Decidability of LTL for Vector Addition Systems with one zero-test

Rémi Bonnet

LSV, CNRS, ENS Cachan

Abstract. We consider the class of Vector Addition Systems with one zero-test and we show that the model-checking problem for LTL is decidable thanks to a reduction to the computability of the cover and the decidability of reachability. Our proof uses the notion of increasing loop, that we refine to fit the non-standard monotony of our system.

1 Introduction

Petri Nets Vector Addition Systems (VAS) are a well-known classes of counter systems, equivalent to Petri Nets. The reachability problem is known to be decidable [14,15,16,17] even if its complexity is still an open problem. As the equality of the reachability sets (the set of states that are reachable from an initial state) of two such systems is undecidable [13], one cannot compute a canonical finite representation of the reachability set. However, there is such an effective finite representation for the *cover*, the downward closure of the reachability set, which is connected to various verification problems, like the control state reachability problem. If we add to VASS the ability to test at least two counters to zero, one obtains a model equivalent to Minsky machines, for which all nontrivial properties are undecidable. The study of VASS with *a single* zero-test transition began recently, and a reasonable number of results are now known. Reinhardt [18] has shown that the reachability problem is decidable. Abdulla and Mayr [2] have provided an algorithm based on the backward procedure of Well Structured Transition Systems [1,9] to decide coverability of a state. Termination and Boundedness were shown by Finkel and Sangnier [8], while an algorithm to compute the maximal elements of the cover has been found by Bonnet, Finkel, Leroux and Zeitoun [3].

LTL Linear-time logic is a widely used logic in order to express safety and liveness properties of a system. Emerson [4] provided an algorithm based on a covering graph that worked on well structured transition systems, but that was not guaranteed to terminate. Esparza [5,6] showed that LTL on the actions of a VASS was decidable, but that CTL was not, and that LTL became undecidable when predicates regarding the states were added. Habermehl [12] completed this proof by showing EXPSPACE-completeness of LTL satisfiability.

Our contribution We complete the works of [3] by showing decidability of LTL model checking. We start by the usual reduction of LTL model-checking to repeated control state reachability by defining the synchronized product of a $VASS_0$ and a Buchi automaton. Then, we show that repeated control state reachability can be decided by looking at the existence of a special kind of increasing loop. We first provide a reduction of this problem of existence of a loop to the reachability problem for $VASS_0$ when the starting point of there is a finite number of such subsets, and hence that if one is able to compute a finite representation of the cover, existence of an increasing loop can be decided by looking at all the subsets.

2 Preliminaries

2.1 Generalities

Sets and Vectors. The cartesian product of two sets X and Y is noted $X \times Y$ and the disjoint union $X \uplus Y$. For $d \geq 1$, we write any $x \in X^d$ as $x = (x[0], \ldots, x[d-1])$, with $x[i] \in X$. For $x_1 \in X^{d_1}$ and $x_2 \in X^{d_2}$, we let (x_1, x_2) be the vector of $X^{d_1+d_2}$ obtained by gluing x_1 and x_2. Addition of vectors is defined by $(x+y)[i] = x[i] + y[i]$ and substraction similarly.
We denote by $\mathbb{N}_\omega$ the set $\mathbb{N} \cup \{\omega\}$ where ω is an element strictly greater than all integers. We will use the notations 0^d to denote the vector composed of d 0's, ω^d for the vector composed of d ω's, and e_i^d be the vector of $\mathbb{N}^d$ such that $e_i^d[i] = 1$ and $e_i^d[j] = 0$ if $i \neq j$.

Orderings. An *ordering* $\preceq$ on a set X is a reflexive, transitive and antisymmetric binary relation on X. Given $x, y \in X$, we write $x \prec y$ for $x \preceq y$ and $x \neq y$. The *pointwise ordering* on X^d, still denoted $\preceq$, is defined by $x \preceq y$ if $x[i] \preceq y[i]$ for all i. Given $Y \subseteq X$, $\downarrow_{\preceq} Y = \{x \in X \mid \exists y \in Y,\ x \preceq y\}$ denotes the downward closure of Y with respect to $\preceq$. The set Y is said *downward closed* if $Y = \downarrow_{\preceq} Y$. In $\mathbb{N}^d$, we shorten $\downarrow_{\leq}$ as $\downarrow$.
An ordering $\preceq$ on X is *well* if, given any sequence $(x_i)_{i \in \mathbb{N}}$ of elements of X, one can find $i < j$ such that $x_i \leq x_j$. The usual ordering on $\mathbb{N}^d$ is well.

Basis in $\mathbb{N}_\omega^d$. Given a downward-closed set $X \subseteq \mathbb{N}^d$, a *basis* of X is a finite subset B of $\mathbb{N}_\omega^d$ such that $\downarrow B \cap \mathbb{N}^d = X$. Any downward-closed set of $\mathbb{N}^d$ admits a basis [7] and one can show that the maximal elements of any basis B of X still form a basis which does not depend of B. It is minimal for inclusion among all basis, and is called the *minimal basis*.

Words. The set of finite words (shortly words) on A is denoted A^*. A word $u \in A^*$ is written $a_1 a_2 \ldots a_n$, $a_i \in A$, and we will also use the notation $u[i]$ to refer to the i-th letter of u. The concatenation of two words u and v is simply written uv and the empty word ε, with $\varepsilon a = a \varepsilon = a$. A^+ denotes the set of non-empty words. An infinite word on A is a sequence $(a_i)_{i \in \mathbb{N}}$. Given an infinite word u, we use the notation $u[k \ldots]$ to refer to the subsequence $(u[k+i])_{i \in \mathbb{N}}$. The set of infinite words on A is written A^ω and the union of finite and infinite words is written $A^{\leq \omega}$

2.2 Transition Systems

Definition 1. *A* Labelled Transition System *(LTS) $\mathcal{S}$ is a tuple $\langle X, A, \rightarrow, s_{in}\rangle$ where X is the set of states, A is the set of transition labels, $\rightarrow \subseteq X \times (A \cup \{\varepsilon\}) \times X$ is the transition relation and s_{in} is the initial state.*

We will use the notations $\mathit{States}(\mathcal{S})$, $\mathit{Actions}(\mathcal{S})$ and $\mathit{Init}(\mathcal{S})$ to refer respectively to X, A, s_{in}. Moreover, we write $s \xrightarrow{a} s'$ if $(s, a, s') \in \rightarrow$ and we extend this notation to words by $s \xrightarrow{\varepsilon} s$ and $s \xrightarrow{uv} s'$ iff $\exists s''$, $s \xrightarrow{u} s'' \xrightarrow{v} s'$. Note that transitions may be labelled by ε and hence that $s \xrightarrow{a} s'$ where $a \in A$ doesn't mean that s' is reached from s by one transition, but by one transition labelled by a and any number of ε-transitions.
A run w of $\mathcal{S}$ is a sequence $(s_i, t_i) \in (\mathit{States}(\mathcal{S}) \times \mathit{Actions}(\mathcal{S}))^{\leq\omega}$ such that $s_0 = \mathit{Init}(\mathcal{S})$ and $\forall i, s_i \xrightarrow{t_i} s_{i+1}$. Given a run $(s_i, t_i)_i$, we define $\mathit{actions}(w)$ as $(t_i)_i$. The *reachability set* is defined as $\mathit{Reach}(\mathcal{S}) = \{y \in \mathit{States}(\mathcal{S}) \mid \exists u \in \mathit{Actions}(\mathcal{S})^* \mid \mathit{Init}(\mathcal{S}) \xrightarrow{u} y\}$. If $\mathit{States}(\mathcal{S})$ is ordered by $\leq$, the *cover* is $\mathit{Cover}_{\leq}(\mathcal{S}) = \downarrow_{\leq}\mathit{Reach}(\mathcal{S})$. The subscript $\leq$ will be omitted when it is clear from the context.

2.3 Vector Addition Systems

Definition 2. *A* Vector Addition System with States and one zero-test *(shortly VASS_0) of dimension d is a tuple $\mathcal{V} = \langle Q, A, a_Z, T, s_{in}\rangle$ where Q is a finite set of control locations, A is a finite alphabet of* actions*, $a_Z \in A$ is called the* zero-test*, $T \subseteq Q \times \mathbb{Z}^d \times A \times Q$ is the finite set of transitions, and $s_{in} = (q_{in}, x_{in}) \in Q \times \mathbb{N}^d$ is the* initial state*.*

Intuitively, a VASS_0 works on d counters, one for each component, whose initial values are given by x_{in}. If $(q, v, a, q') \in T$, $a \in A$ when the VASS_0 is in control location q adds the vector v to the counters and moves the system in the control location q'. This action can be executed only if the resulting counters values are non-negative. Moreover, we have the restriction that a_Z can be fired only if the first counter is zero.
More formally, a VASS_0 $\langle Q, A, a_Z, T, s_{in}\rangle$ induces a transition system $\mathcal{S}$ by:

$$\begin{array}{rcll}
\mathit{States}(\mathcal{S}) & = & Q \times \mathbb{N}^d & \\
\mathit{Actions}(\mathcal{S}) & = & A & \\
\mathit{Init}(\mathcal{S}) & = & (q_{in}, x_{in}) & \\
(q, x) \xrightarrow{a}_{\mathcal{S}} (q', x') & \iff & (q, x' - x, a, q') \in T & \text{for } a \neq a_Z \\
(q, x) \xrightarrow{a_Z}_{\mathcal{S}} (q', x') & \iff & \begin{cases} (q, x' - x, a_Z, q') \in T \\ x[0] = 0 \end{cases} &
\end{array}$$

A finite automaton (FA) is a VASS_0 of dimension 0. We get back the usual definition of VASS (without zero-test) as $\langle Q, A, T, s_{in}\rangle$ whose semantics are the same as the VASS_0 $\langle Q, A \uplus \{a_Z\}, a_Z, T, s_{in}\rangle$ where a_Z doesn't appear in T.

We recall from previous works the following properties of VASS_0 that we will use in the sequel :

Theorem 1. *(Reachability [18], Coverability [2])*
Let $\mathcal{S}$ be the transition system associated to a $VASS_0$. Membership in $Reach(\mathcal{S})$ and $Cover(\mathcal{S})$ is decidable.

Regarding coverability, we can be even more precise. Actually, $Cover(\mathcal{S})$ is not only recursive, but also has a finite representation.

Theorem 2. *(Cover [3])*
Let $\mathcal{S}$ be the transition system associated to a $VASS_0$. One can compute the minimal basis of $Cover(\mathcal{S})$.

To simplify some proofs, we will only consider *normed* VASS_0, i.e. VASS_0 such that there exists a unique (q, q', δ) for which $(q, \delta, a_z, q') \in T$. We show in the appendix (proposition 3) that any VASS_0 can be rewritten in a normed VASS_0 satisfying the same LTL formulas.

3 The LTL Logic

3.1 Buchi Automata and LTL

Definition 3. *A* Buchi automaton *is a pair $\langle \mathcal{A}, F\rangle$ where $\mathcal{A}$ is a finite automaton and $F \subseteq States(\mathcal{S})$.*

An infinite run $((q_i, x_i), t_i)_{i\in\mathbb{N}}$ of a Buchi Automata is accepted iff $\{i \in \mathbb{N} \mid q_i \in F\}$ is infinite.

Definition 4. *Given a set A, the set of LTL formulae is given by the following grammar, where a ranges over A :*

$$\varphi ::= true \mid a \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \mathcal{X}\varphi \mid \varphi_1 \mathcal{U} \varphi_2$$

Formulas are interpreted on infinite words over the alphabet A. We denote that w satisfies a formula φ by $w \models \varphi$. This relation is defined inductively on the structure of φ by:

$$\begin{array}{lll}
w \models true & & \\
w \models a & \iff & w[0] = a \\
w \models \neg\varphi & \iff & w \not\models \varphi \\
w \models \varphi_1 \wedge \varphi_2 & \iff & w \models \varphi_1 \text{ and } w \models \varphi_2 \\
w \models \mathcal{X}\varphi & \iff & w[1\ldots] \models \varphi \\
w \models \varphi_1 \mathcal{U} \varphi_2 & \iff & \exists i,\ \forall 0 \leq j < i,\ w[j\ldots] \models \varphi_1 \wedge w[i\ldots \models \varphi_2
\end{array}$$

Given a LTL formula φ, one can build a Buchi automaton $\mathcal{B}_\varphi$ such that the set of infinite words satisfying φ is exactly the infinite words accepted by $\mathcal{B}_\varphi$. We refer to the abundant literature on this subject for the construction (Proposition 4.1 of [5], but also [11] and [10]).

3.2 Model Checking

We consider two problems on $VASS_0$. *LTL Model Checking* consists in, given a $VASS_0$ $\mathcal{V}$ inducing a transition system $\mathcal{S}$ and a LTL formula φ on $Actions(\mathcal{S})$, determining whether there exists an infinite run w of $\mathcal{S}$ such that $actions(w) \models \varphi$. *Repeated Control State Reachability* consists in, given a $VASS_0$ $\mathcal{S} = \langle Q, A, a_Z, T, s_{in} \rangle$ and a control location $q_f \in Q$, determining whether there exists an infinite run $w = (s_1, t_1) \dots (s_k, t_k) \dots$ of $\mathcal{S}$ such that $\{j \in \mathbb{N} \mid \exists x_j,\ s_j = (q_f, x_j)\}$ is infinite.
We have the following usual reduction :

Proposition 1. *LTL Model Checking on $VASS_0$ reduces to Repeated Control State Reachability on $VASS_0$.*

Proof. Let $\mathcal{V} = \langle Q, A, a_Z, T, (q_{in}, x_{in}) \rangle$ be a $VASS_0$ and φ a LTL formula on A. Let $\mathcal{B} = \langle Q_\mathcal{B}, A, T_\mathcal{B}, q_{in\mathcal{B}}, F \rangle$ be a Buchi automaton representing φ. The synchronized product of $\mathcal{V}$ and $\mathcal{B}$ is defined as the $VASS_0$ $\mathcal{V}' = \langle Q \times Q_\mathcal{B}, A, a_Z, T', ((q_{in}, q_{in\mathcal{B}}), x_{in}) \rangle$ with :

$$T' = \{((q_1, q_2), \delta, a, (q_1', q_2')) \mid (q_1, \delta, a, q_1') \in T \wedge (q_2, a, q_2') \in T_\mathcal{B}\}$$

This $VASS_0$ induces a transition system $\mathcal{S}'$, and it is easy to check that a sequence $((q_i^1, q_i^2, x_i), a_i)_i$ is a run of $\mathcal{S}'$ if and only if $((q_i^1, x_i), a_i)_i$ is a run of $\mathcal{S}$ and $(q_i^2, a_i)_i$ is a run of $\mathcal{B}$. Hence, there exists a run of $\mathcal{S}'$ that visits infinitely often $Q \times F$, if and only if there exists runs w of $\mathcal{S}$ and w' of $\mathcal{B}$ such that $actions(w) = actions(w')$ and w' visits infinitely often F, which means that $actions(w) \models \varphi$.

4 Decidability of Repeated Control State Reachability

Let us introduce the order $\leq_0$ as $x \leq_0 y \iff x \leq y \wedge x[0] = y[0]$. We have the following monotony property for $VASS_0$:

Proposition 2. *Let $q \in Q$ and $x, y \in \mathbb{N}^d$ with $x \leq_0 y$. If a sequence of transitions can be fired from (q, x), it can be fired from (q, y).*

Our idea is to make an equivalence between repeated control location reachability and the existence of an increasing loop going through this state.

Definition 5. *Let $\mathcal{V}$ be a $VASS_0$ and $\mathcal{S}$ its associated transition system. Given ℓ in $\mathbb{N} \times \mathbb{N}_\omega^{d-1}$, we say that $(x, u, y) \in \mathbb{N}^d \times A^+ \times \mathbb{N}^d$ is a ℓ-increasing loop on q in $\mathcal{V}$ if we have $(q, x) \xrightarrow{u}_\mathcal{S} (q, y)$, $x \leq_0 y$ and $x \leq_0 \ell$.*

Our proof is in three steps : First we show that if we have the restriction that $\ell[0] = 0$, we can decide the existence of an ℓ-increasing loop. Then we show that, assuming the run we are looking for goes infinitely through the zero-test, the existence of a run visiting infinitely often a control location reduces to the existence of a ℓ-increasing loop with $\ell[0] = 0$. We conclude by taking also care of runs visiting the zero-test only a finite number of times.

We will fix a normed VASS_0 $\mathcal{V} = \langle Q, A, a_Z, T, s_{in} \rangle$ of dimension d and $\mathcal{S}$ its associated transition system. Unless otherwise specified, all lemmas refer to this VASS_0.

Lemma 1. *Let $q_f \in Q$ and $\ell \in \{0\} \times \mathbb{N}_\omega^{d-1}$. The existence of an ℓ-increasing loop on q_f is decidable.*

Proof. Let us take $\ell \in \{0\} \times \mathbb{N}_\omega^{d-1}$. Without loss of generality (by reordering counters), we have $\ell = (0, \omega^m, b)$, $b \in \mathbb{N}^n$, with $d = 1 + m + n$. We will build a VASS_0 $\mathcal{V}'$ inducing a transition system $\mathcal{S}'$ that will mimic $\mathcal{S}$ in the following sense (x represents a state $\mathcal{S}$, and x' the associated state in $\mathcal{S}'$):

- The counter that can be tested for zero is preserved.
- A counter $x[i]$ for $1 \leq i \leq m$ (the ones for which $\ell[i] = \omega$) is replaced by two counters $x'[i]$ and $x'[i+m]$, such that $x'[i+m] - x'[i] \leq x[i]$. This simulates a counter that can go arbitrarily below its initial value, and that can leak non-deterministically.
- A counter $x[i]$ for $m+1 \leq i \leq m+n$ (the ones for which $\ell[i] \neq \omega$) is replaced by one counter $x'[i+m]$ such that $x'[i+m] \leq x[i]$. This simulates a counter that can leak non-deterministically.

Note that states of $\mathcal{S}$ will be represented as (x, v, z) with $x \in \mathbb{N}$, $v \in \mathbb{N}^m$ and $z \in \mathbb{N}^n$ while states of $\mathcal{S}'$ will be represented as (x, v, w, z) with $x \in \mathbb{N}$, $(v, w) \in \mathbb{N}^m \times \mathbb{N}^m$ and $z \in \mathbb{N}^n$.

Formally, we define $\mathcal{V}' = \langle Q, A, a_Z, T', s'_{in} \rangle$ of dimension $d' = 1 + 2m + n$ by:

$$s'_{in} = (0, 0, 0, b)$$

$$T' = \begin{array}{ll} \{(q, a, (x, 0^m, v, w), q') \mid (q, a, (x, v, w), q') \in T\} \cup & \text{(T1)} \\ \{(q, \varepsilon, (0, 0^m, 0^m, -e_i^n), q) \mid q \in Q \wedge 1 \leq i \leq n\} \cup & \text{(T2)} \\ \{(q, \varepsilon, (0, e_i^m, e_i^m, 0^n), q) \mid q \in Q \wedge 1 \leq i \leq m\} \cup & \text{(T3)} \\ \{(q, \varepsilon, (0, -e_i^m, -e_i^m, 0^n), q) \mid q \in Q \wedge 1 \leq i \leq m\} \cup & \text{(T4)} \\ \{(q, \varepsilon, (0, 0^m, -e_i^m, 0^n), q) \mid q \in Q \wedge 1 \leq i \leq m\} & \text{(T5)} \end{array}$$

(T1) is the traduction of the transition of $\mathcal{S}$. (T2) makes the counters of index from $1 + 2 * m + 1$ to $1 + 2 * m + n$ (we recall these counters represent the counters of index $1 + m + 1$ to $1 + m + n$ in $\mathcal{S}$) lossy. (T3) + (T4) imply that only the relative value of the counters i and $i + m$ matters (for $1 \leq i \leq m$). This simulates a counter living in $\mathbb{Z}$. Finally, (T5) makes the previous counter lossy.

We will show that the existence of $x, y \in \{0\} \times \mathbb{N}^{d-1}$ and $u \in A^+$ such that $x \leq \ell$, $(q_f, x) \xrightarrow{u}_{\mathcal{S}} (q_f, y)$ and $x \leq y$ is equivalent to the reachability in $\mathcal{S}'$ of $(0, 0^m, 0^m, b)$ from itself using at least one non-epsilon transition. Note that reachability by using at least one non-epsilon transition is reducible to reachability by adding a lossy counter, starting at zero, that is increased when a non-epsilon transition is fired.

⇒ Let us assume the existence of $x, y \in \{0\} \times \mathbb{N}^{d-1}$ and $u \in A^+$ such that $x \leq \ell$, $x \xrightarrow{u} y$ and $x \leq y$.
Let $x = (0, \alpha_1, \beta_1)$, $y = (0, \alpha_2, \beta_2)$ and $\ell = (0, \omega^m, b)$ with $\alpha_1 \leq \alpha_2$, $\beta_1 \leq \beta_2$ and $\beta_1 \leq b$.
Because $(q_f, 0, \alpha_1, \beta_1) \xrightarrow{u}_{\mathcal{S}} (q_f, 0, \alpha_2, \beta_2)$, we have $(q_f, 0, \alpha_1, \alpha_1, \beta_1) \xrightarrow{u}_{\mathcal{S}'} (q_f, 0, \alpha_1, \alpha_2, \beta_2)$. Because $\beta_1 \leq b$, we also have that $(q_f, 0, \alpha_1, \alpha_1, b) \xrightarrow{u}_{\mathcal{S}'} (q_f, 0, \alpha_1, \alpha_2, \beta_2 + b - \beta_1)$.
Then, we have :

$$\begin{aligned}(q_f, 0, 0^m, 0^m, b) &\xrightarrow{\varepsilon}_{\mathcal{S}'} (q_f, 0, \alpha_1, \alpha_1, b)\\ &\xrightarrow{u}_{\mathcal{S}'} (q_f, 0, \alpha_1, \alpha_2, b + \beta_2 - \beta_1)\\ &\xrightarrow{\varepsilon}_{\mathcal{S}'} (q_f, 0, \alpha_1, \alpha_1, b)\\ &\xrightarrow{\varepsilon}_{\mathcal{S}'} (q_f, 0, 0^m, 0^m, b)\end{aligned}$$

⇐ Assume that we have $(q_f, 0, 0^m, 0^m, b) \xrightarrow{u} (q_f, 0, 0^m, 0^m, b)$. We will show there exist $x, y \in \{0\} \times \mathbb{N}^{d-1}$ with $x \leq \ell$ such that $(q_f, x) \xrightarrow{u}_{\mathcal{S}} (q_f, y)$.
Let $(t_i, v_i, w_i, z_i)_{0 \leq i \leq k}$ such that $(t_i, v_i, w_i, z_i) \rightarrow_{\mathcal{S}'} (t_{i+1}, v_{i+1}, w_{i+1}, z_{i+1})$ and $(t, v_0, w_0, z_0) = (t, v_k, w_k, z_k) = (0, 0^m, 0^m, b)$.
Let α be the vector defined by $\alpha[i] = max_{0 \leq j \leq k}\{v_j[i]\}$. We define μ from $\mathbb{N}^{d'}$ to $\mathbb{N}^d$ by $\mu(t, v, w, z) = (t, w - v + \alpha, z)$. Then, an induction on the length of the transition sequence gives that $(q, s_1) \xrightarrow{u}_{\mathcal{S}'} (q', s_2) \implies \exists s_3 \in \mathbb{N}^d, s_3 \leq_1 \mu(s_2) \wedge (q, \mu(s_1)) \xrightarrow{u}_{\mathcal{S}} (q', s_3)$.
This gives the result.

Note that we can treat a VASS as a VASS_0 where the component tested for zero is unused. We get the following corollary of lemma 1 (a similar result can be found in [5] and [4]) that we will also need to use:

Corollary 1. *Let $\mathcal{V}' = \langle Q, A, T, s_{in}\rangle$ be a VASS, $q_f \in Q$ and $\ell \in \mathbb{N}^d_\omega$. It is possible to decide whether there exists a ℓ-increasing loop on q_f in $\mathcal{V}'$.*

Lemma 2. *Let q_f be a control location.*
Testing whether there is a run of $\mathcal{S}$ visiting infinitely often q_f and on which the zero-test is fired infinitely often is decidable.

Proof. We reduce this problem to the one of lemma 1. Because $\mathcal{S}$ is normed, there is a single transition labelled by a_Z in T: $(q_z, \delta_z, a_Z, q'_z)$. We define $\mathcal{S}' = \langle Q', A, a_Z, T', s'_{in}\rangle$ of dimension $d+1$ (schematized in figure 4) by:

$$\begin{aligned}Q' &= Q \cup \{q_{pre}, q'_f\}\\ s'_{in} &= (q_{in}, (x_{in}, 0))\\ T' &= \begin{array}{l}\{(q, (\delta, 0), a, q') \mid (q, \delta, a, q') \in T \wedge q' \notin \{q_f, q_z\}\} \cup\\ \{(q, (\delta, 1), a, q_f) \mid (q, \delta, a, q_f) \in T\} \cup\\ \{(q, (\delta, 0), a, q_{pre}) \mid (q, \delta, a, q_z)\} \cup\\ \{(q_{pre}, 0^{d+1}, \varepsilon, q_z), (q_z, (0^d, -1), \varepsilon, q'_f), (q'_f, 0^{d+1}, \varepsilon, q_z)\}\end{array}\end{aligned}$$

Note that in $\mathcal{S}'$, the last component of the state contains the difference between the number of times the system visited q_f and the number of times the system visited q'_f.

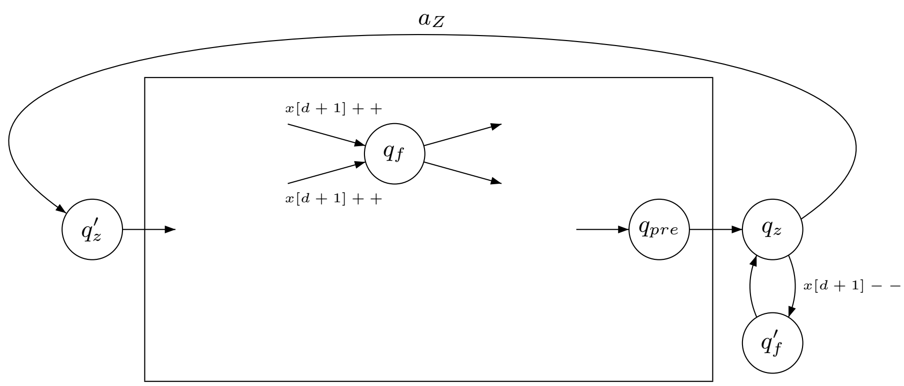

Fig. 1. Schema of the reduction

First, let us show that there is a run visiting infinitely often q_f and going through the zero-test infinitely often in $\mathcal{S}$ if and only if there is a run visiting infinitely often q'_f in $\mathcal{S}'$.

- $\Rightarrow$ Let us assume there is a run in $\mathcal{S}$ that visits infinitely often q_f and that goes infinitely often through the zero-test. This run is also a valid run in $\mathcal{S}'$ because we only added places and a counter that is only incremented by actions of $\mathcal{S}$. Now, we alter this run by inserting as many loops $q_z \rightleftharpoons q_f$ as possible before each zero-test. This new run fulfills $x[d+1] = 0$ infinitely often, and because this counter marks the difference between the number of passages in q_f and the number of passages in q'_f, this means q'_f is visited infinitely often.
- $\Leftarrow$ Let us assume there is a run in $\mathcal{S}'$ that visits q'_f infinitely often. Because of the $x[d+1]$ counter, this run visits q_f infinitely often. Moreover, because q'_f can only be reached by q_z, that can only go to q'_f or through the zero-test, and that the loop $q_z \rightleftharpoons q_f$ can only be done a finite number of times, if q'_f is visited infinitely often on a run, then the zero-test is also fired an infinite number of times. Hence, we have a run of $\mathcal{S}'$ that visits infinitely often q_f and on which the zero-test is fired infinitely often. Now, if we remove in this run the loops $q_z \rightleftharpoons q_f$, we get a run using only transitions of $\mathcal{S}$, and removing the additionnal counter can't make this run non-fireable, so we get a run of $\mathcal{S}$ that visits infinitely often q_f and the zero-test.

Now, assume we have a run visiting infinitely often q'_f. We have an infinite sequence $(x_i)_i$, $x_i \in \mathbb{N}^{d+1}$ such that for all $i \in \mathbb{N}$, $(q_f, x_i) \xrightarrow{*} (q_f, x_{i+1})$. By well-order of $\mathbb{N}^{d+1}$, there exists $i < j$ such that $x_i \leq x_j$. Also, because the zero-test is fired after the iterations $q'_f \rightleftharpoons q_z$, this means that $x_i[1] = 0$. So, we have a run visiting infinitely often q'_f if and only if there exists (q_f, x) reachable state with $x[1] = 0$, y with $x \leq_0 y$ and $u \in A^+$ such that $(q_f, x) \xrightarrow{u} (q_f, y)$ (the "if" part is immediate).

Because the first counter is necessarily 0 on the q'_f control location (assuming an infinite run) and because our system is monotonic with respect to $\leq_0$ (proposition 2), we can replace "(q_f, x) reachable state" by "(q_f, x) coverable state" in the previous equivalence. Hence, our problems reduce to decide whether there exists a ℓ-increasing loop on q_f, for ℓ a maximal element of $Cover(\mathcal{S})$.
By [3], we can compute the maximal elements of $Cover(\mathcal{S})$. Then, for each such maximal element, we can use lemma 1 to get our result.

Lemma 3. *Let q_f be a control location.*
Testing whether there is a run of $\mathcal{S}$ visiting infinitely often q_f and on which the zero-test is not fired infinitely often is decidable.

Proof. Let us consider a run visiting q_f infinitely often. Because the zero-test is fired only a finite number of times, after some point, we have a run visiting q_f infinitely often without firing the zero-test. Hence, we reduce our problem to repeated control location reachability in VASS.

We make the intersection of $Cover(\mathcal{S})$ (computed through [3]) with $(\{q_f\} \times \mathbb{N}^d)$. By well-order, if q_f is visited infinitely often, then there exists $x, x' \in \mathbb{N}^d$ and $u \in (A \backslash \{a_z\})^+$ such that $(q_f, x) \xrightarrow{u} (q_f, x')$, $x \leq x'$. Detecting such an increasing loop in a VASS can be seen as a special case of lemma 1 (corollary 1), and by testing the presence of an increasing loop for each maximal element of the cover, we get our result.

Finally, we can combine lemmas 2 and 3 to get:

Theorem 3. *Let q_f be a control location.*
Testing whether there is a run of $\mathcal{S}$ visiting infinitely often q_f is decidable.

And by proposition 1,

Corollary 2. *Model-Checking LTL is decidable on VASS$_0$.*

5 Conclusion

We have shown that despite VASS$_0$ looking more expressive than VASS, another decidability result of VASS is preserved. Between the numerous decidability results that have recently been shown for VASS$_0$ and this new one, a rule of thumb seems to be that VASS$_0$ and VASS enjoy the same decidability properties, and counter-examples have yet to be found. One can wonder if the few problems (regularity of the recognized language for example) that are decidable for VASS and remain unknown for VASS$_0$ follow this rule.

However, it is interesting to note that, despite repeated control location reachability being independent from reachability for Vector Addition Systems [6], our proof requires both reachability and place-boundedness on VASS$_0$. This makes the complexity of our procedure unknown. One might wonder a proof might exist without using reachability and/or place-boundedness, or whether reachability and place-boundedness can actually be reduced to LTL. We leave these questions for future work.

References

1. P. Abdulla, K. Cerans, B. Jonsson, and Y.-K. Tsay. General decidability theorems for infinite-state systems. *Logic in Computer Science, Symposium on*, 0:313, 1996.
2. P. Abdulla and R. Mayr. Minimal cost reachability/coverability in priced timed petri nets. In L. de Alfaro, editor, *Foundations of Software Science and Computational Structures*, volume 5504 of *Lecture Notes in Computer Science*, pages 348–363. Springer Berlin / Heidelberg, 2009.
3. R. Bonnet, A. Finkel, J. Leroux, and M. Zeitoun. Place-boundedness for vector addition systems with one zero-test. In K. Lodaya and M. Mahajan, editors, *Proceedings of the 30th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'10)*, volume 8 of *Leibniz International Proceedings in Informatics*, pages 192–203, Chennai, India, Dec. 2010. Leibniz-Zentrum für Informatik.
4. E. A. Emerson and K. S. Namjoshi. On model checking for non-deterministic infinite-state systems. In *Proceedings of the 13th Annual IEEE Symposium on Logic in Computer Science*, LICS '98, pages 70–, Washington, DC, USA, 1998. IEEE Computer Society.
5. J. Esparza. On the decidability of model checking for several μ-calculi and petri nets. In S. Tison, editor, *Trees in Algebra and Programming — CAAP'94*, volume 787 of *Lecture Notes in Computer Science*, pages 115–129. Springer Berlin / Heidelberg, 1994. 10.1007/BFb0017477.
6. J. Esparza. Decidability and complexity of petri net problems : An introduction. In W. Reisig and G. Rozenberg, editors, *Lectures on Petri Nets I: Basic Models*, volume 1491 of *Lecture Notes in Computer Science*, pages 374–428. Springer Berlin / Heidelberg, 1998.
7. A. Finkel and J. Goubault Larrecq. Forward analysis for WSTS, Part I: Completions. In Susanne Albers and Jean-Yves Marion, editors, *26th International Symposium on Theoretical Aspects of Computer Science - STACS 2009*, pages 433–444, Freiburg Allemagne, 2009. IBFI Schloss Dagstuhl.
8. A. Finkel and A. Sangnier. Mixing coverability and reachability to analyze vass with one zero-test. In J. van Leeuwen, A. Muscholl, D. Peleg, J. Pokorný, and B. Rumpe, editors, *SOFSEM 2010: Theory and Practice of Computer Science*, volume 5901 of *Lecture Notes in Computer Science*, pages 394–406. Springer Berlin / Heidelberg, 2010.
9. A. Finkel and P. Schnoebelen. Well-structured transition systems everywhere! *Theoretical Computer Science*, 256(1-2):63 – 92, 2001.
10. P. Gastin and D. Oddoux. Fast ltl to büchi automata translation. In G. Berry, H. Comon, and A. Finkel, editors, *Computer Aided Verification*, volume 2102 of *Lecture Notes in Computer Science*, pages 53–65. Springer Berlin / Heidelberg, 2001.
11. R. Gerth, D. Peled, M. Y. Vardi, and P. Wolper. Simple on-the-fly automatic verification of linear temporal logic. In *Proceedings of the Fifteenth IFIP WG6.1 International Symposium on Protocol*

Specification, Testing and Verification XV, pages 3–18, London, UK, UK, 1996. Chapman & Hall, Ltd.
12. P. Habermehl. On the complexity of the linear-time μ-calculus for petri nets. In P. Azéma and G. Balbo, editors, *Application and Theory of Petri Nets 1997*, volume 1248 of *Lecture Notes in Computer Science*, pages 102–116. Springer Berlin / Heidelberg, 1997.
13. M. Hack. The equality problem for vector addition systems is undecidable. *Theoretical Computer Science*, 2(1):77 – 95, 1976.
14. S. R. Kosaraju. Decidability of reachability in vector addition systems. In *Proceedings of the fourteenth annual ACM symposium on Theory of computing*, STOC '82, pages 267–281, New York, NY, USA, 1982. ACM.
15. J. Leroux. The general vector addition system reachability problem by presburger inductive invariants. *Logic in Computer Science, Symposium on*, 0:4–13, 2009.
16. J. Leroux. Vector addition system reachability problem: a short self-contained proof. *SIGPLAN Not.*, 46:307–316, January 2011.
17. E. W. Mayr. An algorithm for the general petri net reachability problem. In *Proceedings of the thirteenth annual ACM symposium on Theory of computing*, STOC '81, pages 238–246, New York, NY, USA, 1981. ACM.
18. K. Reinhardt. Reachability in petri nets with inhibitor arcs. *Electronic Notes in Theoretical Computer Science*, 223:239 – 264, 2008. Proceedings of the Second Workshop on Reachability Problems in Computational Models (RP 2008).

A Additionnal reductions

Definition 6. *Let $\mathcal{S}_1 = \langle Q_1, A, a_Z, T_1, s_{in1}\rangle$ and $\mathcal{S}_2 = \langle Q_2, A, a_Z, T_2, s_{in2}\rangle$ be two VASS$_0$ of respective dimensions d_1 and d_2. $\mathcal{S}_1$ and $\mathcal{S}_2$ are weakly bisimilar if there exists a relation $\sim \subseteq (Q_1 \times \mathbb{N}^{d_1}) \times (Q_2 \times \mathbb{N}^{d_2})$ such that:*

- $s_{in1} \sim s_{in2}$
- $\begin{cases} s_1 \sim s_2 \\ s_1 \xrightarrow{a}_{\mathcal{S}_1} s_1' \end{cases} \implies \exists s_2' \in Q \times \mathbb{N}^{d_2} \begin{cases} s_2 \xrightarrow{a}_{\mathcal{S}_2} s_2' \\ s_1' \sim s_2' \end{cases}$
- $\begin{cases} s_1 \sim s_2 \\ s_2 \xrightarrow{a}_{\mathcal{S}_2} s_2' \end{cases} \implies \exists s_1' \in Q \times \mathbb{N}^{d_2} \begin{cases} s_1 \xrightarrow{a}_{\mathcal{S}_1} s_1' \\ s_1' \sim s_2' \end{cases}$

Note that we are using weak bisimilarity because of the presence of epsilon-transitions. Satisfiability of a LTL formula is stable by weak bisimilarity[1].
We provide here a quick proof of a well known reduction of VASS$_0$.

Proposition 3. *Let $\mathcal{S}$ be a VASS$_0$. There exists a VASS$_0$ $\mathcal{S}'$ weakly bisimilar to $\mathcal{S}$ such that there exists a unique $(q_z, a_Z, q_z', \delta_z) \in T$.*

[1] For a survey of weak bisimilarity and other notions of system equivalence, one might look at "The linear time-branching time spectrum II: The semantics of sequential processes with silent moves", by RJ. van Glabbeek

Proof. If $\mathcal{S}$ has no such transition, we can simply add new unreachable control states and add the required transition, so we will only consider the case of $\mathcal{S}$ having more than one transition.
Let $\mathcal{S} = \langle Q, A, a_Z, T, (q_{in}, x_{in})\rangle$ be a VASS_0 of dimension d.
Let $T_z = \{(q_{z,i}, a_Z, q'_{z,i}, \delta_{z,i}) \mid 0 \leq i \leq p\}$ be the transitions of $\mathcal{S}$ using the zero-test. Let T_0 be the other transitions. $T = T_0 \uplus T_z$. We define $\mathcal{S}' = \langle Q', A, a_Z, T', s'_{in}\rangle$ of dimension $d + 2$ by:

$$\begin{aligned} Q' &= Q \uplus \{q_z, q'_z\} \\ T' &= \begin{array}{l} \{(q, a, q', (\delta, 0, 0)) \mid (q, a, q', \delta) \in T_0)\}\cup \\ \{(q_{z,i}, \varepsilon, q_z, (\delta_{z,i}, i, p - i) \mid 1 \leq i \leq p\}\cup \\ \{(q'_z, \varepsilon, q'_{z,i}, (0^d, -i, -(p - i)) \mid 1 \leq i \leq p\}\cup \\ \{(q_z, a_z, q'_z, 0^{d+2})\} \end{array} \\ s'_{in} &= (q_{in}, (x_{in}, 0, 0)) \end{aligned}$$

We note that we have the invariant that the last two components are always zero in all states of Q. Bisimilarity comes easily from that.