.** Let us first show that replacing inequalities $\text{expr} \leq \alpha$ by $\text{expr} \leq \alpha^*$ in D_u does not
 259 affect the solution set. Let D' denote the modified system of inequalities. Suppose that the
 260 set of solutions $\llbracket D' \rrbracket$ differs from $\llbracket D \rrbracket$. We can have two cases:

XX:8 Symbolic domains and reachability for nets with trajectories.

- 261 ■ There exists a feasible solution $X \in \llbracket D' \rrbracket$ but $X \notin \llbracket D \rrbracket$. However, since X satisfies D' ,
 262 the evaluation of expr , say v satisfies $v \leq \alpha^* \leq \alpha$, and hence X also satisfies $\text{expr} \leq \alpha$,
 263 and hence also D (all other inequalities are unmodified, and hence cannot be violated in
 264 D).
- 265 ■ There exists a feasible solution $X \in \llbracket D \rrbracket$ but $X \notin \llbracket D' \rrbracket$. Again, the only inequality which
 266 could have been violated is $\text{expr} \leq \alpha^*$. However then X is a feasible solution of D but
 267 with evaluation of expr greater than α^* , violating the optimality of $\alpha^* = \max_{X \in \llbracket D \rrbracket} \text{expr}$.

268 Hence by contradiction, we can say $\llbracket D' \rrbracket = \llbracket D \rrbracket$

269 Now, all α^* 's and β^* 's in inequalities are uniquely defined by $\llbracket D \rrbracket$, so D^* is unique. ◀

271 As all domains over a fixed set of progressing trajectories the same types of inequalities,
 272 and differ only by the constants used, a direct consequence of proposition 7 is that two
 273 domains D, D' are equal if and only if $D^* = D'^*$.

274 ▶ **Proposition 8.** *The canonical form for a domain D can be computed in PTIME.*

275 **Proof.** We perform the following linear transformation: $x_i = T_i - t_i$ and $y_i = -t_i$ to get a
 276 new set of inequalities:

$$\begin{aligned}
 277 \quad & \alpha_i^1 \leq x_i - y_i \leq \beta_i^1 \\
 278 \quad & -\beta_i^2 \leq y_i \leq -\alpha_i^2 \\
 279 \quad & y_j - y_i \leq \gamma_{ij}^3 \\
 280 \quad & \alpha_i^4 \leq x_i \leq \beta_i^4 \\
 281 \quad & \alpha_{ij}^5 \leq x_i - y_j \leq \beta_{ij}^5 \\
 282 \quad & -x_i + x_j \leq \gamma_{ij}^6 \\
 283
 \end{aligned}$$

284 Notice that our linear transformation is bijective. Hence, for any solution $\mu : \{T_i, t_i\} \rightarrow \mathbb{R}$
 285 in the original domain, there exists a **unique** solution μ' such that $\mu'(x_i) = \mu(T_i) -$
 286 $\mu(t_i)$, $\mu'(y_i) = -\mu(t_i)$ and for any solution $\mu' : \{x_i, y_i\} \rightarrow \mathbb{R}$ in the new domain, there exists
 287 a **unique** solution μ such that $\mu(T_i) = \mu'(x_i) - \mu'(y_i)$, and $\mu(t_i) = -\mu'(y_i)$. Hence there is a
 288 bijection between the two domains.

289 Observe that this new domain is of dimension 2. It can hence be encoded as a DBM
 290 or a constraint graph, and finding a canonical form for this new domain can be done by
 291 computing the shortest paths in the constraint graph, in $O(n^3)$ for n variables. The optimal
 292 bounds obtained for the domain over variables $\{x_i, y_i\}$ are bounds for expressions of the form
 293 $x_i - y_i, x_i - y_j$, etc. that directly encode expressions of the original domain D (for instance
 294 $x_i - y_i = T_i$, and $x_i - y_j = T_i - t_i + t_j$. The sharp bounds obtained for the new domain can
 295 hence be immediately used as optimal bonds for D , except for the expression of the form
 296 $-(\beta_i^2)^* \leq y_i \leq -(\alpha_i^2)^*$, where we need a sign inversion to obtain $(\alpha_i^2)^* \leq t_i \leq (\beta_i^2)^*$.

297 For a domain D addressing properties of k trajectories, the linear transformation of D is
 298 in $O(k^2)$, as we have $3 \cdot k + 3 \cdot k^2$ inequalities in a domain, and performing the transformation
 299 for each inequality takes constant time. Computing the canonical form of the new domain
 300 can be done in $O(k^3)$ time using the Floyd-Warshall algorithm, as the new domain has $2 \cdot k$
 301 variables. Hence, the canonical form can be computed in $O(k^3)$. Also observe that the
 302 constants obtained in the canonical form are linear combinations of α_i^s and β_i^s with integer
 303 coefficients. ◀

Let \mathcal{N} be a trajectory net, and P be the subset of P_T containing progressing trajectories in the initial configuration C_0 . The initial domain D_0 for \mathcal{N} and P is the set:

$$D_0 = \begin{cases} T_i^0 \leq T_i \leq T_i^0 & \text{for all } i \in P \\ T_i^0 \leq t_i \leq T_i^0 & \text{for all } i \in P \\ t_i - t_j \leq \infty & \text{for all } i \neq j \\ 0 \leq T_i - t_i \leq 0 & \text{for all } i \in P \\ -\infty \leq T_i - t_i + t_j \leq \infty & \text{for all } i \neq j \\ -T_i + t_i + T_j - t_j \leq \infty & \text{for all } i \neq j \end{cases}$$

Let μ_0 be the valuation such that $\mu_0(T_p) = \mu_0(t_p) = T_p^0$ for every place where T_0 is defined. Obviously, $\llbracket D_0 \rrbracket = \{\mu_0\}$. The initial domain D_0 meets the requirements of Defn. 5. To show that the form of domain of Defn. 5 is sufficient to represent all domains of a net, it remains to show that the effect of a transition firing, or of a trajectory blocking after some delay δ as proposed in the semantics of Section 3 can be encoded through algebraic operations (variable changes, unions of inequalities and projections) that preserve the types of inequalities considered in Defn. 5.

4.1 Successors after firing a transition

Let D be a domain with set of progressing trajectories P . We want to compute the set of constraints on variables attached to progressing trajectories of the net after firing a transition σ . Let $p = \bullet(\sigma) \cap P_T$ and $p' = \sigma(\sigma) \cap P_T$. First, σ can fire only if p is an empty place (a trajectory in p was formerly blocked), and p' is also empty. According to our semantics, adding a trajectory in p' means sampling a new trip duration $T_{p'} \in [\alpha_{p'}^s, \beta_{p'}^s]$ and adding in p' a new progressing trajectory $(T_{p'}, T_{p'})$. The sampled value is totally independent from the values of variables in D , so the new set of constraint on progressing trajectories after firing of σ is the set:

$$\begin{aligned} SuccF(D, \sigma) = D & \cup \{\alpha_{p'}^s \leq T_{p'} \leq \beta_{p'}^s\} \cup \{\alpha_{p'}^s \leq t_{p'} \leq \beta_{p'}^s\} \cup \{0 \leq T_{p'} - t_{p'} \leq 0\} \\ & \cup \{t_{p'} - t_i \leq \infty \mid i \in P\} \cup \{t_i - t_{p'} \leq \infty \mid i \in P\} \\ & \cup \{-\infty \leq T_{p'} - t_{p'} + t_i \leq \infty \mid i \in P\} \cup \{-\infty \leq T_i - t_i + t_{p'} \leq \infty \mid i \in P\} \\ & \cup \{-T_{p'} + t_{p'} + T_i + t_i \leq \infty \mid i \in P\} \cup \{-T_i + t_i + T_{p'} + t_{p'} \leq \infty \mid i \in P\} \end{aligned}$$

One can immediately notice that if D is a domain, then so is $SuccF(D, \sigma)$.

4.2 Successors after blocking a trajectory

Blocking a progressing trajectory $tr_p = (T_p, t_p)$ from a configuration occurs after elapsing $\delta = t_p$ time units, and if δ is the minimal duration among all progressing trajectories. We hence have to consider transformations on a domain D occurring after a sequence of timed and discrete moves of the form $C \xrightarrow{\delta} C' \xrightarrow{block^p} C''$. Remark, from the semantics, that $\delta = t_p$, so blocking of tr_p can occur only if $t_p = \min\{t_j \mid \exists p_j \in P_T, \mathcal{T}(p_j) = (T_j, t_j)\}$. This requirement can be easily translated into a new constraint: trajectory tr_p can be blocked from some configuration satisfying domain D iff $D^{p \leq *} ::= D \cup \{t_p \leq t_j \mid j \neq p \wedge (T_j, t_j) \text{ is a progressing trajectory}\}$ is satisfiable.

As a blocked trajectory does not constrain any more possible durations of other trajectories, the domain capturing the remaining constraints in configuration C'' is the projection on remaining variables once t_p time units have elapsed. To obtain this set of constraints, we proceed as follows:

XX:10 Symbolic domains and reachability for nets with trajectories.

- 331 ■ We make a variable change. Let t'_j denote a variable representing the new value of
 332 remaining travel time of trajectory j after elapsing t_p time units. Then we have $t'_j = t_j - t_p$.
 333 We hence replace every variable t_j by $t'_j + t_p$ in every inequality of $D^{p \leq *}$. Let us call D'
 334 this new domain.
- 335 ■ We eliminate variables T_p and t_p from domain D' . This elimination can be done using
 336 the well-known Fourier-Motzkin algorithm (see Appendix A and [7]).
- 337 ■ We replace every occurrence of a variable t'_j by an unprimed variable t_j to obtain a
 338 successor domain $SuccB(D, p)$, and we compute its canonical form.

339 ► **Proposition 9.** *Let D be a domain of a trajectory net \mathcal{N} , and let D' be a system of linear*
 340 *inequalities that is a successor of D via construction of $SuccB(D, p)$ or $SuccF(D, \sigma)$. Then*
 341 *D' is a domain of \mathcal{N} .*

342 **Sketch.** $SuccF(D, \sigma)$ trivially satisfies this property, as it only adds constraints of the form
 343 $\alpha_i^S \leq T_i \leq \beta_i^S$, $\alpha_i^S \leq t_i \leq \beta_i^S$, and $T_i = t_i$. The proof for $SuccB(D, p)$ is more involved, as
 344 it requires eliminating variables for the blocked trajectory. Yet, during elimination, some
 345 inequalities are unchanged because they do not refer to T_i not t_i . For other inequalities,
 346 combining expressions of the form $expr_j \leq t_i$ and $t_i \leq expr_k$ to obtain a new expression
 347 $expr_j \leq expr_k$ during the elimination process either produces tautologies, or new expressions
 348 that are of the form of inequalities in Defn 5. A complete proof is given in Appendix C. ◀

5 Soundness of symbolic runs

350 Now the we have defined domains for trajectory nets, and shown that we can effectively
 351 compute a canonical representation for $SuccF(D, \sigma)$ the set of constraints that hold after
 352 firing a transition σ and $SuccB(D, p)$ the constraints that hold after blocking a trajectory in
 353 place p when starting from a domain D , we can define state classes.

354 ► **Definition 10.** *A state class of a trajectory net \mathcal{N} is a pair $SC = (M, B, D)$, where M is*
 355 *a marking, B is a subset of trajectory places with blocked trajectories, and D is a domain of*
 356 *\mathcal{N} in canonical form.*

357 We can define a symbolic transition relation among state classes, setting

- 358 ■ $(M, B, D) \xrightarrow{Block^p_S} (M', B', D')$ if $B' = B \cup \{p\}$, and D' is the canonical representation of
 359 $SuccB(D, p)$, and
- 360 ■ $(M, B, D) \xrightarrow{\sigma_S} (M', B', D')$ if $M[\sigma]M'$, $B' = B \setminus \bullet(\sigma)$, and D' is the canonical representation
 361 of $SuccF(D, \sigma)$.

362 We will write $(M, B, D) \xrightarrow{S} (M', B', D')$ if either $(M, B, D) \xrightarrow{Block^p_S} (M', B', D')$ or
 363 $(M, B, D) \xrightarrow{\sigma_S} (M', B', D')$. We will denote by $Reach^S(M_0, B_0, D_0)$ the set of state classes
 364 that can be built inductively from the initial state class by application of the symbolic
 365 transition relation \xrightarrow{S} .

366 ► **Definition 11.** *The state class graph of a trajectory net \mathcal{N} is the transition system*
 367 *$SC(\mathcal{N}) = (Reach^S(M_0, B_0, D_0), \xrightarrow{S}, (M_0, B_0, D_0))$.*

368 Notice that $SC(\mathcal{N})$ is defined even if $Reach^S(M_0, B_0, D_0)$ is not finite. We will say that
 369 a configuration $C = (M, B, \mathcal{T})$ matches with a state class $SC = (M', B', D)$ iff $M = M'$,
 370 $B = B'$ and $\mathcal{T} \in \llbracket D \rrbracket$.

371 ► **Definition 12.** *A symbolic run of \mathcal{N} is sequence $\rho^S = (M_0, B_0, D_0) \xrightarrow{e_0} (M_1, B_1, D_1) \xrightarrow{e_1}$*
 372 *... such that for every index $i \geq 0$, $e_i \in \{Block^p_i, \sigma_i\}$ and $(M_i, B_i, D_i) \xrightarrow{e_i_S} (M_{i+1}, B_{i+1}, D_{i+1})$.*

373 ► **Proposition 13** (Soundness). *Let $\rho^S = (M_0, B_0, D_0) \xrightarrow{e_0} (M_1, B_1, D_1) \xrightarrow{e_1} \dots$ be a*
 374 *symbolic run of a trajectory \mathcal{N} . Then, there exists a run $\rho = (M_0, B_0, \mathcal{T}_0) \xrightarrow{e_0} (M_1, B_1, \mathcal{T}_1) \xrightarrow{e_1}$*
 375 *\dots of \mathcal{N} such that for every $i \geq 0$, $(M_i, B_i, \mathcal{T}_i)$ matches with (M_i, B_i, D_i) .*

376 ► **Proposition 14** (Completeness). *Let $\rho = (M_0, B_0, \mathcal{T}_0) \xrightarrow{e_0} (M_1, B_1, \mathcal{T}_1) \xrightarrow{e_1} \dots$ be a run of a*
 377 *trajectory net \mathcal{N} . Then, there exists a symbolic run $\rho^S = (M_0, B_0, D_0) \xrightarrow{e_0} (M_1, B_1, D_1) \xrightarrow{e_1}$*
 378 *\dots of \mathcal{N} such that for every $i \geq 0$, $(M_i, B_i, \mathcal{T}_i)$ matches with (M_i, B_i, D_i) .*

379 The proofs for Propositions 13 and 14 are obtained by induction on the length of runs,
 380 and are detailed in Appendix D.

381 6 Finiteness of Domains and State classes

382 ► **Lemma 15.** *Let \mathcal{N} be a trajectory net, and D be a canonical domain computed inductively*
 383 *from the initial domain D_0 of \mathcal{N} . The constants appearing in D are linear combinations of*
 384 *α_i^s , β_i^s and T_i^0 with integer coefficients.*

385 **Proof.** We can reuse the analysis of expressions generated by variable elimination from a
 386 domain D in the proof of Prop. 9 to show that expressions of the form $expr_j + bj \leq expr_k + b_k$
 387 can be equivalently rewritten in an expression of the form $expr_j - expr_k \leq b_k - bj$ and are
 388 hence linear combinations of constants appearing in D . As constants in D_0 are T_i^0 's, and
 389 new constants introduced by successors are α_i^s and β_i^s , we can conclude. ◀

390 ► **Lemma 16** ([3], Lemma 4). *Let A, B be constants, and q_1, \dots, q_n be rational constants. Then*
 391 *there is only a bounded number of linear combinations of q_1, \dots, q_n , with integer coefficients*
 392 *between A and B .*

393 ► **Lemma 17.** *Let \mathcal{N} be a trajectory net, and D be a domain computed inductively from the*
 394 *initial domain D_0 of \mathcal{N} . Let C_{max} be the maximal value appearing in an interval $[\alpha_i^s, \beta_i^s]$.*
 395 *The constants appearing in D are in interval $[-2 \cdot C_{max}, 2 \cdot C_{max}]$*

396 **Proof.** Consider the general form of domains given in Defn. 5. All inequalities are of the form
 397 $\alpha \leq expr \leq \beta$, with $expr = \sum_{i \in S_1} A_i x_i + \sum_{i \in S_2} B_i x_i$ where x_i are variables with bounds
 398 $\alpha_i \leq x_i \leq \beta_i$ and $A_i > 0$ and $B_i < 0$. We can prove (see lemmas 28,29,30 in Appendix E)
 399 that if bounds for all x_i 's exists, then we can derive a bound for $expr$. For trajectory nets,
 400 we immediately have bounds $[\alpha_i^s, \beta_i^s]$ for variable T_i and $[0, \beta_i^s]$ for variable t_i . ◀

401 ► **Theorem 18.** *Let \mathcal{N} be a bounded trajectory net, with initial configuration C_0 . Then the*
 402 *set of canonical domains that can be computed inductively from D_0^* is finite.*

403 **Proof.** A domain is defined by a set of inequalities. The number of inequalities depend
 404 only on the number of progressing trajectories, and these inequalities involve constants. We
 405 know that each constant appearing in this set is bounded as shown in Lemma 17. Hence by
 406 Lemma 16, these constants can take a finite number of values. Hence only a finite number of
 407 inequalities appear in domains. Let us denote by I_0 this set of possible inequalities. The set
 408 of possible domains is finite, since each domain is a subset of I_0 . ◀

409 Following Theorem 18, we can give an upper bound on the number of state classes in
 410 $SC(\mathcal{N})$. Let us first compute the size of I_0 . Assuming that we consider only domains in
 411 canonical form, every constraint in I_0 is of the form $a \leq expr \leq b$, with $-2 \cdot C_{max} \leq a$ and
 412 $b \leq 2 \cdot C_{max}$. Assuming that all α_i^s and β_i^s are rational numbers with a common denominator
 413 D , there exists at most $4 \cdot C_{max} \cdot D$ possibilities for values of a and b in expressions. Similarly,

XX:12 Symbolic domains and reachability for nets with trajectories.

414 expression $expr$ are of the form given in definition 5, and there are hence $3 \cdot (|P_T| + |P_T|^2)$
415 expressions. The size of I_0 is hence $12 \cdot C_{max} \cdot D \cdot (|P_T| + |P_T|^2)$, and each domain is a selection
416 of inequalities from I_0 . This allows us to give an upper bound on the number of domains,
417 which is in $O(2^{|I_0|})$. In a state class, component B is a subset of trajectory places, and
418 hence there are at most 2^{P_T} possible values for B . Last, for a K bounded trajectory net,
419 the number of possible markings for control places is in $O(2^{\log K \cdot |P_C|})$. The number of state
420 classes is hence in $O(2^{|I_0| + |P|})$.

421 **7** Reachability, Coverability, Safety

422 An important property of the state class graph is that all solutions for domains that are
423 reachable in $SC(\mathcal{N})$ are also reachable in \mathcal{N} . This immediately gives an algorithm to check
424 coverability or reachability properties.

425 **► Theorem 19.** *Given a state class (M_n, B_n, D_n) reachable from initial state class (M_0, B_0, D_0) ,*
426 *and a solution $\mathcal{T}_n \in \llbracket D_n \rrbracket$, there exists a run in the original trajectory net that ends in*
427 *configuration $(M_n, B_n, \mathcal{T}_n)$.*

428 **Proof Sketch.** The proof is similar to the proof for soundness (Prop. 13), i.e. uses an
429 induction on the length of path in the state class graph. (See details in Appendix D). ◀

430 **► Theorem 20.** *Reachability, boolean reachability, and boolean coverability are decidable in*
431 *PSPACE for bounded nets*

432 **Proof.** These problems can be solved by a non-deterministic exploration of the state class
433 graph. Let us first consider reachability of a given configuration (M, B, \mathcal{T}) . Assume that a
434 state class (M, B, D) such that $\mathcal{T} \in \llbracket D \rrbracket$ is reachable in $SC(\mathcal{N})$. Then, according to Thm. 19,
435 there exists a run of \mathcal{N} reaching (M, B, \mathcal{T}) . Consider now the boolean reachability and
436 coverability problems. Let $sc = (M, B, D)$ be a reachable state class. One can notice that the
437 boolean marking M_C is identical for every configuration C matching sc . We hence denote
438 this marking by M_{sc} , and compute it by assigning a token to a place $p_i \in P_T$ iff T_i, t_i are
439 variables used in D or if $p_i \in B$. Then, deciding reachability (resp. coverability) of a marking
440 M consists in finding a state class sc such that $M_{sc} = M$ (resp. $M_{sc} \geq M$).

441 As shown in Section 6, the size of the state class graph is exponential w.r.t. the number
442 of places and w.r.t. the value on constants appearing in intervals attached to trajectory
443 places. Encoding a state class can be done in $\log |SC(\mathcal{N})|$ hence in polynomial space, and
444 reachability questions can be addressed in nlogspace w.r.t. the size of the graph. At each
445 step of an exploration reaching a particular state class $SC_i = (M_i, B_i, D_i)$, checking $M = M_i$,
446 $B = B_i$, $M = M_{SC_i}$ or $M \leq M_{SC_i}$ can be done in linear time w.r.t the number of places,
447 and checking $\mathcal{T} \in \llbracket D_i \rrbracket$ can be done in linear time w.r.t the number of inequalities in D by
448 replacing every variable by its value in each inequality. As the number of inequalities in
449 canonical domains is quadratic w.r.t. the number of trajectory places, checking $\mathcal{T} \in \llbracket D_i \rrbracket$
450 can be done in PTIME. Hence, Reachability boolean reachability, and coverability can be
451 checked in NPSPACE, which is equivalent to PSPACE by Savitch's theorem. ◀

452 **► Remark 21.** Notice that a trajectory net without trajectory places is a Petri net (transitions
453 can fire as soon as their preset is filled, after any delay). Hence, reachability and coverability
454 in trajectory nets are at least as hard as reachability and coverability in 1-safe Petri nets.
455 These problems are known to be NP-Hard [5, 9].

456 **► Corollary 22.** *Reachability, boolean reachability, and boolean coverability for bounded nets*
457 *are PSPACE-Complete.*

7.1 Extending coverability to safety properties

Reachability is often a too precise question and one is usually interested in properties that address ranges of values for positions of objects. Let us get back to a particular case study, a metro network. Metro networks can be easily represented by trajectory nets: trajectory places represent track portions between two stations, or a finer partition of a physical network into track portions called *blocks*, transitions symbolize departures or arrivals, Obviously, metro networks have very strict safety requirements that must be guaranteed by physical equipments such as signals and brakes. Safety issues also appear at the operational level. At any instant, evacuation of passengers should be feasible with the lowest risks, and for that reason, operators want to avoid situations where more than K trains are in tunnels or on bridges.

Addressing such safety properties is not a reachability nor a coverability question: it requires that *at any instant*, all trajectories of trains avoid a *set of unsafe positions*. Let $p \in P_T$ be a trajectory place, of length H_p^s , and assume that the track portion represented by p contains a tunnel. We can easily define two values d_p^s and d_p^e defining respectively the position of the entry and exit of the tunnel in that track (for instance, in Figure 2, we have $d_p^s = 400$ and $d_p^e = 800$). Let $\mathcal{T}p = (T_p, t_p)$. The function gives us the initial duration T_p of a trip from a station to the next one, the remaining time before the end of this trip t_p , but we can also compute the position d_p of the considered train on the track. We have assumed that all objects moving in a trajectory net have a constant speed during the whole duration of a trajectory, sampled when the object enters a place. A consequence is that $\frac{H_p^s}{T_p} = \frac{H_p^s - d_p}{t_p}$ at any instant. Hence $d_p = H_p^s \left(1 - \frac{t_p}{T_p}\right)$, and a train in place p is in a tunnel iff the following property is satisfied:

$$Tunnel(p) ::= d_p^s \leq H_p^s \left(1 - \frac{t_p}{T_p}\right) \leq d_p^e$$

Now assume that we want to avoid having trains in places $p_1, \dots, p_k \in P_T$ tunnels at the same instant. It means that we have to avoid any configuration that satisfies the property $Unsafe(p_1, \dots, p_k) ::= \bigcup_{i \in 1..k} Tunnel(i)$.

The domains computed in the state class graph $SC(\mathcal{N})$ are symbolic representations of configurations reached immediately after discrete moves. It can be the case that, after each discrete move, all trains are located before a tunnel in their respective track segment. Verifying safety of a train network does not amount to verifying that $D \cap unsafe(p_1, \dots, p_k)$ for all subsets of places containing tunnels. We need to consider how D evolves when elapsing time, i.e., build symbolic representations of configurations reached an arbitrary duration after a discrete move. Hence, we introduce time closure (M, B, D_\downarrow) of a state class (M, B, D) .

► **Definition 23 (Time Closure).** Let $S = (M, B, D)$ be any state class having a set of active trajectories $P \subseteq P_T$. We introduce variables δ (to represent the timed move) and t'_i for all $i \in P$ (to represent time remaining in trajectories after a timed move of duration δ). The time closure is then defined as a 3-tuple $S_\downarrow = (M, B, D_\downarrow)$ with D_\downarrow defined as:

Case I: No transition is firable from the given marking M and set of blocked trajectories B , and hence timed moves are allowed by the semantics of trajectory nets. We have $D_\downarrow = D \cup \{0 \leq \delta \leq t_i \mid i \in P\} \cup \{t'_i = t_i - \delta \mid i \in P_k\}$

Case II: There exists a firable transition for the given marking M and set of blocked trajectories B , and hence timed moves are not allowed. We hence have $D_\downarrow = D \cup \{\delta = 0\} \cup \{t'_i = t_i \mid i \in T_k\}$.

XX:14 Symbolic domains and reachability for nets with trajectories.

501 The time closure of a state classes $S = (M, B, D)$ is a symbolic representation of
 502 possible configurations reachable after timed moves of arbitrary duration δ , including the
 503 configurations in domain D (i.e., when $\delta = 0$). As explained above, a property of interest
 504 for metro networks is that no more than K trains are in a tunnel at any given instant. A
 505 state class $S = (M, B, D)$ is K -safe for places p_1, \dots, p_K iff $\llbracket D_\downarrow \cup \text{Unsafe}(p_1, \dots, p_K) \rrbracket = \emptyset$.
 506 Verifying that a state class is K -safe for every subset of places of size k amounts to asking
 507 that $\llbracket D_\downarrow \cup \text{Unsafe}(X) \rrbracket = \emptyset$ for every subset $X \subseteq P_T$ of places of size K containing tunnels.
 508 Notice that the set of all X 's can be enumerated in $\log(|P_T|)$ space.

509 ► **Remark 24.** Non-emptiness of $D_\downarrow \cap \text{Unsafe}(p_1, \dots, p_K)$ implies existence of a configuration
 510 violating the safety property, because all configurations in a state class D are reachable, and
 511 hence all configurations in D_\downarrow too. Hence, checking safety for all state classes of $SC(\mathcal{N})$
 512 guarantees that the trajectory net does not violate the safety property. This gives us a
 513 method to check safety of a metro network modelled with a trajectory net using its state
 514 class graph.

515 The constraint $\text{Unsafe}(p_1, \dots, p_k)$ can be rewritten as $\left\{ d_i^s \leq H_i^s (1 - t'_i/T_i) \leq d_i^f \mid i \in T_k \right\}$
 516 and as every T_i is a positive value, simplified to get $\left\{ d_i^s T_i \leq H_i^s (T_i - t'_i) \leq d_i^f T_i \mid i \in T_k \right\}$.
 517 One can immediately observe that this set contains only linear inequalities of dimension 2
 518 involving T_i and t'_i . Let us now consider D_\downarrow , obtained by replacing t_i by $t'_i + \delta$. It is a set of
 519 constraints of the form:

$$\begin{aligned} 520 & \alpha_i^1 \leq T_i \leq \beta_i^1 \text{ for all } i \in P \\ 521 & \alpha_i^2 \leq t'_i + \delta \leq \beta_i^2 \text{ for all } i \in P \\ 522 & t'_i - t'_j \leq \gamma_{ij}^3 \text{ for all } i \neq j \\ 523 & \alpha_i^4 \leq T_i - t'_i + \delta \leq \beta_i^4 \text{ for all } i \in P \\ 524 & \alpha_{ij}^5 \leq T_i - t'_i + t'_j \leq \beta_{ij}^5 \text{ for all } i \neq j \\ 525 & -T_i + t'_i + T_j - t'_j \leq \gamma_{ij}^6 \text{ for all } i \neq j \end{aligned}$$

527 ► **Remark 25.** Checking emptiness of $D_\downarrow \uplus \left\{ d_i^s T_i \leq H_i^s (T_i - t'_i) \leq d_i^f T_i \mid i \in T_k \right\}$ can be
 528 done by elimination of variables one after another, and stopping as soon as an inequality
 529 is unsatisfiable, or when all variables are eliminated. We can use the same variable change
 530 as in Proposition 8, and obtain an equivalent system of inequalities of dimension 2. Hence,
 531 checking satisfiability of $D_\downarrow \uplus \text{Unsafe}(p_1, \dots, p_k)$ can be done in PTIME.

532 ► **Proposition 26.** *Checking a safety property for bounded trajectory nets is in PSPACE.*

533 **Proof.** \mathcal{N} violates a safety property of the form "no more than k trains in a tunnel" iff there
 534 exists a reachable configuration $C = (M, B, \mathcal{T})$ such that $\mathcal{T} \models \text{Unsafe}(p_1, \dots, p_k)$ for some
 535 subset of places p_1, \dots, p_k . According to remark 24, this holds only if there exists a reachable
 536 state class $S = (M, B, D)$ such that $D_\downarrow \cup \text{Unsafe}(p_1, \dots, p_k) \neq \emptyset$

537 We know from the decision procedures for reachability and coverability that exploration
 538 of all state classes can be done in PSPACE. Then, for each state class $S = (M, B, D)$ reached,
 539 we need to compute D_\downarrow enumerate all subsets X of k places containing tunnels and with an
 540 active trajectory, and check emptiness of $D_\downarrow \cup \text{Unsafe}(p_1, \dots, p_k)$. Enumeration of subsets
 541 of places of size k can be done in $\log(P_T)$ space. Then, following remark 25, we know that
 542 (un)satisfiability of $D_\downarrow \cup \text{Unsafe}(p_1, \dots, p_k)$ can be verified in PSPACE. ◀

8 Conclusion

We have considered an extension of Petri nets enhanced with time and linear functions depicting trajectories of moving objects. Most problems for this model are undecidable in general. However, one can decouple the continuous and the control part of the net. As soon as the control part is bounded, the behaviour of the model can be abstracted to a finite state class graph, and coverability, reachability, and safety properties addressing distance issues can be decided in PSPACE. Finiteness of the state class graph of trajectory nets comes from bounds on the values of variables, and from the particular structure of domains, that are conjunctions of linear inequalities with at most 4 variables, and coefficients in $\{1, -1\}$. This structure is preserved by projection, and hence by the successor relation among state classes.

As future work, several extensions of the model can be considered. First of all, trajectories of moving objects are simple linear functions, i.e. the speed of an object is sampled once, and remains constant until a trajectory gets blocked. A possible extension is to consider trajectories of objects with acceleration and braking phases, that are better described by polynomial functions. Another limitation is that the semantics of the model assigns at most one trajectory per place. A sensible extension is to allow several objects to share a physical space, for instance by considering safety headways that have to be kept among objects. This is for instance needed to model road traffic. First experiments seem to show that domains for these extensions cannot be defined with linear inequalities, but need polynomial inequalities, i.e. expressions of the form $P(X) \leq c$, where $P(X)$ is a multivariate polynomial. It is known that such domains are closed under projection [13]. However, elimination requires a doubly exponential complexity [6]. Further, we conjecture that finiteness of domains does not hold any more.

References

- 1 R. Alur, C. Courcoubetis, N. Halbwachs, T.A. Henzinger, P.-H. Ho, X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine. The algorithmic analysis of hybrid systems. *TCS*, 138(1):3–34, 1995.
- 2 R. Alur and D.L. Dill. A theory of timed automata. *TCS*, 126(2):183–235, 1994.
- 3 B. Berthomieu and M. Diaz. Modeling and verification of time dependent systems using time Petri nets. *IEEE Trans. Software Eng.*, 17(3):259–273, 1991.
- 4 V. Chandru. Variable elimination in linear constraints. *Comput. J.*, 36(5):463–472, 1993.
- 5 A. Cheng, J. Esparza, and J. Palsberg. Complexity results for 1-safe nets. *TCS*, 147(1&2):117–136, 1995.
- 6 G. E. Collins. Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In *Automata Theory and Formal Languages*, pages 134–183, 1975.
- 7 G. Dantzig and B. C. Eaves. Fourier-Motzkin Elimination and Its Dual. *J. Comb. Theory, Ser. A*, 14(3):288–297, 1973.
- 8 D.L. Dill. Timing assumptions and verification of finite-state concurrent systems. In *Automatic Verification Methods for Finite State Systems*, volume 407 of *LNCS*, pages 197–212, 1989.
- 9 J. Esparza. Decidability and complexity of Petri net problems - an introduction. In *Proc. of Petri Nets*, volume 1491 of *LNCS*, pages 374–428, 1998.
- 10 L. Hélouët and P. Agrawal. Waiting nets. In *Proc. of PETRI NETS 2022*, volume 13288 of *LNCS*, pages 67–89, 2022.
- 11 D. Lime and O.H. Roux. Model checking of time Petri nets using the state class timed automaton. *Discret. Event Dyn. Syst.*, 16(2):179–205, 2006.
- 12 P. M. Merlin. *A Study of the Recoverability of Computing Systems*. PhD thesis, University of California, Irvine, CA, USA, 1974.
- 13 A. Tarski. A decision method for elementary algebra and geometry. Technical report, RAND Corporation, 1957.

591 **A** Fourier-Motzkin elimination

592 Fourier-Motzkin Elimination [7] is a method to eliminate a set of variables $V \subseteq X$ from
 593 a system of linear inequalities over X . Elimination produces another system of linear
 594 inequalities over $X \setminus V$, such that both systems have the same solutions over the remaining
 595 variables. Elimination can be done by removing one variable from V after another.

596 Let $X = \{x_1, \dots, x_r\}$ be a set of variables, and w.l.o.g., let us assume that x_r is the
 597 variable to eliminate in m inequalities. All inequalities are of the form

$$c_1.x_1 + c_2.x_2 + \dots + c_r.x_r \leq d_i$$

598 where c_j 's and d_i are rational values, or equivalently $c_r.x_r \leq d_i - (c_1.x_1 + c_2.x_2 + \dots +$
 599 $c_{r-1}.x_{r-1})$

600 If c_r is a negative coefficient, the inequality can be rewritten as $x_r \geq b_i - (a_{i,1}.x_1 +$
 601 $a_{i,2}.x_2 + \dots + a_{i,r-1}.x_{r-1})$, and if c_r is positive, the inequality rewrites as $x_r \leq b_i - (a_{i,1}.x_1 +$
 602 $a_{i,2}.x_2 + \dots + a_{i,r-1}.x_{r-1})$, where $b_i = \frac{d_i}{c_r}$ and $a_i = \frac{c_i}{c_r}$.

603 We can partition our set of inequalities as follows.

- 604 ■ inequalities of the form $x_r \geq b_i - \sum_{k=1}^{r-1} a_{ik}x_k$; denote these by
 605 $x_r \geq A_j(x_1, \dots, x_{r-1})$ (or simply $x_r \geq A_j$ for short), for j ranging from 1 to n_A where
 606 n_A is the number of such inequalities;
- 607 ■ inequalities of the form $x_r \leq b_i - \sum_{k=1}^{r-1} a_{ik}x_k$; denote these by
 608 $x_r \leq B_j(x_1, \dots, x_{r-1})$ (or simply $x_r \leq B_j$ for short), for j ranging from 1 to n_B where
 609 n_B is the number of such inequalities;
- 610 ■ inequalities $\phi_1, \dots, \phi_{m-(n_A+n_B)}$ in which x_r plays no role.

611 The original system is thus equivalent to:

$$612 \max(A_1, \dots, A_{n_A}) \leq x_r \leq \min(B_1, \dots, B_{n_B}) \wedge \bigwedge_{i \in 1..m-(n_A+n_B)} \phi_i.$$

613 One can find a value for x_r in a system of the form $a \leq x \leq b$ iff $a \leq b$. Hence, the above
 614 formula is equivalent to:

$$615 \max(A_1, \dots, A_{n_A}) \leq \min(B_1, \dots, B_{n_B}) \wedge \bigwedge_{i \in 1..m-(n_A+n_B)} \phi_i$$

616 Now, this inequality can be rewritten as system of $n_A \times n_B + m - (n_A + n_B)$ inequalities
 617 $\{A_i \leq B_j \mid i \in 1..n_A, j \in 1..n_B\} \cup \{\phi_i \mid i \in 1..m - (n_A + n_B)\}$, that does not contain x_r and
 618 is satisfiable iff the original system is satisfiable.

619 ► **Remark 27.** The Fourier-Motzkin elimination preserves finiteness and satisfiability of a
 620 system of constraints. In general, the number of inequalities can grow in a quadratic way at
 621 each variable elimination. In the case of trajectory nets, where domains are in canonical form,
 622 they always contain less than $2 \cdot |T|^2 + 2 \cdot |T|$ inequalities, and then elimination produces a
 623 system of at most $2 \cdot |T|^2 + 2 \cdot |T|$ inequalities once useless inequalities have been removed.

624 **B** Undecidability

625 **Theorem 3** Reachability, boolean reachability and coverability are undecidable for trajectory
 626 nets

627 **Proof.** We can simulate the behavior of an unbounded two counters machine with an
 628 unbounded trajectory net.

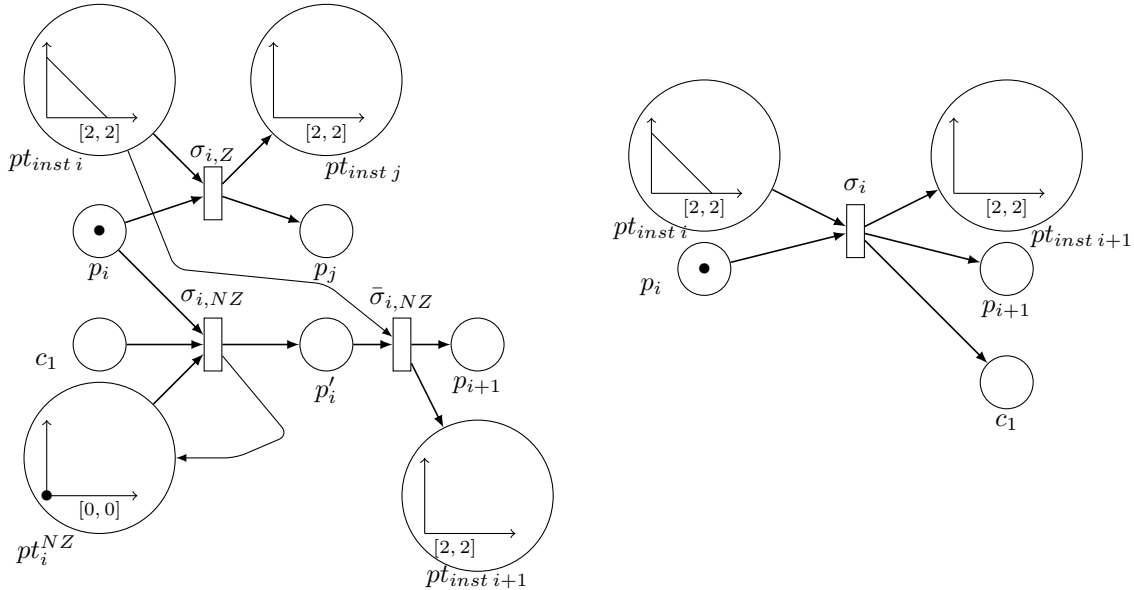
629 A two-counter machine is specified with two counters C_1, C_2 that remember positive
 630 integral values, and a list of instructions $Inst_1, Inst_2, \dots, Inst_m$. These instructions are of
 631 the form :

- 632 ■ $Inst_i : Inc(C_n)$, which effect is to increment counter C_n , and then move to instruction
 633 $i + 1$
 634 ■ $Inst_i : If C_n > 0 Dec(C_n) else Inst_k$, that tests the value of counter C_n , decrements it
 635 and moves to the next instruction if $C_n > 0$, or moves to instruction $Inst_k$ otherwise.
 636 ■ $Halt$, that stops the computation of the counter machine.

637 A configuration of a counter machine is a triple (i, c_1, c_2) representing the current
 638 instruction to execute, and the values of counters c_1, c_2 . It is well known that counter
 639 machines are sufficient to encode Turing Machines, and hence that the question of whether a
 640 machine starting at instruction $Inst_1$ with counter values $c_1 = 0, c_2 = 0$ eventually halts is
 641 undecidable.

642 Consider an arbitrary counter machine \mathcal{M} with counters C_1, C_2 , and a list of instructions
 643 $Inst_1, Inst_2, \dots, Inst_m$. We build a trajectory net $\mathcal{N}_{\mathcal{M}} = (P, T, I)$ that contains

- 644 ■ Two control places p_{c1}, p_{c2} (one per counter), and one trajectory place pt_{inst_i} with
 645 $I(pt_{inst_i}) = [2, 2]$ for each instruction $Inst_i$. One control place p_i per instruction $Inst_i$,
 646 and an additional place p'_i if instruction i is a decrement instruction.
 647 ■ one transition $\sigma_{i,inc(n)}$ for each instruction of the form $Inst_i = Inc(C_n)$, such that
 648 $\bullet(\sigma_{i,inc(n)}) = \{pt_{inst_i}, p_i\}$ $(\sigma_{i,inc(n)})^\bullet = \{pt_{inst_{i+1}}, p_{c_n}, p_{i+1}\}$
 649 ■ three transition $\sigma_{i,Z}, \sigma_{i,NZ}, \bar{\sigma}_{i,NZ}$ for each decrement instruction of the form $If C_n >$
 650 $0 Dec(C_n) else Inst_j$, with $\bullet(\sigma_{i,Z}) = \{pt_{inst_i}, p_i\}$ $(\sigma_{i,Z})^\bullet = \{pt_{inst_j}, p_j\}$ $\bullet(\sigma_{i,NZ}) =$
 651 $\{pt_{inst_i}, p_i\}$ $(\sigma_{i,NZ})^\bullet = \{p'_i\}$ $\bullet(\bar{\sigma}_{i,NZ}) = \{pt_{inst_i}, p'_i\}$ $(\bar{\sigma}_{i,NZ})^\bullet = \{pt_{inst_{i+1}}, p_{i+1}\}$



■ **Figure 4** Encoding instructions of a counter Machine with a trajectory net.

652 The trajectory net assembled this way encodes step of the counter machine as follows: At
 653 every instant, to simulate execution of instruction i , all tokens are located in places p_i, p'_i (for
 654 a single i) and the value of counter C_n is stored in place p_{c_n} . A configuration $C = (M, B, \mathcal{T})$
 655 with p_i marked and $\mathcal{T}(pt_{inst_i})$ where $Inst_i$ is an increment of counter C_n encodes a state
 656 of the machine reaching instruction i . From this configuration, time can elapse up to the
 657 maximal amount of time allowed by $\mathcal{T}(p_{inst_i})$, then trajectory tr_i is blocked, and transition σ_i
 658 fires. As a consequence, place $pt_{inst_{i+1}}$ is filled with a new trajectory, $M(p_{c_n})$ is incremented,

XX:18 Symbolic domains and reachability for nets with trajectories.

659 and place p_{i+1} receives a token. The new contents of $pt_{inst\ i+1}$, p_{i+1} , and p_{c_n} simulate the
660 next configuration of the counter machine after execution of instruction $Inst_i$. Similarly, let
661 us consider a configuration $C = (M, B, \mathcal{T})$ with p_i marked and $\mathcal{T}(pt_{inst\ i}$ and where $Inst_i$
662 is an decrement of counter C_n . From this configuration, two distinct scenarios can occur.
663 If p_{c_n} is empty, then σ_i^{NZ} cannot fire, and hence the maximal amount of time allowed by
664 $\mathcal{T}(p_{inst\ i}$ elapses, then trajectory tr_i is blocked, and transition σ_i^Z fires. As a consequence,
665 place $pt_{inst\ j}$ is filled with a new trajectory, and place p_j receives a token. The new contents
666 of $pt_{inst\ j}$, p_j , and p_{c_n} simulate the next configuration of the counter machine after execution
667 of instruction $Inst_i$ when counter C_n is equal to 0. conversely, assume that p_{c_n} holds m
668 tokens. Then the firing of σ_i^{NZ} is urgent, and hence the marking of p_{c_n} is decremented, and
669 a token is moves from p_i to p'_i . A new trajectory of duration 0 is put in place pt_i^{NZ} for use
670 at the next occurrence of instruction $Inst_i$. The last step of simulation of the decrement
671 instruction is to block and consume the contents of $p_{inst\ i}$ via transition $\bar{\sigma}_i^{NZ}$, and produce a
672 new trajectory in place $p_{inst\ i+1}$ and a token in place p_{i+1} . One can easily see that for every
673 sequence of configurations of a counter machine \mathcal{M} , there exists a run of $\mathcal{N}_{\mathcal{M}}$ such that the
674 sequences of indexes of marked instruction places and the markings of p_{c_1}, p_{c_2} is exactly the
675 run \mathcal{M} . Conversely, for every run of $\mathcal{N}_{\mathcal{M}}$, the sequences of indexes of marked instruction
676 places and the markings of p_{c_1}, p_{c_2} coincide with a run of machine \mathcal{M} . As a consequence,
677 if $Inst_m$ is a halting instruction, then one cannot decide in general whether some marking
678 with $M(p_m) \geq 1$ is reachable. Coverability is hence undecidable. The result easily extends
679 to reachability questions, as one can add to $\mathcal{N}_{\mathcal{M}}$ a gadget that consumes the contents of
680 counter places and of $pt_{inst\ m}$, and ask whether marking M_{halt} such that $M_{halt}(p_m) = 1$ and
681 all other places are empty is reachable. ◀

C Closure of Domains

682 **Proposition 9** Let D be a domain of a trajectory net \mathcal{N} , and let D' be a system of linear
683 inequalities that is a successor of D via construction of $SuccB(D, p)$ or $SuccF(D, t)$. Then
684 D' is a domain of \mathcal{N} .
685

686 **Proof.** The initial domain satisfies the lemma. When a new trajectory is added in a place,
687 then the variables T_i and t_i meet the constraint $\alpha_i^s \leq T_i \leq$ and $\alpha_i^s \leq T_i \leq$. So in the original
688 domain reached before firing a transition has constants that are linear combinations of α_i^s
689 and β_i^s , then so has the newly computed domain in $SuccF(D, t_i)$ for every transition t_i .

690 It now remains to show that the elimination of variables preserves the property. Assume
691 that we want to eliminate variable t_i in a domain $D \cup \{t_i \leq t_j \mid j \neq i\}$, and obtain a new
692 domain over variables $\{T_j, t'_j\}$, where t'_j is the remaining travel time for a trajectory. We
693 have $t'_j = t_j - t_i$ so we do a variable change of the form $t_j - > t'_j + t_i$ in domain D before
694 starting elimination

D , with the constraint that $t_i \leq t_j$ can be written as

$$\begin{aligned}
 & \alpha_i^2 \leq t_i \leq \beta_i^2 \\
 & 0 \leq t_j - t_i && \text{for all } j \\
 & t_i - t_j \leq \gamma_{ij}^3 && \text{for all } i \neq j \\
 & t_j - t_i \leq \gamma_{ji}^3 && \text{for all } i \neq j \\
 D \setminus \{t_i\} \cup & \alpha_i^4 \leq T_i - t_i \leq \beta_i^4 \\
 & \alpha_{ij}^5 \leq T_i - t_i + t_j \leq \beta_{ij}^5 && \text{for all } i \neq j \\
 & \alpha_{ji}^5 \leq T_j - t_j + t_i \leq \beta_{ji}^5 && \text{for all } i \neq j \\
 & -T_i + t_i + T_j - t_j \leq \gamma_{ij}^6 && \text{for all } i \neq j \\
 & -T_j + t_j + T_i - t_i \leq \gamma_{ji}^6 && \text{for all } i \neq j
 \end{aligned}$$

695 where $D_{\setminus\{t_i\}}$ is the set of all inequalities of D that do not contain t_i . With the variable
 696 change, we get

$$\begin{aligned}
 & \alpha_i^2 \leq t_i \leq \beta_i^2 && && \\
 & 0 \leq (t'_j + t_i) - t_i && && \text{for all } j \\
 & t_i - (t'_j + t_i) \leq \gamma_{ij}^3 && && \text{for all } i \neq j \\
 & t'_j + t_i - t_i \leq \gamma_{ji}^3 && && \text{for all } i \neq j \\
 D' \cup & \alpha_i^4 \leq T_i - t_i \leq \beta_i^4 && && \\
 & \alpha_{ij}^5 \leq T_i - t_i + t'_j + t_i \leq \beta_{ij}^5 && && \text{for all } i \neq j \\
 & \alpha_{ji}^5 \leq T_j - (t'_j + t_i) + t_i \leq \beta_{ji}^5 && && \text{for all } i \neq j \\
 & -T_i + t_i + T_j - (t'_j + t_i) \leq \gamma_{ij}^6 && && \text{for all } i \neq j \\
 & -T_j + t'_j + t_i + T_i - t_i \leq \gamma_{ji}^6 && && \text{for all } i \neq j
 \end{aligned}$$

697 where D' is the domain obtained by replacing every t_j by $t'_j + t_i$

698 This system can be rewritten as

$$\begin{aligned}
 & \alpha_i^2 \leq t_i \leq \beta_i^2 && && \\
 & 0 \leq t'_j && && \text{for all } j \neq i \\
 & -t'_j \leq \gamma_{ij}^3 && && \text{for all } i \neq j \\
 & t'_j \leq \gamma_{ji}^3 && && \text{for all } i \neq j \\
 D' \cup & \alpha_i^4 \leq T_i - t_i \leq \beta_i^4 && && \\
 & \alpha_{ij}^5 \leq T_i + t'_j \leq \beta_{ij}^5 && && \text{for all } i \neq j \\
 & \alpha_{ji}^5 \leq T_j - t'_j \leq \beta_{ji}^5 && && \text{for all } i \neq j \\
 & -T_i + T_j - t'_j \leq \gamma_{ij}^6 && && \text{for all } i \neq j \\
 & -T_j + t'_j + T_i \leq \gamma_{ji}^6 && && \text{for all } i \neq j
 \end{aligned}$$

699 We hence obtain a domain of the form

$$\begin{aligned}
 & \alpha_k^2 \leq t'_k + t_i \leq \beta_k^2 && && \text{for all } k \neq i \\
 & t'_k - t'_j \leq \gamma_{kj}^3 && && \text{for all } k \neq j \neq i \\
 & \alpha_k^4 \leq T_k - t'_k - t_i \leq \beta_k^4 && && \text{for all } k \neq i \\
 & \alpha_{kj}^5 \leq T_k - t'_k + t'_j \leq \beta_{kj}^5 && && \text{for all } k \neq j \neq i \\
 & -T_k + t'_k + T_j - t'_j \leq \gamma_{kj}^6 && && \text{for all } k \neq j \neq i
 \end{aligned}$$

$$\begin{aligned}
 700 & \alpha_i^2 \leq t_i \leq \beta_i^2 && && \\
 & 0 \leq t'_j && && \text{for all } j \neq i \\
 & -t'_j \leq \gamma_{ij}^3 && && \text{for all } i \neq j \\
 & t'_j \leq \gamma_{ji}^3 && && \text{for all } i \neq j \\
 & \alpha_i^4 \leq T_i - t_i \leq \beta_i^4 && && \\
 & \alpha_{ij}^5 \leq T_i + t'_j \leq \beta_{ij}^5 && && \text{for all } i \neq j \\
 & \alpha_{ji}^5 \leq T_j - t'_j \leq \beta_{ji}^5 && && \text{for all } i \neq j \\
 & -T_i + T_j - t'_j \leq \gamma_{ij}^6 && && \text{for all } i \neq j \\
 & -T_j + t'_j + T_i \leq \gamma_{ji}^6 && && \text{for all } i \neq j
 \end{aligned}$$

701 We can hence isolate inequalities that do not refer to T_i nor t_i in a set D'' , Identity
 702 a set D_{T_i} of inequalities that refer to T_i , but not to t_i , and a set D_{t_i} that contain all lines
 703 with t_i . We have:

$$704 \quad D'' = D' \cup \begin{cases} t'_k - t'_j \leq \gamma_{kj}^3 & \text{for all } k \neq j \neq i \\ \alpha_{kj}^5 \leq T_k - t'_k + t'_j \leq \beta_{kj}^5 & \text{for all } k \neq j \neq i \\ -T_k + t'_k + T_j - t'_j \leq \gamma_{kj}^6 & \text{for all } k \neq j \neq i \\ \\ 0 \leq t'_j & \text{for all } j \neq i \\ -t'_j \leq \gamma_{ij}^3 & \text{for all } i \neq j \\ t'_j \leq \gamma_{ji}^3 & \text{for all } i \neq j \\ \alpha_{ji}^5 \leq T_j - t'_j \leq \beta_{ji}^5 & \text{for all } i \neq j \end{cases}$$

705 then

$$706 \quad D_{T_i} = \begin{cases} \alpha_{ij}^5 \leq T_i + t'_j \leq \beta_{ij}^5 & \text{for all } i \neq j \\ -T_i + T_j - t'_j \leq \gamma_{ij}^6 & \text{for all } i \neq j \\ -T_j + t'_j + T_i \leq \gamma_{ji}^6 & \text{for all } i \neq j \end{cases}$$

708 and

$$709 \quad D_{t_i} = \begin{cases} \alpha_k^2 \leq t'_k + t_i \leq \beta_k^2 & \text{for all } k \neq i \\ \alpha_k^4 \leq T_k - t'_k - t_i \leq \beta_k^4 & \text{for all } k \neq i \\ \alpha_i^2 \leq t_i \leq \beta_i^2 \\ \alpha_i^4 \leq T_i - t_i \leq \beta_i^4 \end{cases}$$

710 Eliminating t_i then consists in eliminating t_i from D_{t_i}

$$711 \quad D_{t_i} = \begin{cases} \alpha_k^2 - t'_k \leq t_i \leq \beta_k^2 - t'_k & \text{for all } k \neq i \\ \alpha_k^4 - T_k + t'_k \leq -t_i \leq \beta_k^4 - T_k + t'_k & \text{for all } k \neq i \\ \alpha_i^2 \leq t_i \leq \beta_i^2 \\ \alpha_i^4 - T_i \leq -t_i \leq \beta_i^4 - T_i \end{cases}$$

712 which can be rewritten as

$$713 \quad D_{t_i} = \begin{cases} \alpha_k^2 - t'_k \leq t_i \leq \beta_k^2 - t'_k & \text{for all } k \neq i \\ T_k - t'_k - \beta_k^4 \leq t_i \leq T_k - t'_k - \alpha_k^4 & \text{for all } k \neq i \\ \alpha_i^2 \leq t_i \leq \beta_i^2 \\ T_i - \beta_i^4 \leq t_i \leq T_i - \alpha_i^4 \end{cases}$$

714 Eliminating t_i , we get :

$$715 \quad D_{\bar{t}_i} = \begin{cases} \alpha_k^2 - t'_k \leq \beta_k^2 - t'_k & \text{for all } k \neq i \\ \alpha_k^2 - t'_k \leq T_k - t'_k - \alpha_k^4 & \text{for all } k \neq i \\ \alpha_k^2 - t'_k \leq \beta_i^2 \\ \alpha_k^2 - t'_k \leq T_i - \alpha_i^4 \\ T_k - t'_k - \beta_k^4 \leq \beta_k^2 - t'_k & \text{for all } k \neq i \\ T_k - t'_k - \beta_k^4 \leq T_k - t'_k - \alpha_k^4 & \text{for all } k \neq i \\ T_k - t'_k - \beta_k^4 \leq \beta_i^2 \\ T_k - t'_k - \beta_k^4 \leq T_i - \alpha_i^4 \\ \alpha_i^2 \leq \beta_k^2 - t'_k & \text{for all } k \neq i \\ \alpha_i^2 \leq T_k - t'_k - \alpha_k^4 & \text{for all } k \neq i \\ \alpha_i^2 \leq \beta_i^2 \\ \alpha_i^2 \leq T_i - \alpha_i^4 \\ T_i - \beta_i^4 \leq \beta_k^2 - t'_k & \text{for all } k \neq i \\ T_i - \beta_i^4 \leq T_k - t'_k - \alpha_k^4 & \text{for all } k \neq i \\ T_i - \beta_i^4 \leq \beta_i^2 \\ T_i - \beta_i^4 \leq T_i - \alpha_i^4 \end{cases}$$

716 One can notice that in $D_{\bar{t}_i}$, all inequalities that do not contain T_i are either tautologies,

720 or inequalities of the form given in Defn. 5

721 Let $D''' = D'' \cup X_{\bar{T}_i} \cup D_{T_i} \cup X_{T_i}$, where $X_{\bar{T}_i}$ is the set of inequalities that do not contain
722 T_i in $D_{\bar{T}_i}$ and X_{T_i} is the set of inequalities that do contain T_i in $D_{\bar{T}_i}$. Then, eliminating T_i
723 from D''' boils down to eliminating T_i from $D_{T_i} \cup X_{T_i}$

$$724 \quad D_{T_i} \cup X_{T_i} = \begin{cases} \alpha_{ij}^5 \leq T_i + t'_j \leq \beta_{ij}^5 & \text{for all } i \neq j \\ -T_i + T_j - t'_j \leq \gamma_{ij}^6 & \text{for all } i \neq j \\ -T_j + t'_j + T_i \leq \gamma_{ji}^6 & \text{for all } i \neq j \\ \alpha_k^2 - t'_k \leq T_i - \alpha_i^4 & \\ 725 \quad T_k - t'_k - \beta_k^4 \leq T_i - \alpha_i^4 & \\ \alpha_i^2 \leq T_i - \alpha_i^4 & \\ T_i - \beta_i^4 \leq \beta_k^2 - t'_k & \text{for all } k \neq i \\ T_i - \beta_i^4 \leq T_k - t'_k - \alpha_k^4 & \text{for all } k \neq i \\ 726 \quad T_i - \beta_i^4 \leq \beta_i^2 & \end{cases}$$

This can be rewritten as :

$$727 \quad D_{T_i} \cup X_{T_i} = \begin{cases} \alpha_{ij}^5 - t'_j \leq T_i \leq \beta_{ij}^5 - t'_j & \text{for all } i \neq j \\ T_j - t'_j - \gamma_{ij}^6 \leq T_i & \text{for all } i \neq j \\ T_i \leq \gamma_{ji}^6 + T_j - t'_j & \text{for all } i \neq j \\ \alpha_k^2 + \alpha_i^4 - t'_k \leq T_i & \\ 728 \quad T_k - t'_k - \beta_k^4 + \alpha_i^4 \leq T_i & \\ \alpha_i^2 + \alpha_i^4 \leq T_i & \\ T_i \leq \beta_k^2 + \beta_i^4 - t'_k & \text{for all } k \neq i \\ T_i \leq T_k - t'_k + \beta_i^4 - \alpha_k^4 & \text{for all } k \neq i \\ 729 \quad T_i \leq \beta_i^2 + \beta_i^4 & \end{cases}$$

729 Again, producing an expression $A_m \leq B_n$ from a pair of inequalities $A_m \leq T_i$ and
730 $T_i \leq B_n$ in $D_{T_i} \cup X_{T_i}$ either produces a tautology, or an inequality of one of the forms
731 in Defn. 5. Hence, domains are closed under computation of a successor for all symbolic
732 moves. ◀

733 D Soundness and completeness of state class graphs abstraction

734 **Proposition 13** Let $\rho^S = (M_0, B_0, D_0) \xrightarrow{e_0} (M_1, B_1, D_1) \xrightarrow{e_1} \dots$ be a symbolic run of a
735 trajectory \mathcal{N} . Then, there exists a run $\rho = (M_0, B_0, \mathcal{T}_0) \xrightarrow{e_0} (M_1, B_1, \mathcal{T}_1) \xrightarrow{e_1} \dots$ of \mathcal{N} such
736 that for every $i \geq 0$, $(M_i, B_i, \mathcal{T}_i)$ matches with (M_i, B_i, D_i) .

737 **Proof.** To show that for any finite symbolic run $(M_0, B_0, D_0) \rightarrow \dots \rightarrow (M_n, B_n, D_n)$ (where
738 the domain of each state class has a feasible solution), there exists a corresponding run of the
739 trajectory net $(M_0, B_0, \mathcal{T}_0) \rightarrow \dots \rightarrow (M_n, B_n, \mathcal{T}_n)$, we will proceed by backward induction
740 on the length of runs. Assume a run ending in state class (M_n, B_n, D_n) .

741 First for the base case, i.e. sequences of length 1 ending in state class (M_n, B_n, D_n) , we
742 can show that there always exists a map \mathcal{T}_n such that $\mathcal{T}_n \in \llbracket D_n \rrbracket$ because D_n is satisfiable.
743 So $(M_n, B_n, \mathcal{T}_n)$ matches (M_n, B_n, D_n) . It then remains to show that, if we can build a
744 matching run up to length j , i.e. find a run $(M_j, B_j, \mathcal{T}_j) \xrightarrow{e_j} \dots \xrightarrow{e_{n-1}} (M_n, B_n, \mathcal{T}_n)$ such that
745 $(M_k, B_k, \mathcal{T}_k)$ matches (M_k, B_k, D_k) for $k \in j \dots n$, then we can find a predecessor configuration
746 for $(M_j, B_j, \mathcal{T}_j)$ and extend this run. We have two types of symbolic transitions:

- 747 1. For a transition $(M_i, B_i, D_i) \xrightarrow{\sigma} (M_j, B_j, D_j)$ of the state class graph, let us show
748 that there exists a corresponding transition in the run of trajectory net $(M_i, B_i, \mathcal{T}_i) \xrightarrow{\sigma}$
749 $(M_j, B_j, \mathcal{T}_j)$ with $\mathcal{T}_i \in \llbracket D_i \rrbracket$. Let $p(\sigma) \bullet \cap P_T$. The map depicting trajectories \mathcal{T}_i can be

750 constructed simply by dropping the variables t_p, T_p of the forward trajectory from \mathcal{T}_j . The
 751 trajectories in other places are unaffected by the firing of σ . By induction hypothesis, we
 752 know that $\mathcal{T}_j \in \llbracket D_j \rrbracket$. Since the inequalities involving variables of unaffected trajectories
 753 are the same in D_i and D_j , we have $\mathcal{T}_i \in \llbracket D_i \rrbracket$.

754 2. For a transition $(M_i, B_i, D_i) \xrightarrow{\text{block } p} (M_j, B_j, D_j)$, let us show that there exists a
 755 corresponding pair of moves in the trajectory net $(M_i, \mathcal{T}_i) \xrightarrow{\delta=t_p} (M_i, \mathcal{T}_i'') \xrightarrow{\text{block } p} (M_j, \mathcal{T}_j)$
 756 with $\mathcal{T}_i \in \llbracket D_i \rrbracket$. The trajectories \mathcal{T}_i can be constructed from \mathcal{T}_j as follows:

- 757 \dashv By the correctness of Fourier-Motzkin Elimination, there exists t_p, T_p such that $\mathcal{T}_j \cup$
 758 $\{t_p, T_p\}$ is a solution to the domain D_i' of the transformed variables. Also, $\mathcal{T}_i'' =$
 759 $\mathcal{T}_j \cup \{t_p'' = 0, T_p'' = T_p\}$ is the required trajectories for the configuration.
- 760 \dashv We can fix consistent values for t_p and T_p , and then, perform the inverse transformation
 761 $T_i = T_i''$ and $t_i = t_i'' + t_p$ to obtain $\mathcal{T}_i \in \llbracket D_i \rrbracket$.

762 Then, $\mathcal{T}_i \in \llbracket D_i \rrbracket$ by construction of \mathcal{T}_i and $(M_i, \mathcal{T}_i) \xrightarrow{\delta=t_f} (M_i, B_i, \mathcal{T}_i'') \xrightarrow{\text{block } p} (M_j, B_j, \mathcal{T}_j)$
 763 is the corresponding transition extending the run of \mathcal{N} .

764 ◀

765 **Proposition 14** Let $\rho = (M_0, B_0, \mathcal{T}_0) \xrightarrow{e_0} (M_1, B_1, \mathcal{T}_1) \xrightarrow{e_1} \dots$ be a run of a trajectory net
 766 \mathcal{N} . Then, there exists a symbolic run $\rho^S = (M_0, B_0, D_0) \xrightarrow{e_0} (M_1, B_1, D_1) \xrightarrow{e_1} \dots$ of \mathcal{N}
 767 such that for every $i \geq 0$, $(M_i, B_i, \mathcal{T}_i)$ matches with (M_i, B_i, D_i) .

768 **Proof.** We proceed by induction on the length of runs to show that symbolic runs have a
 769 counterpart run in the concrete semantics of \mathcal{N} . Let us start with the base case, i.e. a run of
 770 size 1. The initial configuration $(M_0, B_0, \mathcal{T}_0)$ matches the state class (M_0, B_0, D_0) because
 771 values $T_p^0 = t_p^0$ for progressing trajectories are sampled from the respective time intervals
 772 $[\alpha_p^S, \beta_p^S]$ associated to trajectory places containing a progressing trajectory.

773 Let us consider that the property holds up to index n , that is, for the run $(M_0, B_0, \mathcal{T}_0) \xrightarrow{\delta_0} (M_0, B_0, \mathcal{T}_0 + \delta_0) \xrightarrow{e_0} (M_1, B_1, \mathcal{T}_1) \dots (M_n, B_n, \mathcal{T}_n)$, there exists a sequence $(M_0, B_0, D_0) \xrightarrow{e_0} (M_1, B_1, D_1) \dots (M_n, B_n, D_n)$ such that $(M_i, B_i, \mathcal{T}_i)$ matches (M_i, B_i, D_i) for every $i \leq n$.

774 We know from the induction hypothesis that $\mathcal{T}_n \in \llbracket D_n \rrbracket$. Let us now consider possible
 775 successors of $(M_n, B_n, \mathcal{T}_n)$. We have two possible types of moves:

- 776 1. A move of the form $(M_n, B_n, \mathcal{T}_n) \xrightarrow{\sigma} (M_j, B_j, \mathcal{T}_j)$. By the semantics of model, the effect
 777 of firing of a transition σ on trajectories is the removal of a blocked trajectory and of tokens
 778 in the preset of σ and the creation of tokens in the postset of σ and of a new trajectory in
 779 $(\sigma) \bullet \cap P_T$. The rest of trajectories are unaffected by the firing. By construction of successor
 780 of state classes, we know that markings and blocked transitions follow the same rules in the
 781 semantics and in the symbolic moves. Hence, if $(M_n, B_n, D_n) \xrightarrow{\sigma} (M_{n+1}, B_{n+1}, D_{n+1})$,
 782 we necessarily have $M_{n+1} = M_j, B_{n+1} = B_j$. We also know that D_{n+1} is uniquely defined
 783 from D_n and σ (it is the canonical form of $\text{SuccF}(D_n, \sigma)$). It hence remains to show
 784 that $\mathcal{T}_j \in \llbracket D_{n+1} \rrbracket$. We know that D_n is satisfiable, because $\mathcal{T}_n \in \llbracket D_n \rrbracket$. Let p be the
 785 place receiving the new trajectory after firing of σ . We have $\mathcal{T}_j(p_k) = \mathcal{T}_n(p_k)$ for every
 786 place $p_k \neq p$, and $\mathcal{T}_j(p) = (T_p, T_p)$ with $T_p \in [\alpha_p^s, \beta_p^s]$. To compute D_{n+1} , we add the
 787 inequalities defined in $\text{SuccF}(D_n, \sigma)$ (see section 4.1). The fresh variables satisfy the
 788 inequalities $\alpha_p^s \leq T_p \leq \beta_p^s, \alpha_p^s \leq t_p \leq \beta_p^s$ and $0 \leq T_p - t_p \leq 0$. The other inequalities in
 789 D_{n+1} involve only variables of unaffected trajectories and are unchanged from D_n . As
 790 $\mathcal{T}_n \in \llbracket D_n \rrbracket$, we have that \mathcal{T}_j satisfies all inequalities in D_{n+1} too. So the induction step
 791 preserves the property for transitions frings.

794 2. A pair of moves $(M_n, B_n, \mathcal{T}_n) \xrightarrow{\delta=t_p} (M_n, B_n, \mathcal{T}_n'') \xrightarrow{\text{block } p} (M_j, B_j, \mathcal{T}_j)$. According to
 795 the semantics, we have $\mathcal{T}(p) = (T_p, t_p)$ and $t_p = \min(\{t_i\})$. Let $(M_n, B_n, D_n) \xrightarrow{\text{block } p}_S$
 796 $(M_{n+1}, B_{n+1}, D_{n+1})$, where D_{n+1} is the canonical form of $\text{Succ}B(D, p)$. The configuration
 797 $(M_j, B_j, \mathcal{T}_j)$ must match the state class $(M_{n+1}, B_{n+1}, D_{n+1})$. We have $M_{n+1} = M_n = M_j$,
 798 and $B_{n+1} = B_j = B_n \cup \{p\}$. It hence remains to show that $\mathcal{T}_j \in \llbracket D_{n+1} \rrbracket$. Performing the
 799 timed move $\delta = t_p$ from $(M_n, B_n, \mathcal{T}_n)$ consists in replacing trajectories of the form (T_i, t_i)
 800 by trajectories (T_i'', t_i'') , to obtain configuration $(M_n, B_n, \mathcal{T}_n'')$. Let us now denote by D_n''
 801 the domain obtained after replacing variables t_i by $t_i'' + t_p$. We have $\mathcal{T}_n'' \in \llbracket D_n'' \rrbracket$. Note
 802 that this variable change is exactly the first step performed when computing $\text{Succ}B(D, p)$.
 803 The rest of the calculus is the elimination of variables T_p and t_p using the Fourier-
 804 Motzkin projection. By correctness of Fourier-Motzkin elimination, we know that for
 805 any solution μ'' of D_n'' , the projection of μ'' on remaining variables is a solution of D_{n+1} .
 806 According to the semantics of trajectory nets, \mathcal{T}_j is the projection of \mathcal{T}_n'' on variables
 807 $\{T_k, t_k \mid k \neq p \wedge \mathcal{T}_n''(k) \text{ is defined}\}$. Hence, $\mathcal{T}_j \in \llbracket D_{n+1} \rrbracket$, and this induction step preserves
 808 matching.

809

810 **Theorem 19** Given a state class (M_n, B_n, D_n) reachable from initial state class (M_0, B_0, D_0) ,
 811 and a solution $\mathcal{T}_n \in \llbracket D_n \rrbracket$, there exists a run in the original trajectory net that ends in
 812 configuration $(M_n, B_n, \mathcal{T}_n)$.

813 **Proof.** The proof is similar to the proof for soundness (13). Let us assume that a run in
 814 the state class graph is $\rho^S = (M_0, B_0, D_0) \rightarrow \dots \rightarrow (M_i, B_i, D_i) \rightarrow (M_j, B_j, D_j) \rightarrow \dots \rightarrow$
 815 (M_n, B_n, D_n) . We can inductively construct a run ρ of the trajectory net that matches
 816 ρ^S . Let us assume that we have a partial run $(M_j, B_j, \mathcal{T}_j) \rightarrow \dots \rightarrow (M_n, B_n, \mathcal{T}_n)$ with the
 817 induction hypothesis that $\mathcal{T}_j \in \llbracket D_j \rrbracket$. We can construct $(M_i, B_i, \mathcal{T}_i)$ such that $\mathcal{T}_i \in \llbracket D_i \rrbracket$. We
 818 can have two cases:

- 819 ■ $(M_i, B_i, D_i) \xrightarrow{\sigma}_S (M_j, B_j, D_j)$. The effect of firing σ on markings and blocked transitions
 820 is deterministic, and is the same in the symbolic moves and in the semantics. Hence,
 821 we have $M_i = M_j + \bullet(\sigma) - (\text{transition})^\bullet$, $B_i = B_j$ and it remains to find a map \mathcal{T}_i such
 822 that $\mathcal{T}_i \in \llbracket D_i \rrbracket$. Let t_p, T_p represent the variables associated with the trajectory created
 823 in place $p = (\sigma)^\bullet \cap P_T$ when firing σ . In this case, \mathcal{T}_i is the projection of \mathcal{T}_j on all its
 824 variables except $\{t_p, T_p\}$. Clearly, \mathcal{T}_i satisfies the induction hypothesis $\mathcal{T}_i \in \llbracket D_i \rrbracket$. This is
 825 because, $D_j = \text{Succ}F(D_i, \sigma)$. Hence, when building D_j from D_i , we just add inequalities
 826 involving variables t_p, T_p and the remaining inequalities of D_j are unaffected. Hence the
 827 projection of \mathcal{T}_j gives a map $\mathcal{T}_i \in \llbracket D_i \rrbracket$, and the discrete move $(M_i, B_i, \mathcal{T}_i) \xrightarrow{\sigma} (M_j, B_j, \mathcal{T}_j)$
 828 is a valid move in the semantics of trajectory net.
- 829 ■ $(M_i, B_i, D_i) \xrightarrow{\text{block } p}_S (M_j, B_j, D_j)$. By correctness of Fourier-Motzkin elimination, we
 830 know there exists t_p, T_p such that $\mathcal{T}_i'' = \mathcal{T}_j \cup \{p \rightarrow (T_p, t_p)\}$ is a solution of domain D_i''
 831 obtained after transformation of variables in D_i . Then, for every fixed pair of values
 832 t_p, T_p , $\mathcal{T}_i = \{t_i, T_i \mid t_i = t_j - t_p, T_i = T_j\} \cup \{t_p, T_p\}$ is the set of trajectories satisfying the
 833 induction hypothesis $\mathcal{T}_i \in \llbracket D_i \rrbracket$. This is because:
 - 834 - By construction of successor of domains, we know after performing the inverse
 835 transformation $t_i = t_i'' - t_p$ and $T_i = T_i''$ on \mathcal{T}_i'' gives a solution a solution \mathcal{T}_i which
 836 satisfies the inequalities $t_p \leq t_i, \forall i \neq p$, in addition to all the inequalities of D_i . Hence
 837 $\mathcal{T}_i \in \llbracket D_i \rrbracket$.

XX:24 Symbolic domains and reachability for nets with trajectories.

838 – Also $(M_i, B_i, \mathcal{T}_i) \xrightarrow{\delta=t_p} (M_i, B_i, \mathcal{T}_i'')$ is a valid time move since $t_p = \min\{t_i\}$.

839 Hence, $(M_i, B_i, \mathcal{T}_i) \xrightarrow{\delta=t_p} (M_i, B_i, \mathcal{T}_i'')$ $\xrightarrow{\text{block } p} (M_j, B_j, \mathcal{T}_j)$ are valid moves according to the
840 semantics of trajectory net and the induction hypothesis is satisfied.

841 By induction, we can hence construct a valid run of \mathcal{N} which ends in a configuration
842 $(M_n, B_n, \mathcal{T}_n)$ matching state class (M_n, B_n, D_n) . When considering the last step $(M_0, B_0, D_0) \xrightarrow{e_0}$
843 (M_1, B_1, D_1) , the induction step is still valid, but the matching run must start from
844 $C_0 = (M_0, B_0, \mathcal{T}_0)$. So, assuming that $\mathcal{T}_0(p_i) = (T_i, T_i)$ with $T_i > 0$ for every non-empty
845 place in P_T , e_0 is the blocking of a trajectory in some place p , and the choice of t_p, T_p is
846 restricted to $T_p = t_p = \mathcal{T}_0(p)$. ◀

847 **E Finiteness of Domains**

848 ▶ **Lemma 28.** *Given an inequality $\alpha \leq \text{expr} \leq \beta$, with $\text{expr} = \sum_{i \in S_1} A_i x_i + \sum_{i \in S_2} B_i x_i$
849 where x_i are variables with bounds $\alpha_i \leq x_i \leq \beta_i$ and $A_i > 0$ and $B_i < 0$, an equivalent
850 inequality is:*

$$851 \quad \max(\alpha, \sum_{i \in S_1} A_i \alpha_i + \sum_{i \in S_2} B_i \beta_i) \leq \text{expr} \leq \min(\beta, \sum_{i \in S_1} A_i \beta_i + \sum_{i \in S_2} B_i \alpha_i)$$

852 **Proof.** Note that the minimum and maximum possible values that expr can take, based on
853 the bounds on x_i are $\sum_{i \in S_1} A_i \alpha_i + \sum_{i \in S_2} B_i \beta_i$ and $\sum_{i \in S_1} A_i \beta_i + \sum_{i \in S_2} B_i \alpha_i$ respectively.
854 The Lemma 28 follows directly from this. ◀

855 ▶ **Lemma 29.** *For the same settings as Lemma 28 if $\alpha > \sum_{i \in S_1} A_i \beta_i + \sum_{i \in S_2} B_i \alpha_i$ or
856 $\beta < \sum_{i \in S_1} A_i \alpha_i + \sum_{i \in S_2} B_i \beta_i$, the inequality has no solution*

857 **Proof.** The proof again follows from the bound on expr based on bounds on x_i ◀

858 ▶ **Lemma 30.** *For the inequality in Lemma 28, if it has a solution, then there exists an
859 equivalent inequality $\alpha' \leq \text{expr} \leq \beta'$ with:*

$$860 \quad \sum_{i \in S_1} A_i \alpha_i + \sum_{i \in S_2} B_i \beta_i \leq \alpha' \leq \sum_{i \in S_1} A_i \beta_i + \sum_{i \in S_2} B_i \alpha_i$$

$$861 \quad \sum_{i \in S_1} A_i \alpha_i + \sum_{i \in S_2} B_i \beta_i \leq \beta' \leq \sum_{i \in S_1} A_i \beta_i + \sum_{i \in S_2} B_i \alpha_i$$

$$862$$

863 **Proof.** – The lower bound on α' comes from Lemma 28 with $\alpha' = \max(\alpha, \sum_{i \in S_1} A_i \alpha_i +$
864 $\sum_{i \in S_2} B_i \beta_i)$

865 – The upper bound on α' comes from Lemma 29 since the inequality has a solution

866 – The lower bound on β' comes from Lemma 29 since the inequality has a solution

867 – The upper bound on β' comes from Lemma 28 with $\beta' = \min(\beta, \sum_{i \in S_1} A_i \beta_i + \sum_{i \in S_2} B_i \alpha_i)$
868 ◀

869 In the setting of trajectory nets, we immediately have bounds for variables t_i and T_i : T_i
870 is sampled from interval $[\alpha_i^s, \beta_i^s]$ and since time remaining time t_i for a trajectory has an
871 original value T_i and then decreases we have $t_i \in [0, \beta_i^s]$. Now using Lemma 30, we have the
872 following bounds on the constants for an equivalent system of inequalities to have a solution:

$$873 \quad \alpha_i^s \leq \alpha_i^1 \leq \beta_i^s \quad (7)$$

$$874 \quad \alpha_i^s \leq \beta_i^1 \leq \beta_i^s \quad (8)$$

$$875 \quad 0 \leq \alpha_i^2 \leq \beta_i^s \quad (9)$$

$$876 \quad 0 \leq \beta_i^2 \leq \beta_i^s \quad (10)$$

$$877 \quad -\beta_j^s \leq \gamma_{ij}^3 \leq \beta_i^s \quad (11)$$

$$878 \quad \alpha_i^s - \beta_i^s \leq \alpha_i^4 \leq \beta_i^s \quad (12)$$

$$879 \quad \alpha_i^s - \beta_i^s \leq \beta_i^4 \leq \beta_i^s \quad (13)$$

$$880 \quad \alpha_i^s - \beta_i^s \leq \alpha_{ij}^5 \leq \beta_i^s + \beta_j^s \quad (14)$$

$$881 \quad \alpha_i^s - \beta_i^s \leq \beta_{ij}^5 \leq \beta_i^s + \beta_j^s \quad (15)$$

$$882 \quad -\beta_i^s + \alpha_j^s - \beta_j^s \leq \gamma_{ij}^6 \leq -\alpha_i^s + \beta_i^s + \beta_j^s \quad (16)$$

883