Concurrent Games on VASS with Inhibition

Béatrice Bérard, Serge Haddad, Mathieu Sassolas, Nathalie Sznajder

June 2012

Research report LSV-12-10

Laboratoire Spécification & Vérification

École Normale Supérieure de Cachan
61, avenue du Président Wilson
94235 Cachan Cedex France

Concurrent Games on VASS with Inhibition*

B. Bérard[1], S. Haddad[2], M. Sassolas[3], and N. Sznajder[1]

[1] Université Pierre & Marie Curie, LIP6/MoVe, CNRS UMR 7606, Paris, France
{Beatrice.Berard,Nathalie.Sznajder}@lip6.fr
[2] ENS Cachan, LSV, CNRS UMR 8643 & INRIA, Cachan, France
Serge.Haddad@lsv.ens-cachan.fr
[3] Département d'Informatique, Université Libre de Bruxelles, Bruxelles, Belgium
mathieu.sassolas@ulb.ac.be

Abstract. We propose to study concurrent games on a new extension of Vector Addition Systems with States, where inhibition conditions are added for modeling purposes. Games are a well-suited framework to solve control problems, and concurrent semantics reflect realistic situations where the environment can always produce a move before the controller, although it is never required to do so. This is in contrast with previous works, which focused mainly on turn-based semantics. Moreover, we consider asymmetric games, where environment and controller do not have the same capabilities, although they both have restricted power. In this setting, we investigate reachability and safety objectives, which are not dual to each other anymore, and we prove that (i) reachability games are undecidable for finite targets, (ii) they are 2-EXPTIME-complete for upward-closed targets and (iii) safety games are co-NP-complete for finite, upward-closed and semi-linear targets. Moreover, for the decidable cases, we build a finite representation of the corresponding controllers.

1 Introduction

Context. Games on infinite structures, and their relation to control theory, have been largely studied in the last ten years [1], [16], [17], [11], [12], [19], [18], [4], [6]. Given a plant in an environment and a specification, controllability asks if there exists a controller such that the controlled plant satisfies the specification. When the answer is positive, the synthesis problem requires to build a controller. This problem can be expressed as a game with two players, environment and controller, and the question becomes the existence (and construction) of a controller strategy to win the game.

* Work partially supported by projects CoChaT (DIGITEO-2009-27HD), ImpRo (ANR-2010-BLAN-0317), ERC Starting Grant inVEST (279499) and the European Union Seventh Framework Programme [FP7/2007-2013] under grant agreement 257462 HYCON2 NOE.

In this context, various parameters come into play. The underlying models can be continuous or discrete transition systems, the latter being those considered here. The game semantics can be turn-based or concurrent, with identical or asymmetrical rules for the two players, with or without the ability to waive a move, and so on. Finally, different winning objectives can be considered: from basic reachability or avoidance objectives (w.r.t. some target set S of system configurations) to general LTL specifications [19,18,2]. In addition, the target set S can be specified in several ways: a finite set, an upward-closed set (with respect to some ordering), a set of (bounded) linear constraints, a semi-linear set, etc.

Related Work. In [12,11], the underlying models are Symbolic Transition Systems or Assignment Program Models with turn-based semantics and avoidance objectives, for which controllability is undecidable. Abstract interpretation techniques are proposed to compute over-approximations of the subset of unsafe states [12] and decidability results are obtained for particular cases, among them Petri nets with upward-closed targets [11]. In [1,16,17], the authors introduce monotonic game structures, which also include Petri nets. The games are turn-based and symmetrical, with safety, reachability and parity objectives for finite and upward-closed target sets. While the problems are still undecidable, the authors investigate subclasses like B-game structures [16,17] or B-downward closed games [1] (where A and B are the two players), thus breaking the symmetry, and they establish decidability results for these games.

Vector Addition Systems with States (VASS) were also used as a model for control and two-player games. A possibly infinitely branching extension of VASS is studied in [4], again with a turn-based symmetrical game, reachability objectives, and a target set containing configurations where one of the counters is null. Decidability is obtained in this case, with an EXPSPACE upper bound, while adding the selection of control states again brings undecidability. Among other results, the complexity bound mentioned above is improved in [6] in the more general framework of Energy and Mean-Payoff games, which is another way of dealing with VASS with specific targets corresponding to minimal or mean values for the counters.

Contribution. In this work, we consider another extension of VASS, called VASSI, obtained by adding inhibition conditions, which correspond to inhibitor arcs in Petri nets (as is done in [2] with boundedness constraints). This feature is useful for modeling purposes: for instance, consider the cooling system of a plant, where temperature can increase when the water level is below some threshold. This can be described by an environment's transition with inhibition conditions (see Fig. 1 in Section 2).

Concerning semantics, we consider concurrent and asymmetric games: we argue that such games are more realistic than turn-based symmetric games in the context of controllability problems, since usually the environment can always produce a move, whatever the controller is willing to do. Hence we mean concurrent here in a sense similar to the setting of timed games [7], where a player can "surprise" its opponent by playing faster. Along the same line, no player is forced to play. Moreover, environment and controller do not have the same capabilities. They both have restricted power but in an asymmetrical way. Our model is described in Section 2.

Note that in this setting, safety and reachability are not dual objectives with respect to the two players. Also, contrary to [1,16,17], the games are not monotonic anymore. We prove in Section 3 that reachability games are undecidable for finite target sets (hence also for semi-linear sets) and 2EXPTIME-complete for upward-closed targets. On the other hand, we establish in Section 4 that safety games are co-NP-complete for semi-linear targets, as well as finite and upward-closed sets (see summary in Table 1). For decidable games, we also provide finite representations of controllers, the one for safety games implementing a most permissive strategy.

Table 1. Summary of results.

Objective/Target	Finite	Semi-linear	Upward-closed
Reachability	Undecidable $\Longrightarrow$	Undecidable	2-EXPTIME-complete
Safety	co-NP-complete	co-NP-complete	co-NP-complete

2 Games on VASS with Inhibition Conditions

We denote by A^* (resp. A^ω) the set of finite (resp. infinite) sequences of elements of a set A, with ε the empty sequence, and $|w|$ the length of $w \in A^*$. A finite sequence u is a *prefix* of w, if there is a sequence v such that $uv = w$. We write $A^+ = A^* \setminus \{\varepsilon\}$ and $A^\infty = A^* \cup A^\omega$. The set of all subsets of A is denoted by $\mathcal{P}(A)$ and $\uplus$ denotes the disjoint union of subsets.

We write $\mathbb{Z}$ (resp. $\mathbb{N}$) for the set of integers (resp. nonnegative integers). For $n \in \mathbb{N}$, let $[n]$ denote the set $\{1, \ldots, n\}$. For a vector $v = (v_j)_{j \in [n]} \in \mathbb{Z}^n$ and for $i \in [n]$, let $v(i) = v_i$ be the ith component of v and $v[i] = (v_j)_{j \in [i]}$ be the projection of v onto its first i components. The vector with all components equal to 0 is denoted by $\mathbf{0}$. Given $v_1, v_2 \in \mathbb{N}^n$, operations are defined componentwise: $v_1 \geq v_2$ if $v_1(i) \geq v_2(i)$ for all $i \in [n]$, and $v_1 + v_2$ is defined by $(v_1 + v_2)(i) = v_1(i) + v_2(i)$ for all $i \in [n]$.

We extend the definition of Vector Addition System with States to include inhibition conditions.

Definition 1 (Vector Addition Systems with States and Inhibition conditions). *A* Vector Addition System with States and Inhibition conditions *(VASSI) is a tuple* $\mathcal{V} = (Q, n, T, \alpha, \beta, \delta, Inh)$ *where*

- Q *is a finite set of states,*
- $n \in \mathbb{N}$ *is the number of counters (called the dimension),*
- T *is the set of transitions,* $\alpha, \beta : T \to Q$ *associate respectively with each* $t \in T$*, its source and target states,*
- $\delta : T \to \mathbb{Z}^n$ *is the displacement function,*
- *and* $Inh : T \to (\mathbb{N} \setminus \{0\} \cup \{\infty\})^n$ *is the inhibition function.*

A *configuration* of a VASSI $\mathcal{V} = (Q, n, T, \alpha, \beta, \delta, Inh)$ is a pair $c = (q, m) \in \mathcal{C} = Q \times \mathbb{N}^n$. The semantics of $\mathcal{V}$ is given by the transition system $\mathcal{T}_{\mathcal{V}} = (\mathcal{C}, \to)$, where $\to \subseteq \mathcal{C} \times \mathcal{C}$ is the transition relation defined by $(q, m) \to (q', m')$ if and only if there is a transition $t \in T$ such that $\alpha(t) = q$, $\beta(t) = q'$, $m < Inh(t)$ and $m' = m + \delta(t)$; note that since $m' \in \mathbb{N}^n$, $m + \delta(t) \geq \mathbf{0}$. In such a case, we say that t is *fireable* in (q, m) and we may also write the transition as $(q, m) \xrightarrow{t} (q', m')$.

A *run* of $\mathcal{T}_{\mathcal{V}}$ (or, equivalently, of $\mathcal{V}$) is a sequence of configurations $\rho = c_0 c_1 \cdots \in \mathcal{C}^\infty$ such that $c_i \to c_{i+1}$ for all $0 \leq i < |\rho|$.

Given $c, c' \in \mathcal{C}$ two configurations, we say that c' is *reachable* from c if there is a finite run $c_0 c_1 \ldots c_k$ of $\mathcal{V}$ with $c = c_0$ and $c' = c_k$. Like above, we may also write $c \xrightarrow{\tau} c'$, indicating the corresponding sequence of transitions $\tau = t_1 t_2 \ldots t_k$, which forms what we call a *fireable path* in the underlying graph (Q, T).

Our games are played by two players (environment and controller) on a subclass of VASSI, where the set of transitions is partitioned into controllable and uncontrollable transitions, with the additional constraint that uncontrollable transitions can only increase the values of the counters (as in [16,17]) and controllable transitions cannot be inhibited:

Definition 2 (Asymmetric VASSI). *An* Asymmetric VASSI *(shortly AVASSI) is a VASSI where the set of transitions is partitioned into two subsets:* $T = T_c \uplus T_u$*, and such that* $\delta(T_u) \subseteq \mathbb{N}^n$ *and* $Inh(T_c) = \{(\infty)^n\}$*.*

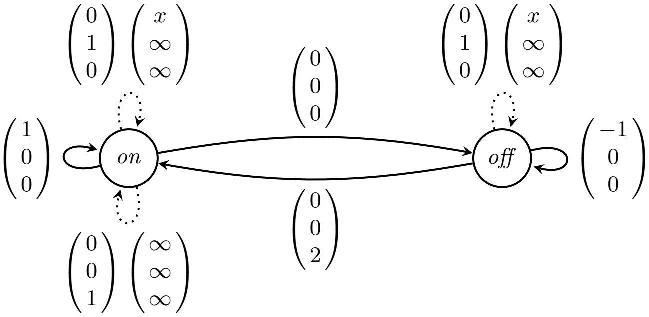

on
off
$\begin{pmatrix}0\\1\\0\end{pmatrix}\begin{pmatrix}x\\\infty\\\infty\end{pmatrix}$
$\begin{pmatrix}0\\1\\0\end{pmatrix}\begin{pmatrix}x\\\infty\\\infty\end{pmatrix}$
$\begin{pmatrix}1\\0\\0\end{pmatrix}$
$\begin{pmatrix}0\\0\\0\end{pmatrix}$
$\begin{pmatrix}0\\0\\2\end{pmatrix}$
$\begin{pmatrix}-1\\0\\0\end{pmatrix}$
$\begin{pmatrix}0\\0\\1\end{pmatrix}\begin{pmatrix}\infty\\\infty\\\infty\end{pmatrix}$

Fig. 1. Cooling system as an AVASSI. Solid edges belong to the controller while dotted edges belong to the environment.

If we consider that environment sends events to the system through a unidirectional channel, the counters can represent the number of environment events the system is *aware of* that have not been handled yet (actual content of events is abstracted away). The system does not necessarily observe all the events in the channel (due to delay of transmission from a sensor for instance), hence it cannot test the value 0 of the counter (which corresponds to the fact that the transition cannot be inhibited).

To illustrate this definition, we give another example where our model is appropriate: the case of a (simple) cooling system is depicted by the AVASSI in Fig. 1, where the three counters represent respectively the amount of water in a tank, the temperature, and the cost associated with pumping water into the tank. A transition of the controller is represented by a solid line and labeled by a column vector corresponding to the displacement function δ. A transition of the environment is represented by a dotted line and labeled by two column vectors corresponding to the displacement function δ and the inhibition function Inh. When the pump is *on*, the controller can add water into the tank. The environment can increase the global cost. When the pump is *off*, the controller can choose to empty the tank. In both cases, when the water gets below some threshold x, cooling is prevented, which is described by an environment's transition with inhibition condition that increases the temperature counter. Of course, this toy example could be made more realistic.

Strategies. Given an AVASSI $\mathcal{V}$, a *strategy* for the controller is a mapping $f : \mathcal{C}^+ \to 2^{T_c}$ that gives the subset of fireable transitions of T_c permitted after a sequence of configurations. A strategy f is *memoryless* if $f(\rho_1 \cdot c) =$

$f(\rho_2 \cdot c)$, for all $\rho_1, \rho_2 \in \mathcal{C}^*,\ c \in \mathcal{C}$. In this case, we may simply define it as a mapping $f : \mathcal{C} \to 2^{T_c}$.

Outcome of a Strategy. Given an AVASSI $\mathcal{V}$ and a strategy $f : \mathcal{C}^+ \to 2^{T_c}$, a run $\rho = c_0 c_1 \cdots \in \mathcal{C}^\infty$ is *f-consistent* (and also called an *f-run*) if, at each step, either a transition permitted by the strategy has been fired, or the environment has played instead, *i.e.* for all $0 < i < |\rho|$, there exists a transition $t \in f(c_0 \ldots c_{i-1}) \cup T_u$ such that $c_{i-1} \xrightarrow{t} c_i$.

An f-run ρ is *f-maximal* if it is infinite or such that $f(\rho) = \emptyset$. Given a configuration $c \in \mathcal{C}$, we define *Outcome*$(f, \mathcal{V}, c)$ as the set of f-maximal f-runs of $\mathcal{V}$ that start in c.

Winning Condition and Winning Strategies. Given a AVASSI $\mathcal{V}$, a *winning condition* is a set of sequences $W \subseteq \mathcal{C}^\infty$. A run is *winning* if it belongs to W and a strategy f is *winning* from configuration $c \in \mathcal{C}$ for W if *Outcome*$(f, \mathcal{V}, c) \subseteq W$.

Control Problem. The control problem for AVASSI can be expressed as follows: given an AVASSI $\mathcal{V}$, an initial configuration $c_0 \in \mathcal{C}$, and a winning condition W, does there exist a winning strategy for the controller for W from c_0? We consider in this work two variants of winning conditions: given a AVASSI, and a set of configurations $S \subseteq \mathcal{C}$ (the *target*),

- a *reachability objective* is defined by $W = \mathcal{C}^* \cdot S \cdot \mathcal{C}^\infty$,
- a *safety objective* is defined by $W = (\mathcal{C} \setminus S)^\infty$.

In the rest of the paper, we call these problems respectively *reachability game* and *safety game* and we consider three types of targets: finite sets, upward-closed sets, and semi-linear sets of configurations.

Upward-closed Sets. Let $(A, \preceq)$ be an ordered set. A subset $S \subseteq A$ is *upward-closed* if for all $a_1 \in S$ and $a_2 \in A$, if $a_1 \preceq a_2$, then $a_2 \in S$. Such a set can be represented by a finite set of minimal elements.

In this work, we consider upward-closed sets of configurations with respect to the covering order on configurations of an AVASSI: (q_1, m_1) *covers* (q_2, m_2), written $(q_1, m_1) \succeq (q_2, m_2)$, if $q_1 = q_2$ and $m_1 \geq m_2$.

Semi-linear Sets. A *linear set* is a subset of $\mathbb{N}^n$ (for $n > 0$) of the form $\{v + k_1 u_1 + \cdots + k_p u_p \mid k_1, \cdots, k_p \in \mathbb{N}\}$ where $v, u_1, \cdots, u_p \in \mathbb{N}^n$. A *semi-linear set* is a finite union of linear-sets. Semi-linear sets are closed by intersection, complementation, and application of a linear mapping. Moreover, emptiness of a semi-linear set is decidable. Remark that finite sets and upward-closed sets are particular cases of semi-linear sets.

In the sequel, we consider semi-linear sets over the set of configurations seen as $\mathbb{N}^{Q \uplus [n]}$: a configuration (q, m) is represented by the vector $(\mathbf{1}_q, m)$, with $\mathbf{1}_q$ the vector defined by $\mathbf{1}_q(q) = 1$ and $\mathbf{1}_q(q') = 0$ for $q' \neq q$.

3 Reachability Games

Finite Targets. In the simplest case where the target is a finite set of configurations, reachability games are undecidable.

Theorem 3. *Reachability games are undecidable on AVASSI for finite targets.*

Proof. The proof relies on the encoding of a two-counter machine, on which it is well known that the halting problem is undecidable. Recall that such a machine consists in two counters a and b and a set of labeled instructions which can be (for $e \in \{a, b\}$):

- an incrementation $e := e + 1$; goto ℓ',
- or a decrementation if $e > 0$ then ($e := e - 1$; goto ℓ') else goto ℓ''.

Given a two-counter machine M, let $\mathcal{V}_M = (Q, 2, T, \alpha, \beta, \delta, \mathit{Inh})$ be an AVASSI, in which the two counters encode those of the machine and Q contains a state for each instruction ℓ, along with some additional states. The simulation of the two-counter machine will be ensured by transitions of $\mathcal{V}_M$ in the following way.

An instruction of the form ($\ell : a := a + 1$; goto ℓ') is easily encoded by a (controllable) transition t from ℓ to ℓ' with $\delta(t) = (1; 0)$ (respectively $(0; 1)$ to increment the counter b). To encode decrementation of a, and hence zero-test of a counter, we use uncontrollable transitions, as described in Fig. 2 (decrementation of counter b is symmetrical), where dotted transitions are environment transitions and therefore bear the inhibition vector next to the displacement vector. The target is given by the single configuration $(\mathit{end}, \mathbf{0})$.

The widget of Fig. 2 thus works as follows: starting from configuration $(\ell, (n; m))$, it is environment's choice to decide whether counter a is empty (*i.e.* $n = 0$) or not. Observe however that the inhibition condition on the transition from ℓ to ℓ'' makes it impossible to go to ℓ'' when counter c is greater than 0. On the other hand, assume that the environment decides to go to NZ_ℓ when the value of the counter is in fact 0. Then, after taking the transition to end_a and decrementing b until its value reaches 0, the winning configuration can be reached by taking the transition from end_a to end. If the environment decides to block the game in the state ℓ by taking no transition, the controller can go to the state halt in order to empty both counters. Then, it can again reach the winning configuration. Finally, if the environment plays faithfully, and if M halts, then the controller can also reach the state halt by playing faithfully too, and then again, after emptying the counters, reach the configuration $(\mathit{end}, \mathbf{0})$.

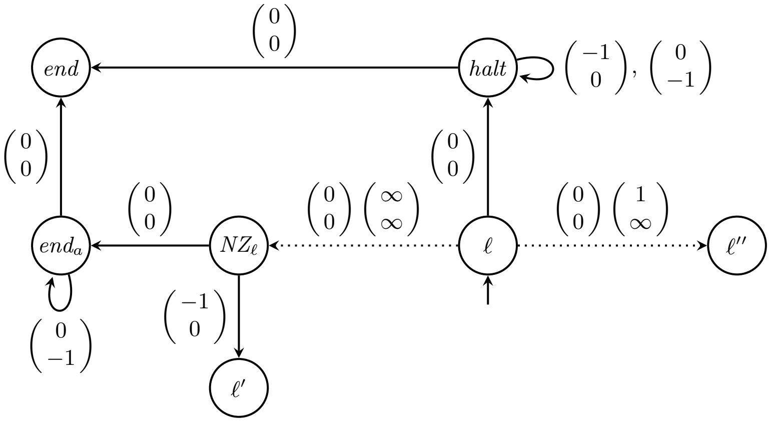


Fig. 2. Widget to encode counter a decrementation within a control game.

Conversely, if the machine does not halt, assume that the environment plays faithfully. Then if the controller simulates the correct run of M, it won't reach *halt* hence won't reach *end* either. Alternatively, if the controller takes the transition from NZ_ℓ to end_a while $n > 0$, it can never reach a configuration $(end, (0; m))$.

We can conclude that there is a winning strategy for the controller if and only if the two-counter machine halts. □

A direct consequence of this result is that the control problem for reachability objective with semi-linear targets is also undecidable.

Upward-closed Targets. We now consider the case of upward-closed targets:

Theorem 4. *Reachability games on AVASSI with upward-closed targets are 2-EXPTIME-complete.*

Before giving the proof of Theorem 4, we establish in Proposition 5 (reminiscent of [15]) an upper bound on the "size" of the optimal winning strategy, when it exists. By "size", we mean the depth of the tree of possible configurations encountered while playing according to this strategy, where branches stop growing as soon as they reach a winning configuration.

In this section, we say that a run is a min-winning f-run if it is winning while none of its prefixes is. It is sufficient to consider only those runs, since any suffix starting from a configuration covering the target is irrelevant to the winning condition.

The input consists of an AVASSI $\mathcal{V} = (Q, n, T, \alpha, \beta, \delta, \mathit{Inh})$, an initial configuration $c_0 \in \mathcal{C}$, and an upward-closed set as target, given by the finite set of its minimal elements $B = \{b_1, \ldots, b_m\}$. Its size is defined as follows. The space needed to describe $\alpha : T \to Q$ and $\beta : T \to Q$ is $|\alpha| = |\beta| = |T| \cdot \log_2(|Q|)$, whereas the size needed for $\delta : T \to \mathbb{Z}^n$ and $\mathit{Inh} : T \to (\mathbb{N}^n \setminus \{0\})$ is respectively $|T| \cdot n \cdot (\log_2(\delta_{\max}) + 1)$ and $|T| \cdot n \cdot \log_2(\mathit{Inh}_{\max})$, with $\delta_{\max} = 1 + \max_{t \in T; i \in [n]}(|\delta(t)(i)|)$ and $\mathit{Inh}_{\max} = 1 + \max_{t \in T; i \in [n]}\{\mathit{Inh}(t)(i) \mid \mathit{Inh}(t)(i) < \infty\}$. Then, the size needed to describe the AVASSI $\mathcal{V}$ is $|\mathcal{V}| = \log_2 |Q| + \log_2(n) + \log_2 |T| + |\alpha| + |\beta| + |\delta| + |\mathit{Inh}|$. The size of the target base is $|B| = \Sigma_{i=1}^{m} |b_i|$ with $|b_i| = n \cdot \log_2(max_{1 \leq j \leq n}(1 + b_i(j)))$. Finally we denote by $\mathcal{K} = |\mathcal{V}| + |B| + |c_0|$ the size of the input for our problem, with $|c_0| = \log_2 |Q| + \sum_{i \in [n]} \log_2(c_0(i))$.

Proposition 5. *For an AVASSI $\mathcal{V}$ and an upward-closed target described by $B = \{b_1, \ldots, b_m\}$, there is a winning strategy for the reachability game if and only if there is a winning strategy f such that all the min-winning f-runs have length less than or equal to $2^{\mathcal{K}^{\mathcal{K}+1}}$.*

Proof. We proceed inductively on the AVASSI obtained by projecting onto the p first counters and removing transitions of the environment that contained inhibition conditions on the omitted counters. Formally for $p \leq n$, let $\mathcal{V}_p = (Q, p, T_p, \alpha_p, \beta_p, \delta_p, \mathit{Inh}_p)$, where $T_p = T_c \uplus \{t \in T_u \mid \mathit{Inh}(t)(i) = \infty$, for all $p < i \leq n\}$, α_p and β_p are the functions α and β restricted on T_p, and δ_p and Inh_p are respectively the functions δ and Inh restricted to T_p and projected onto the first p dimensions. We set $\mathcal{C}_p = Q \times \mathbb{N}^p$. We say that a run (resp. strategy) is p-winning if it is winning in $\mathcal{V}_p$ for the projection of B (minimal elements of the target) on the first p components. In particular, n-winning means winning.

A run $\rho_p = c_1 \ldots c_k \in \mathcal{C}_p^+$ of $\mathcal{V}_p$ is *p-covering* if it is a minimal p-winning run: c_k covers $b[p]$ for some $b \in B$ and for all $i < k$, for all $b \in B$, c_i does not cover $b[p]$. Note that any p-winning run starts with a p-covering run.

Given $c \in \mathcal{C}_p$ and $f : \mathcal{C}_p^+ \to 2^{T_c}$ a strategy, we define $\mathrm{size}(f, p, c) = \max\{|\sigma| \mid \sigma$ is a prefix of $\rho, \rho \in \mathit{Outcome}(f, \mathcal{V}_p, c)$ and σ is p-covering$\}$ if f is p-winning from c, and $\mathrm{size}(f, p, c) = \infty$ otherwise. From a configuration c, a strategy f reaches the target (in $\mathcal{V}_p$) in at most $\mathrm{size}(f, p, c)$ steps (which can be infinite if the strategy f is not p-winning).

A strategy f is *(p, c)-optimal* if $\mathrm{size}(f, p, c) \leq \mathrm{size}(f', p, c)$ for any strategy $f' : \mathcal{C}_p^+ \to 2^{T_c}$. We denote by $f_{p,c}$ a (p, c)-optimal strategy. Note that since the objective here is reachability, $f_{p,c}$ can be assumed memoryless. If it is not, it is possible to define another (p, c)-optimal strategy that is memoryless in the following way: if $f_{p,c}$ is winning, for all $d \in \mathcal{C}$, we let $f'_{p,c}(d) = f_{p,c}(\sigma d)$ where σd is one of the longest $f_{p,c}$-run having not covered the target yet. If $f_{p,c}$ is not winning, we let $f'_{p,c}(d) = f_{p,c}(\sigma d)$ for some $f_{p,c}$-run σ.

We now assume that there exists a winning strategy from the initial configuration. In the rest of this proof, we therefore consider only configurations for which there exists a winning strategy: $\mathcal{C}_p^{\mathrm{w}} = \{c \in \mathcal{C}_p \mid \exists f,\ p\text{-winning from } c\}$. Let

$$\ell(p) = \max\{\mathrm{size}(f_{p,c}, p, c) \mid c \in \mathcal{C}_p^{\mathrm{w}},\ f_{p,c} \text{ is a } p\text{-winning strategy from } c\}$$

be the maximal number of steps required to win in $\mathcal{V}_p$ with an optimal winning strategy.

In order to bound $\ell(n)$, we compute by induction on $p \leq n$ an upper bound for $\ell(p)$. Namely, we show that $\ell(0) \leq |Q|$ and $\ell(p+1) \leq (2^{\mathcal{K}})^{p+2} \cdot (\ell(p)+1)^{p+1} + \ell(p)$.

Case $p = 0$. In this case, we prove that $\ell(0) \leq |Q|$. Indeed, here the goal of the controller is to reach a given state, since the value of all counters is discarded. Let $c \in \mathcal{C}$ and f be a memoryless $(0, c)$-optimal strategy and suppose there exists a 0-covering f-run $\sigma = q_0 \cdots q_k$ of length greater than $|Q|$. Then there exists $i < j < k$ such that $q_i = q_j$. In that case, $q_0 \cdots (q_i \cdots q_j)^\omega$ is also a f-maximal f-run and it is not 0-covering and f is not winning from c. Then, $\ell(0) = \max\{\mathrm{size}(f_{i,c}, i, c) \mid c \in \mathcal{C},\ f_{i,c}\} \leq |Q|$.

Recurrence relation. Let $c \in \mathcal{C}_{p+1}^{\mathrm{w}}$ and $f_{p+1,c}$ be a memoryless $(p+1, c)$-optimal strategy. Let $\sigma = c_1 \dots c_k \in \mathcal{C}_{p+1}^{+}$ be the longest $p+1$-covering $f_{p+1,c}$-run, with $c_1 = c$. Remark that since $f_{p+1,c}$ is $(p+1, c)$-optimal, it is also $(p+1, c_i)$-optimal for $1 \leq i \leq k$. Otherwise, consider f' $(p+1, c_i)$-optimal strategy; replacing the choices of $f_{p+1,c}$ by those of f' for all c_j $(i \leq j \leq k)$ yields another strategy with a shorter longest $p+1$-covering run, contradicting optimality of $f_{p+1,c}$. We now consider two cases: either the values of all counters along the run are below the bound $2^{\mathcal{K}} \cdot (\ell(p)+1)$, or this bound is exceeded.

Case 1. Assume that $c_i(j) \leq 2^{\mathcal{K}} \cdot (\ell(p) + 1)$, for all $1 \leq i \leq k$ and $1 \leq j \leq p+1$. Then, since the strategy is memoryless, all the c_i are different, for $1 \leq i \leq k$ (otherwise there would be a loop and the same reasoning as in the case $p = 0$ above would apply, hence the environment would win). A combinatorial argument then yields $|\sigma| \leq |Q| \cdot (2^{\mathcal{K}} \cdot (\ell(p) + 1))^{p+1} \leq (2^{\mathcal{K}})^{p+2} \cdot (\ell(p) + 1)^{p+1}$.

Case 2. Otherwise, there is some $1 \leq i \leq k$, some $1 \leq j \leq p+1$ such that $c_i(j) > 2^{\mathcal{K}} \cdot (\ell(p)+1)$. Let i be the smallest index satisfying this property, and assume, without loss of generality, that $c_i(p+1) > 2^{\mathcal{K}} \cdot (\ell(p)+1)$ (the order of the counters is irrelevant, then we can assume that the counter violating the construction is exactly $p+1$). We write $\sigma = \sigma_1\sigma_2$ with $\sigma_1 = c_1 \cdots c_{i-1}$, and $\sigma_2 = c_i \cdots c_k$. Since σ_1 falls in the preceding case, we know that $|\sigma_1| \leq (2^{\mathcal{K}})^{p+2} \cdot (\ell(p)+1)^{p+1}$.

Let f_p be an $(p, c_i[p])$-optimal strategy. We build the corresponding strategy f_p^* on $\mathcal{V}_{p+1}$ in the following way: $f_p^*(c) = f_p(c[p])$ for all $c \in \mathcal{C}_{p+1}$.

Let ρ be an f_p^*-run starting in c_i of length $|\rho| \leq \ell(p)$. We show by induction on $|\rho|$ that $\rho[p]$ is an f_p-run. The base case is trivial since the empty execution is consistent with any strategy. Consider $\rho = \tau \cdot c \cdot c'$ a f_p^*-run of $\mathcal{V}_{p+1}$ of length smaller than $\ell(p)$ with $t \in T$ such that $c \xrightarrow{t} c'$. By induction, τc is an f_p-run. If $t \in T_p$ then, either $t \in T_c$ and by definition of f_p^*, $\tau[p] \cdot c[p] \cdot c'[p]$ is also an f_p-run, or $t \in T_u$ and the same holds true (since the environment can always fire transitions). Now if $t \in T_{p+1} \setminus T_p$, it means that $t \in T_u$ and $\mathit{Inh}(t)(p+1) < \infty$. However, at each step, the value of counter $p+1$ cannot be decreased by more than $\delta_{\max}$. In addition $|\tau| < \ell(p)$, and $c_i(p+1) > 2^{\mathcal{K}} \cdot (\ell(p)+1)$, with $2^{\mathcal{K}} > \delta_{\max} + \mathit{Inh}_{\max}$. Therefore:

$$\begin{aligned} c(p+1) &> c_i(p+1) - \ell(p) \cdot \delta_{\max} \geq 2^{\mathcal{K}} \cdot (\ell(p)+1) - \ell(p) \cdot \delta_{\max} \\ &\geq \delta_{\max} \cdot \ell(p) + \mathit{Inh}_{\max} - \ell(p) \cdot \delta_{\max} \geq \mathit{Inh}_{\max} \geq \mathit{Inh}(t)(p+1) \end{aligned}$$

and t was not fireable from c in $\mathcal{V}_{p+1}$ either.

Then, since f_p is winning, one can see that any f_p^*-run ρ starts with a prefix τ of length smaller than $\ell(p)$ such that $\tau[p]$ is p-covering. Again, since $c_i(p+1) > 2^{\mathcal{K}} \cdot (\ell(p)+1)$ and the decrease of counters is bounded by $\delta_{\max}$ at each step, we deduce as above that $c(p+1) > b(p+1)$ for all $b \in B$. Then, the sequence τ is also $(p+1)$-covering, and f_p^* is a winning strategy from c_i in $\mathcal{V}_{p+1}$.

Then, $\text{size}(f_{p+1,c}, p+1, c_i) \leq \text{size}(f_p^*, p+1, c_i) = \text{size}(f_p, p, c_i[p]) \leq \ell(p)$, and $|\sigma_2| \leq \ell(p)$. Hence, $|\sigma| = \text{size}(f_{p+1}, p+1, c) \leq (2^{\mathcal{K}})^{p+2} \cdot (\ell(p)+1)^{p+1} + \ell(p)$.

We can conclude that $\ell(p+1) \leq (2^{\mathcal{K}})^{p+2} \cdot (\ell(p)+1)^{p+1} + \ell(p)$.

This recurrence relation can now be used in order to bound $\ell(n)$. Let g be the function defined by $g(0) = 2^{\mathcal{K}}$ and $g(p+1) = g(p)^{2p+4}$. We show by recurrence that $\ell(p) \leq g(p)$ for all p. The case $p = 0$ is trivial. Now assume the inequality holds for p. By the previous recurrence relation, we have:

$$\begin{aligned} \ell(p+1) &\leq \left(2^{\mathcal{K}}\right)^{p+2} \cdot (\ell(p)+1)^{p+1} + \ell(p) \leq \left(2^{\mathcal{K}}\right)^{p+2} \cdot (g(p)+1)^{p+1} + g(p) \\ &\leq \left(2^{\mathcal{K}}\right)^{p+2} \cdot g(p)^{p+2} \qquad \text{(since } g(p) \geq p+2) \end{aligned}$$

$$\le g(p)^{p+2} \cdot g(p)^{p+2} \le g(p)^{2p+4}$$

Hence: $\ell(p+1) \le g(p+1)$.
On the other hand, one can show that $g(p) = 2^{\mathcal{K}\cdot 2^p \cdot (p+1)!}$. Therefore

$$L = \ell(n) \le g(n) \le 2^{\mathcal{K}\cdot 2^n \cdot (n+1)!} \le 2^{\mathcal{K}\cdot n^{n+1}} \le 2^{\mathcal{K}\cdot \mathcal{K}^{\mathcal{K}}} \le 2^{\mathcal{K}^{\mathcal{K}+1}}.$$

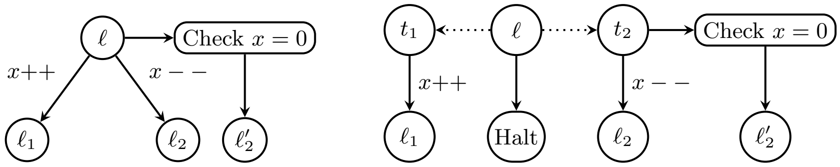
□

```
begin
    $\mathcal{C}$ := the set of configurations with counters bounded by $c_0 + \delta_{\max} \cdot L$;
    $\mathcal{C}_A, \mathcal{C}_E$ := copies of $\mathcal{C}$; $\forall c \in \mathcal{C}$, $c_A$ (resp. $c_E$) is the copy of $c$ in $\mathcal{C}_A$ (resp. $\mathcal{C}_E$);
    mark(c):= false for each $c$ in $\mathcal{C}_A \uplus \mathcal{C}_E$;
    If $c_A \in \mathcal{C}_A$, $succ(c_A)$ := successors of $c$ in $\mathcal{C}_A$ by transitions of $T_u$ and
    $c_E \in \mathcal{C}_E$;
    If $c_E \in \mathcal{C}_E$, $succ(c_E)$ := successors of $c$ in $\mathcal{C}_A$ by transitions of $T_c$;
    forall the configurations $c$ in $\mathcal{C}_A \uplus \mathcal{C}_E$ do
        if $c \succcurlyeq b$ for some $b \in B$ then mark(c):= true;
    while not end do
        end:= true;
        forall the $c \in \mathcal{C}_A$ do
            if all $c' \in succ(c)$ such that mark(c')=true then mark(c):=true;
            end:=false;
        forall the $c \in \mathcal{C}_E$ do
            if there is $c' \in succ(c)$ such that mark(c')=true then
            mark(c):=true; end:=false;
    return mark($c_{0,A}$);
```

Algorithm 1: Deciding reachability of an upward-closed target.

Proof (Theorem 4). Having a bound $L = 2^{\mathcal{K}^{\mathcal{K}+1}}$ on the size of the optimal strategy gives us the decision procedure described by Algorithm 1, which runs in doubly exponential time.

We now prove the lower bound. As in [8], we reduce the following problem: given an alternating counter machine of size N, does it have a halting computation in which the value of each counter is bounded by 2^{2^N}? This problem is AEXPSPACE-hard, hence 2-EXPTIME-hard [5]. Given such an alternating counter machine, we build an AVASSI with an upward-closed target for which there is a winning strategy if and only if there is a 2^{2^N}-bounded halting computation in the counter machine. We know from Lipton [13] that a 2^{2^N}-bounded counter machine of size N can be simulated by a Petri net of size $O(N^2)$. This construction is easily adapted to our case.



(a) Case of an existential state.

(b) Case of a universal state.

Fig. 3. Simulation of a state ℓ of the machine with available transitions t_1: $x{+}{+}$ `goto` ℓ_1 and t_2: `if` $x = 0$ `then` $x - -$ `goto` ℓ_2 `else goto` ℓ_2'. Two cases corresponding to whether ℓ is an existential state or a universal one.

The VASS hence built (in which the set of states contains the set of states of the counter machine) can be turned into an AVASSI in the following way: to each existential state of the counter machine corresponds a state of the AVASSI from which all the outgoing transitions are controllable (and simulate the instructions available from this state in the machine). To each universal state of the counter machine corresponds a state of the AVASSI from which all the outgoing transitions are uncontrollable and lead to intermediate states simulating the instructions. An additional controllable transition to a winning state forces the environment to play. From each intermediate state, there is a single transition, which is controllable, leaving no choice to the controller. This transition simulates the instruction[4]. The target is the set of configurations in an halting state. An example of this simulation in the case of existential and universal states are depicted Fig. 3. □

Observe that an alternate proof for deciding reachability games with upward-closed targets can be performed using the classical construction of *controllable predecessors*. In this case, it can be shown that if a set of configurations is upward-closed, then so is the set of its controllable predecessors. Since the covering order is a well-quasi-ordering, this construction terminates, but this does not provide a complexity upper bound. However, using this alternate construction gives a finite representation of a controller. We do not detail it here as it is standard.

4 Safety Games

In this section, we prove the co-NP-completeness of safety games with semi-linear, finite and upward-closed targets, and we give the construction of the most permissive strategy. We first establish:

[4] The environment cannot decrement vectors: it cannot perform the instruction itself.

Theorem 6. *Safety games on AVASSI with semi-linear targets are in co-NP.*

Proof. To solve a safety game with target S, we consider the AVASSI *restricted to uncontrollable transitions*. Indeed, if only uncontrollable transitions are allowed, and the target cannot be reached, then an obvious winning strategy for the controller is to forbid every controllable transition. Conversely, if the set of configurations S to avoid can be reached by using only uncontrollable transition, there can be no winning strategy for the controller: any run obtained by firing only uncontrollable transitions is an f-run, for any strategy f. Let $\mathit{Target} = \bigcup_{i\in I} \left\{m_i^* + \sum_{u\in U_i} y_u \cdot u \;\middle|\; y_u \in \mathbb{N}\right\}$ be the semi-linear target and let $\mathcal{V}$ be an AVASSI restricted to uncontrollable transitions.

We first introduce some additional notations. Transition t is said *enabled* in configuration $c = (q, m)$ if it is *not inhibited* by m, *i.e.* $m < \mathit{Inh}(t)$. The set of transitions enabled in (q, m) is denoted by $\mathit{En}(q, m)$; we also use the notation $\mathit{En}(m)$ since q is not relevant here. A path $\tau = t_1 \cdots t_k$ in (Q, T) is fireable from configuration $c = (q, m)$ iff for all $j \in [k]$, $t_j \in \mathit{En}(m + \sum_{i=1}^{j-1} \delta(t_i))$. We define the *flow* vector $\mathit{Flow}(t) \in \{-1, 0, 1\}^Q$ ranging over Q as follows:

- for $q \in Q \setminus \{\alpha(t), \beta(t)\}$, $\mathit{Flow}(t)(q) = 0$;
- if $\alpha(t) = \beta(t)$, then $\mathit{Flow}(t)(\alpha(t)) = 0$;
- if $\alpha(t) \neq \beta(t)$, then $\mathit{Flow}(t)(\alpha(t)) = -1$ and $\mathit{Flow}(t)(\beta(t)) = 1$.

If we represent a configuration $c = (q, m)$ by a vector in $\mathbb{N}^{Q\uplus[n]}$, where $c(q') = 1$ if $q' = q$, and $c(q') = 0$ otherwise, we define also the *effect* vector $\mathit{Effect}(t) \in \mathbb{Z}^{Q\uplus[n]}$, which aggregates $\mathit{Flow}(t)$ and $\delta(t)$: for $i \in Q$, $\mathit{Effect}(t)(i) = \mathit{Flow}(t)(i)$ and for $i \in [n]$, $\mathit{Effect}(t)(i) = \delta(t)(i)$. We also denote respectively by $\mathit{Flow}(t_1 \cdots t_k)$ and $\mathit{Effect}(t_1 \cdots t_k)$ the sums $\sum_{j=1}^{k} \mathit{Flow}(t_j)$ and $\sum_{j=1}^{k} \mathit{Effect}(t_j)$, for any sequence of transitions $t_1, \ldots, t_k$.

The decision procedure described by Algorithm 2 proceeds as follows.

- It (non deterministically) builds a linear system $\mathcal{S}$ with two sets of variables: X, the number of occurrences of some transitions in a sequence τ, and Y, the coefficients of a linear set U of *Target*.
- It guesses a small potential solution of this system (in case of non emptiness) as in [14, Chap. 13][5] and returns `true` if it is an actual solution.

[5] If the integer system $AX = B$, with A an (m, n) matrix, has a feasible solution, then it has a feasible solution with coefficients bounded by $n \times (ma)^{2m+4}$, where a is greater than the maximal absolute value of all coefficients of A and B.

```
begin
    Choose $k \leq |T|$; Choose $q \in Q$;
    $\beta(t_0) := q_0$ ($t_0$ is a fictitious transition);
    $\alpha(t_{k+1}) := q$ ($t_{k+1}$ is a fictitious transition);
    $X = \emptyset$; $i := 1$;
    while $i \leq k+1$ do
        if $i \leq k$ then choose $t_i \in T$;
        Choose $(Q_i, T_i)$ a connected subgraph containing $\beta(t_{i-1})$ and $\alpha(t_i)$;
        $X := X \cup \{x_{i,t} \mid t \in T_i\}$;
        if $i \leq k$ then $T'_i := T_i \cup \{t_i\}$ else $T'_i := T_i$;
        $i := i+1$;
    Choose a linear set $U = \left(m^* + \sum_{u \in U} y_u \cdot u\right) \in \mathit{Target}$;
    Define the linear system $\mathcal{S}$;
    $\mathcal{S} := \begin{cases} \forall x \in X, x \geq 1 \wedge \\ \forall 1 \leq i \leq k+1, \mathbf{1}_{\beta(t_{i-1})} + \sum_{t \in T_i} x_{i,t} \mathit{Flow}(t) = \mathbf{1}_{\alpha(t_i)} & (*) \\ \forall 1 \leq i \leq k+1, \forall t \in T'_i, \\ m_0 + \sum_{j \leq i} \sum_{t \in T_j} x_{j,t} \delta(t) + \sum_{j<i} \delta(t_i) < \mathit{Inh}(t) & (**) \\ m_0 + \sum_{i=1}^{k+1} \sum_{t \in T_i} x_{i,t} \cdot \delta(t) + \sum_{i=1}^{k} \delta(t_i) = m^* + \sum_{u \in U} y_u \cdot u & (***) \end{cases}$
    Choose small values for $(x_{i,t})_{i \leq k+1, t \in T}$ and $(y_u)_{u \in U}$;
    return whether $(x_{i,t})_{i \leq k+1, t \in T}, (y_u)_{u \in U}$ is a solution for $\mathcal{S}$
```

Algorithm 2: Guessing a Parikh vector for a firing sequence to an offending configuration.

The sequence τ (which is not built) is of the form $\tau = \tau_1 t_1 \tau_2 \ldots t_k \tau_{k+1}$ with $k \leq |T|$. The algorithm guesses the following items: k, $\{t_i\}_{1 \leq i \leq k}$, connected subgraphs $\{(Q_i, T_i)\}_{1 \leq i \leq k+1}$ of (Q, T) such that T_i is exactly the set of transitions fired in τ_i and finally a linear subset U of *Target*. The set of variables is $X = \{x_{i,t} \mid 1 \leq i \leq k+1 \wedge t \in T_i\}$ and $Y = \{y_u \mid u \in U\}$. The system $\mathcal{S}$ checks if there is a fireable sequence τ whose Parikh vector is $\sum_{i=1}^{k} \mathbf{1}_{t_i} + \sum_{i=1}^{k+1} \sum_{t \in T_i} x_{i,t} \mathbf{1}_t$ and whose final marking belongs to U.

Complexity. The construction of the set of transitions appearing in the solution is done in polynomial time, and the number of variables created is at most $|T|(|T|+1)$. The coefficients of $\mathcal{S}$ are either coefficients of $\delta(t)$ or the integers occurring in U. Hence the size of the system is polynomial. Furthermore, the bound on the small solution provided in [14, Chap. 13] has a polynomial representation in the size of the system. Therefore in our case, this solution can be guessed and checked in polynomial time w.r.t. the input of the safety problem.

Soundness. Assume the algorithm returns true and consider the corresponding solution. For $1 \leq i \leq k+1$, since transitions in T_i form a connected subgraph (when the underlying graph is seen as an undirected one), condition $(*)$ of $\mathcal{S}$ is an Euler condition ensuring that one can derive a path τ_i from $\beta(t_{i-1})$ to $\alpha(t_i)$ in which every transition $t \in T_i$ appears exactly $x_{i,t}$ times. Let us denote m_i the marking reached after the sequence $\tau_1 t_1 \ldots \tau_i$. Condition $(**)$ ensures that transitions of T'_i are enabled in m_i, thus they are also enabled in any previous marking occurring along the sequence (since the marking does not decrease after a transition firing). Thus by recurrence, $\tau_1 t_1 \ldots \tau_{k+1}$ is a firing sequence. At last condition $(***)$ ensures that marking $m_{k+1} \in U \subseteq$ *Target*.

Completeness. Let $c_0 \xrightarrow{t_1} \cdots c_{k-1} \xrightarrow{t_k} \cdots$ be a fireable sequence of transitions from c_0, and let I_{Inh} be the subset of indices of those transition occurrences that actually disable other transitions: $j \in I_{Inh}$ if and only if $En(c_j) \subsetneq En(c_{j-1})$. In the worst case, each transition firing with index in I_{Inh} inhibits exactly one other transition. Then, there cannot be more elements in I_{Inh} than the total number of transitions: $|I_{Inh}| \leq |T|$.

Now, assume there is a reachable configuration $m_f = m^* + \sum_{u \in U} \beta_u \cdot u$ in some linear subset $U \subset$ *Target*. Let $\tau_1 t_1 \cdots \tau_k t_k \tau_{k+1}$ be the sequence of transitions leading to this configuration, where the transitions t_i are exactly the ones inducing a modification in the set of enabled transitions. By the above observation, $k \leq |T|$. Let T_i be the transitions occurring in τ_i. Since the enabled transitions are unchanged during the firing of τ_i, transitions T_i for $i \leq k$ (resp. $i = k+1$) are still enabled before the firing of t_i (resp. in m_f). So denoting by $\sum_{t \in T_i} \alpha_{i,t} \mathbf{1}_t$ the Parikh vector of τ_i, the $\alpha_{i,t}$'s and the β_u's are a solution of the corresponding system $\mathcal{S}$. Using the results of [14, Chap. 13], the algorithm will then find a *small* solution of $\mathcal{S}$.

Summarizing the results, the problem of existence of a winning strategy to ensure a safety objective is in co-NP. □

In general, the set of reachable markings of a Petri Net (and therefore configurations of a VASS) is not semi-linear [10]. However, it was shown to be the case for some restricted models [9,3]. If one determinizes Algorithm 2 and one sets for *Target* all the possible markings, one obtains:

Theorem 7. *Let $\mathcal{V} = (Q, n, T, \alpha, \beta, \delta, Inh)$ be a VASSI s.t. $\delta(T) \subseteq \mathbb{N}^n$. Then its set of reachable configurations is effectively semi-linear.*

It is not always possible to build off-line a most permissive strategy: for instance if we consider the control problem of a Petri Net with inhibitor arcs and a safety objective, where the target consists of a single configuration. In this case, the problem is decidable but the most permissive strategy is not off-line. We show now that in our problem, it is possible and show how to build off-line the most permissive strategy.

Theorem 8. *The most permissive strategy for safety games on AVASSI with semi-linear targets can be represented by a finite-state machine.*

Proof. We define the set of configurations from which the system cannot avoid the target, and show that this set is semi-linear. These configurations are exactly the ones the strategy should avoid. For that we define a partition $\mathcal{P}$ of the set of configurations and, for each $P \in \mathcal{P}$ we define the set $\mathit{Access}_P = \{(c, c') \mid c \in P \text{ and } c \xrightarrow{\tau} c' \text{ with } \tau \in T_e^*\}$. Then $\mathit{Forbid} = \Pi_1((\bigcup_{P \in \mathcal{P}} \mathit{Access}_P) \cap (\mathcal{C} \times \mathit{Target}))$, where $\Pi_1(c, c') = c$. Since in our case, values of a counter only increase during a computation, when a counter has a value that inhibits all the transitions it can inhibit, its precise value along the run does not matter anymore. Following this idea, we define the following partition of the configurations. Let $\mathcal{C}' \subset \mathcal{C}$ be the set of configurations c such that $c(q) = 1$ for some $q \in Q$ and, for all $j \in [n]$, $c(j) \le \max_{t \in T}(\mathit{Inh}(t)_j \mid \mathit{Inh}(t)_j < \infty)$, with the convention that $\max \emptyset = 0$. The set $\mathcal{C}'$ is finite. Moreover, any configuration $c \in \mathcal{C}'$ implicitly defines a set $J \subseteq [n]$ such that $c(j) = \max_{t \in T}(\mathit{Inh}(t)_j \mid \mathit{Inh}(t)_j < \infty)$ if and only if $j \in J$. Then we define, for all $c \in \mathcal{C}'$, the set $P_c = \{c' \in \mathcal{C} \mid c' \ge c \text{ and, for all } j \notin J, c'(j) = c(j)\}$. It is clear that P_c is a linear set, and that $\{P_c \mid c \in \mathcal{C}'\}$ forms a partition of $\mathcal{C}$.

We show now that, for all $c \in \mathcal{C}'$, Access_{P_c} is semi-linear. Fix $c \in \mathcal{C}'$, with $J \subseteq [n]$ such that $c(j) = \max_{t \in T}(\mathit{Inh}(t)_j \mid \mathit{Inh}(t)_j < \infty)$ if and only if $j \in J$, and let $\mathit{UCReach}(c)$ be the set of configurations reachable from c using only uncontrollable transitions. From Theorem 7, $\mathit{UCReach}(c)$ is semi-linear. Let now $(v_k)_{k \in J}$ be the set of vectors of $\mathbb{N}^{Q \cup [n]}$ such that $P_c = \{c + \sum_{k \in J} \lambda_k v_k \mid \lambda_k \in \mathbb{N}, \text{ for all } k\}$. We then define $\mathit{Acc}_{P_c} = \{(c + \sum_{k \in J} \lambda_k v_k, c' + \sum_{k \in J} \lambda_k v_k) \mid \lambda_k \in \mathbb{N} \text{ for all } k \in J, c' \in \mathit{UCReach}(c)\}$. Since $\mathit{UCReach}(c)$ is semi-linear, so is the set $\{c\} \times \mathit{UCReach}(c)$, and Acc_{P_c} is semi-linear. In fact, $\mathit{Access}_{P_c} = \mathit{Acc}_{P_c}$.

Indeed, let $(c_1, c_2) \in \mathit{Access}_{P_c}$ be a pair of configurations such that $c_1 \in P_c$ and $c_1 \xrightarrow{\tau} c_2$, for some $\tau \in T_e^*$. Since $c_1 \in P_c$ then $c_1 = c + \sum_{k \in J} \lambda_k v_k$ for some $(\lambda_k)_{k \in J}$. Moreover, for all $t \in T_e$, $\delta(t) \in \mathbb{N}$. Then, τ being fireable from c_1, it is fireable from c and $c_2 = c + \sum_{k \in J} \lambda_k v_k + \mathit{Effect}(\tau) = c' + \sum_{k \in J} \lambda_k v_k$ where $c' = c + \mathit{Effect}(\tau)$ is in $\mathit{UCReach}(c)$ by construction. Hence $(c_1, c_2) \in \mathit{Acc}_{P_c}$.

Conversely, let (c, c') with $c' \in \mathit{UCReach}(c)$; there is then a sequence of uncontrollable transitions $\tau = t_1 \cdots t_\ell$ such that $c \xrightarrow{\tau} c'$. Let $(\lambda_k)_{k \in J} \in \mathbb{N}^| J|$; we show that $(c + \sum_{k \in J} \lambda_k v_k, c' + \sum_{k \in J} \lambda_k v_k) \in \mathit{Access}_{P_c}$. Let $c_1 = c + \sum_{k \in J} \lambda_k v_k$. By definition of c, t_1 is fireable from c_1. Indeed, for all $j \in [n]$, $c_1(j) > c(j)$ implies that $c(j) = \max_{t \in T}(\mathit{Inh}(t)_j \mid \mathit{Inh}(t)_j < \infty)$.

Then for any transition t, either it cannot be inhibited on dimension j (*i.e.* $\mathit{Inh}(t)_j = \infty$), or it was already inhibited in c (*i.e.* $c(j) \geq \mathit{Inh}(t)_j$). We then have $c_1 \xrightarrow{t_1} c_1 + \mathit{Effect}(t_1) = c + \sum_{k \in J} \lambda_k v_k + \mathit{Effect}(t_1)$. Let $1 \leq i < \ell$, and assume by induction hypothesis that $c_1 \xrightarrow{t_1 \cdots t_i} c + \mathit{Effect}(t_1 \cdots t_i) + \sum_{k \in J} \lambda_k v_k$. Since t_{i+1} is fireable from $c + \mathit{Effect}(t_1 \cdots t_i)$, we know that for all $j \in [n]$, $\big(c + \mathit{Effect}(t_1 \cdots t_i)\big)(j) < \mathit{Inh}(t_{i+1})_j$. Moreover, if $\big(c + \mathit{Effect}(t_1 \cdots t_i) + \sum_{k \in J} \lambda_k v_k\big)(j) > (c + \mathit{Effect}(t_1 \cdots t_i))(j)$, then either $c(j) \geq \mathit{Inh}(t_{i+1})_j$ or $\mathit{Inh}(t_{i+1})_j = \infty$. The first case is impossible, then t_{i+1} is fireable from $c + \mathit{Effect}(t_1 \cdots t_i) + \sum_{k \in J} \lambda_k v_k$ and $c_1 \xrightarrow{t_1 \cdots t_{i+1}} c + \mathit{Effect}(t_1 \cdots t_{i+1}) + \sum_{k \in J} \lambda_k v_k$. We have then shown by induction that $c + \sum_{k \in J} \lambda_k v_k \xrightarrow{\tau} c' + \sum_{k \in J} \lambda_k v_k$, and $(c + \sum_{k \in J} \lambda_k v_k, c' + \sum_{k \in J} \lambda_k v_k) \in \mathit{Access}_{P_c}$.

Hence, for all $c \in \mathcal{C}'$, Access_{P_c} is semi-linear. It follows that *Forbid* is also semi-linear.

One can then compute, for a given controllable transition t, the set of configurations from which this transition is allowed. Let $\mathit{Pre}_{\mathit{Forbid}}(t) = \{c \in \mathcal{C} \mid \exists c' \in \mathit{Forbid}, c = c' - \mathit{Effect}(t)\}$. Since *Forbid* is semi-linear and the image of a semi-linear by an affine application is still semi-linear, we get that $\mathit{Pre}_{\mathit{Forbid}}(t)$ is semi-linear, for any controllable transition t. Then, the set of configurations from which t is allowed is given by $\mathcal{C} \setminus \mathit{Pre}_{\mathit{Forbid}}(t)$, which is still semi-linear. □

By a reduction from 3-SAT, we also obtain the following result.

Theorem 9. *Safety games on AVASSI with finite targets or upward-closed targets are co-NP-hard even with* $|Q| = 1$.

Proof. An instance of 3-SAT is transformed into an AVASSI (where only the environment plays) that is won by the controller if and only if it cannot reach a given marking. This yields co-NP-hardness for safety games with finite or upward-closed targets (hence also for semi-linear ones).

Let X be a set of k variables $\{x_1, \ldots, x_k\}$. The literals over X are the elements of $\mathit{Lit}(X) = X \uplus \{\overline{x} \mid x \in X\}$, with the usual convention: $\overline{\overline{x}} = x$. Let $\varphi = C_1 \wedge C_2 \cdots \wedge C_n$ be an instance of 3-SAT: each C_i is of the form $\ell_{i,1} \vee \ell_{i,2} \vee \ell_{i,3}$ where $\ell_{i,j} \in \mathit{Lit}(X)$.

We construct an AVASSI $\mathcal{V}_\varphi$, where only the environment plays, as explained below. A counter is associated with each literal, and also with each clause (they are named accordingly, with a slight abuse of notation). An additional counter *cont* is used, its role is explained below. Hence a vector of the AVASSI is of the form: $(x_1, \overline{x_1}, \ldots, x_k, \overline{x_k}, C_1, \ldots, C_n, \mathit{cont})$.

There are three states in the AVASSI: q_{set} (the initial state), q_{eval}, q_{fill}, each corresponding to a phase of the execution of the model.

The figures will represent Petri Nets with bounded places (pictured by double circles) for clarity. The global structure of the AVASSI is represented on Figure 4 by a Petri Net (whose successive bounded places represent respectively q_{set}, q_{eval} and q_{fill}) that can be easily turned into a state machine and hence into an AVASSI.

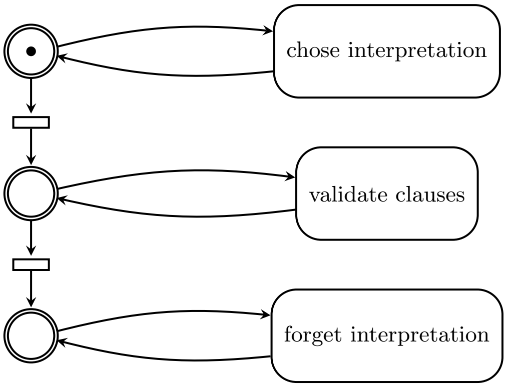


Fig. 4. A high-level view of the Petri Net to solve 3-SAT

On state q_{set} are self loops that allow the environment to add 1 to at most one literal for each variable. The counter *cont* keeps track of how many variables where set that way. Thus for $j \in [k]$ there are two transitions t_j and $t_{\overline{j}}$ such that:

$$\alpha(t_j) = \alpha(t_{\overline{j}}) = \beta(t_j) = \beta(t_{\overline{j}}) = q_{set}$$

$$Inh(t_j) = Inh(t_{\overline{j}}) = (\infty, \dots, \infty, \underset{\underset{x_j}{\uparrow}}{1}, \underset{\underset{\overline{x_j}}{\uparrow}}{1}, \infty, \dots, \infty)$$

$$\delta(t_j) = (0, \dots, 0, \underset{\underset{x_j}{\uparrow}}{1}, \underset{\underset{\overline{x_j}}{\uparrow}}{0}, 0, \dots, \underset{\underset{cont}{\uparrow}}{1})$$

$$\delta(t_{\overline{j}}) = (0, \dots, 0, \underset{\underset{x_j}{\uparrow}}{0}, \underset{\underset{\overline{x_j}}{\uparrow}}{1}, 0, \dots, \underset{\underset{cont}{\uparrow}}{1})$$

Remark that each for each j, only t_j or $t_{\overline{j}}$ can be fired, and only once. Also that when $cont = k$, for all variables x, either x or $\overline{x}$ is marked. In order to go to the next phase, there is a transition t_{set} from q_{set} to q_{eval} without inhibition nor displacement: $Inh(t_{set}) = (\infty, \ldots, \infty)$ and $\delta(t_{set}) = (0, \ldots, 0)$.

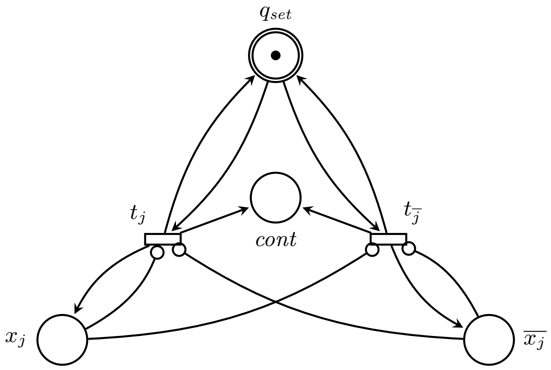


Fig. 5. Detailed view of the phase "chose interpretation" for a single variable, seen as a Petri net. This widget is active only in state q_{set}.

On state q_{eval}, the counters of clauses that are satisfied w.r.t. the values of the variables that were incremented. Namely, the counter of a clause can be incremented if *the opposite* of one of its literal has value 0: for example, if $C = x_1 \vee \overline{x_2} \vee x_3$, then C can be incremented when $\overline{x_1}$, x_2 or $\overline{x_3}$ is empty. Note that requiring $\overline{x}$ to be empty is equivalent to requiring x to be 1 if and only if $cont = k$.

Thus for $i \in [n]$ and $j \in \{1, 2, 3\}$, there is a transition $\tau_{i,j}$ such that

$$\alpha(\tau_{i,j}) = \beta(\tau_{i,j}) = q_{eval}$$

$$Inh(\tau_{i,j}) = (\infty, \ldots, \infty,\ \underset{\underset{\overline{\ell_{i,j}}}{\uparrow}}{1},\ \infty, \ldots, \infty)$$

$$\delta(\tau_{i,j}) = (\infty, \ldots, \infty,\ \underset{\underset{C_i}{\uparrow}}{1},\ \infty, \ldots, \infty)$$

Remark that nothing prevents transition $\tau_{i,j}$ to be fired multiple times, nor several transitions $\tau_{i,j'}$ that increment the same counter C_i to be fired, although it would have been possible through inhibition constraints on counter C_i. This is actually irrelevant to the coding of 3-SAT performed here.

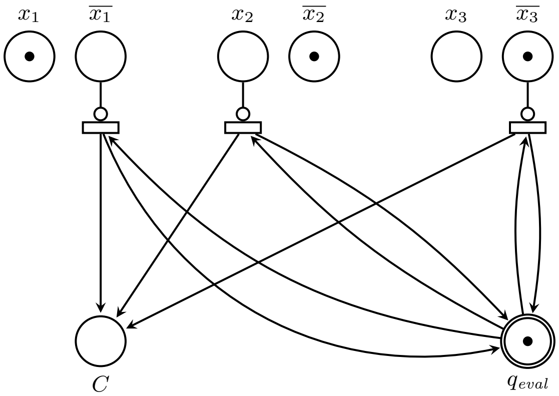


Fig. 6. Detailed view of the phase "validate clauses" for a single clause $C = x_1 \vee \overline{x_2} \vee x_3$, seen as a Petri net. This widget is active only in state q_{eval}. The current marking, which satisfies clause C, allows place C to be marked.

As before, in order to go to the next phase, there is a also a transition τ_{eval} from q_{eval} to q_{fill} without inhibition nor displacement.

On state q_{fill}, the counters of literals can be incremented. Thus for $j \in [k]$ there are two transitions t'_j and $t'_{\bar{j}}$ such that:

$$\alpha(t'_j) = \alpha(t'_{\bar{j}}) = \beta(t'_j) = \beta(t'_{\bar{j}}) = q_{\mathit{fill}} \qquad \mathit{Inh}(t'_j) = \mathit{Inh}(t'_{\bar{j}}) = (\infty, \ldots, \infty)$$

$$\delta(t'_j) = (0, \ldots, 0, \underset{\substack{\uparrow \\ x_j}}{1}, 0, \ldots, 0) \qquad \delta(t'_{\bar{j}}) = (0, \ldots, 0, \underset{\substack{\uparrow \\ \overline{x_j}}}{1}, 0, \ldots, 0)$$

Note that counter *cont* is not incremented in this case. So its value remains unchanged in this phase of the execution.

Having set the structure of the game, we can now define its target. First, for finite objectives, the goal of the environment is to reach the configuration $(q_{\mathit{fill}}, (1, \ldots, 1, k))$. In the case of upward-closed objectives, the goal of the environment is to cover this very same configuration.

Assume that the instance φ of 3-SAT has a solution $S \subseteq X$ given by the set of variables set to *true*. Let $\hat{S} = S \uplus \{\overline{x} \mid x \notin S\}$ be the set of literals that evaluate to *true*. Then the environment has a strategy to reach the configuration $(q_{\mathit{fill}}, (1, \ldots, 1, k))$, as follows. In state q_{set}, for $j \in [k]$, the environment fires the transition t_j if $x_j \in S$ and $t_{\overline{j}}$ otherwise. After that, remark that counter *cont* has value k, for $\ell \in \hat{S}$, counter ℓ has value 1 and all other counters have value 0.

Then, the environment moves to q_{eval}. Since φ has a solution, then for all $i \in [n]$, there is a literal $\ell_{i,j} \in \hat{S}$. So the counter $\overline{\ell_{i,j}}$ has value 0, and transition $\tau_{i,j}$ can be fired. The environment must chose to fire only one $\tau_{i,j}$ (if several are fireable) in the case of finite targets; for upward-closed targets, it may chose to fire several of them (or the same several times). After this phase, all counters C_i have value (at least) 1, while other counters have not changed.

Finally, the environment goes to q_{fill}. It then has to fire transitions to set all literal counters to 1: for $j \in [k]$, the environment fires the transition t'_j if $x_j \notin S$ and $t'_{\overline{j}}$ otherwise. At this point, the system has reached configuration $(q_{\mathit{fill}}, (1, \ldots, 1, k))$, and the environment wins, so no controller can avoid this target. In the case of upward-closed targets, more transitions can be fired afterwards, since the goal is already achieved.

Now suppose that the environment reaches its goal, *i.e.* reaches or covers $(q_{\mathit{fill}}, (1, \ldots, 1, k))$. Then in particular, there were at least k firings of self-loops over state q_{set}, since only these increment counter *cont*. Note that due to how the transitions are inhibited, there are at most k such firings. So exactly one counter per variable (out of two) was incremented. Let v be the value of the counters upon leaving q_{set}, and S be the set of literals that were incremented in the pair relative to a given variable: $S = \{x \in \mathit{Lit}(X) \mid v(x) = 1\}$. Let $\hat{S} = S \cap X$ be the set of positive literals in this set. We claim that setting variables of $\hat{S}$ to *true* and other variables to *false* yields a valuation that satisfies φ. Clearly, with valuation $\hat{S}$, all literals of S have value *true*.

For every clause C_i, a transition $\tau_{i,j}$ was fired in state q_{eval}. Let $U_i = \{\ell_{i,j} \mid \tau_{i,j} \text{ was fired}\}$ be the set of literals that were "used" in q_{eval}. For every $\ell \in U_i$, the counter $\overline{\ell}$ was empty after leaving q_{set} (otherwise the corresponding transition would have been inhibited), so $\ell \in S$. Hence clause C_i is satisfied by valuation $\hat{S}$. As a result φ is satisfiable.

Finally remark that the construction can be modified to use a single state instead of three states, by adding counters with outgoing inhibitor arcs to ensure the succession of the three phases. □

Corollary 10. *Safety games on AVASSI with finite, upward-closed or semi-linear targets are co-NP-complete.*

5 Conclusion

We solve reachability and safety games with concurrent semantics for an extension of VASS with inhibition conditions, for finite, upward-closed and semi-linear targets. When the reachability games are decidable, the procedures are elementary. For safety games, which are co-NP-complete, the procedure allows to construct the most permissive strategy. Future work includes studying more complex winning objectives, *e.g.*, parity games. Another direction could concern games on continuous models, like timed extensions of Petri nets.

References

1. Abdulla, P.A., Bouajjani, A., d'Orso, J.: Monotonic and downward closed games. J. Log. Comput. **18**(1) (2008) 153–169.
2. Bollue, K., Slaats, M., Abraham, E., Thomas, W., Abel, D.: Synthesis of Behavioral Controllers for DES: Increasing Efficiency. In: WODES'10, IFAC (2010).
3. Bouziane, Z., Finkel, A.: Cyclic petri net reachability sets are semi-linear effectively constructible. Electr. Notes Theor. Comput. Sci. **9** (1997) 15–24.
4. Brázdil, T., Jančar, P., Kucera, A.: Reachability games on extended vector addition systems with states. In Abramsky, S., Gavoille, C., Kirchner, C., Meyer auf der Heide, F., Spirakis, P.G., eds.: ICALP'10, part II. Volume 6199 of Lecture Notes in Computer Science, Springer (2010) 478–489.
5. Chandra, A.K., Kozen, D., Stockmeyer, L.J.: Alternation. J. ACM **28**(1) (1981) 114–133.
6. Chatterjee, K., Doyen, L., Henzinger, T.A., Raskin, J.F.: Generalized mean-payoff and energy games. In Lodaya, K., Mahajan, M., eds.: FSTTCS'10. Volume 8 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2010) 505–516.
7. De Alfaro, L., Faella, M., Henzinger, T.A., Majumdar, R., Stoelinga, M.: The Element of Surprise in Timed Games. In Amadio, R., Lugiez, D., eds.: CONCUR'03. Volume 2761 of Lecture Notes in Computer Science, Springer (2003) 142–156.
8. Demri, S., Jurdziński, M., Lachish, O., Lazić, R.: The covering and boundedness problems for branching vector addition systems. In: Proc. of the 29th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'09). Volume 4 of Leibniz International Proceedings in Informatics, Leibniz-Zentrum für Informatik (December 2009) 181–192.
9. Esparza, J.: Petri nets, commutative context-free grammars, and basic parallel processes. Fundam. Inform. **31**(1) (1997) 13–25.
10. Hopcroft, J.E., Pansiot, J.J.: On the reachability problem for 5-dimensional vector addition systems. Theor. Comput. Sci. **8** (1979) 135–159.
11. Kumar, R., Garg, V.: On computation of state avoidance control for infinite state systems in assignment program framework. IEEE Trans. Autom. Sci. Eng. **2**(1) (2005) 87–91.
12. Le Gall, T., Jeannet, B., Marchand, H.: Supervisory Control of Infinite Symbolic Systems using Abstract Interpretation. In: CDC'05, IEEE Press (2005) 31–35.
13. Lipton, R.: The reachability problem requires exponential space. Technical Report 62, Dept. of Computer Science, Yale University (1976).

14. Papadimitriou, C.H., Steiglitz, K.: Combinatorial Optimization: algorithms and complexity. Prentice-Hall (1982).
15. Rackoff, C.: The covering and boundedness problems for vector addition systems. Theor. Comput. Sci. **6** (1978) 223–231.
16. Raskin, J.F., Samuelides, M., Van Begin, L.: Petri games are monotonic but difficult to decide. Technical Report 508, Université Libre de Bruxelles (2003).
17. Raskin, J.F., Samuelides, M., Van Begin, L.: Games for counting abstractions. Electr. Notes Theor. Comput. Sci. **128**(6) (2005) 69–85.
18. Serre, O.: Parity games played on transition graphs of one-counter processes. In Aceto, L., Ingólfsdóttir, A., eds.: FOSSACS'06. Volume 3921 of LNCS, Springer (2006) 337–351.
19. Sreenivas, R.S.: Some observations on supervisory policies that enforce liveness in partially controlled free-choice petri nets. Math. Comp. Simul. **70**(5-6) (2006) 266–274.