Alternating Vector Addition Systems with States*

Jean-Baptiste Courtois and Sylvain Schmitz

LSV, ENS Cachan & CNRS & INRIA, France

Abstract. Alternating vector addition systems are obtained by equipping vector addition systems with states (VASS) with 'fork' rules, and provide a natural setting for infinite-arena games played over a VASS. Initially introduced in the study of propositional linear logic, they have more recently gathered attention in the guise of *multi-dimensional energy* games for quantitative verification and synthesis.

We show that establishing who is the winner in such a game with a state reachability objective is 2-EXPTIME-complete. As a further application, we show that the same complexity result applies to the problem of whether a VASS is simulated by a finite-state system.

1 Introduction

Vector addition systems with states (VASS) allow to model systems manipulating multiple discrete resources, for instance bank accounts balances or numbers of processes running concurrently. Extending their definition to two-players games is both a very natural endeavour and a tricky problem: the most immediate definition, where both players can freely update the vector values, leads to an undecidable game even with the simplest winning condition, namely (control) state reachability [2].

Facing this difficulty, one might expect to see a flurry of competing definitions for VASS games that would retain decidability through various restrictions. Surprisingly, this is not really the case: if there is indeed a large number of denominations (e.g. B-VASS games [16], Z-reachability games [5], multi-dimensional energy games [7]), Abdulla, Mayr, Sangnier, and Sproston [3] noted last year that they all defined essentially the same *asymmetric* class of games, where one player is restricted and cannot update the vector values.

Our contention in this paper is that so many different people coming up independently with the same model is not a coincidence, but a sure sign of a fundamental idea deserving investigation in its own right. We find further arguments in our own initial interest in such games, which comes from the study of simulation problems between Petri nets and finite-state systems [9, 12] where they arise naturally—Abdulla et al. [1] recently made a similar observation. Furthermore the model was already explicitly defined in the '90s in the study of substructural logics [13, 10], and appears implicitly in recent proofs of complexity lower bounds in [8, 4]. We show in this paper that determining the winner of

* Work funded in part by the ANR grant 11-BS02-001-01 REACHARD.

E. Csuhaj-Varjú et al. (Eds.): MFCS 2014, Part I, LNCS 8634, pp. 220–231, 2014.

an asymmetric VASS game with a state reachability objective is 2-ExpTime-complete. We extend for this well-known techniques by Rackoff [15] and Lipton [14] used to establish the complexity of VASS problems, see sections 3 and 4. We also provide refined bounds when the dimension of the problem is fixed, and show how to compute the *Pareto frontier* for such games.

Perhaps more importantly than those technical contributions, we single out in Sec. 2 a simple definition for alternation in VASS by way of 'fork' rules (following [13]), for which the complexity analyses of sections 3 and 4 are relatively easy, and establish it as a pivotal definition for VASS games. Indeed, we relate it to energy games in Sec. 5 (following [3]) and to regular simulation problems for VASS in Sec. 6. Our lower bound improves on all the published bounds for those problems, including the ExpSpace-hardness of simulations between *basic parallel processes* and finite-state processes due to Lasota [12]. Our upper bound applies to the simulation of Petri nets by finite-state systems, for which only decidability was known [9].

Due to page limits, some material is omitted, but can be found in the full version of the paper at http://hal.inria.fr/hal-00980878.

2 Alternating VASS

VASS were originally called 'and-branching' counter machines by Lincoln, Mitchell, Scedrov, and Shankar [13], and were introduced to prove the undecidability of propositional linear logic. Kanovich [10] later identified a fragment of linear logic, called the $(!, \oplus)$-Horn fragment, that captures exactly alternation in VASS, and adopted a game viewpoint. As discussed in sections 5 and 6, this class of systems has since reappeared in other contexts, which motivates its study in earnest.

2.1 Basic Definitions

An *alternating vector addition system with states* (AVASS) is syntactically a tuple $\mathcal{A} = \langle Q, d, T_u, T_f \rangle$ where Q is a finite set of *states*, d is a *dimension* in $\mathbb{N}$, and $T_u \subseteq Q \times \mathbb{Z}^d \times Q$ and $T_f \subseteq Q^3$ are respectively finite sets of *unary* and *fork* rules. We denote unary rules $(q, \mathbf{u}, q_1)$ in T_u with $\mathbf{u}$ in $\mathbb{Z}^d$ by '$q \xrightarrow{\mathbf{u}} q_1$' and fork rules (q, q_1, q_2) in T_f by '$q \to q_1 \wedge q_2$.' A *vector addition system with states* (VASS) is an AVASS with $T_f = \emptyset$.

Deduction Semantics. Given an AVASS, its semantics is defined by a deduction system over *configurations* $(q, \mathbf{v})$ in $Q \times \mathbb{N}^d$. For rules $q \xrightarrow{\mathbf{u}} q_1$ and $q \to q_1 \wedge q_2$,

$$\frac{q, \mathbf{v}}{q_1, \mathbf{v} + \mathbf{u}} \text{ (unary)} \qquad \frac{q, \mathbf{v}}{q_1, \mathbf{v} \quad q_2, \mathbf{v}} \text{ (fork)}$$

where '+' denotes component-wise addition in $\mathbb{N}^d$, and implicitly $\mathbf{v} + \mathbf{u}$ has no negative component, i.e. is in $\mathbb{N}^d$. When working with finite deduction trees t, we define the *height* $h(t)$ of t as the maximal length among all its branches. A *(multi)-context* C is a finite tree with n distinguished leaves labelled with

n distinct variables $x_1, \ldots, x_n$; $C[t_1, \ldots, t_n]$ then denotes the tree obtained by substituting for each $1 \leq j \leq n$ the tree t_j for the variable x_j.

Game Semantics. The top-down direction of the deduction semantics allows for potentially infinite deduction trees, and defines in a natural way an *asymmetric* VASS game as defined by Kanovich [10] and later by Raskin et al. [16]. Two players, 'Controller' and 'Environment', play over the infinite arena $Q \times \mathbb{N}^d$. In a current configuration $(q, \mathbf{v})$, Controller chooses among the applicable rules in $T_u \cup T_f$. In case of a unary rule $q \xrightarrow{\mathbf{u}} q'$, the next configuration is $(q', \mathbf{v} + \mathbf{u})$, where by assumption $\mathbf{v} + \mathbf{u} \geq \mathbf{0}$ where '$\mathbf{0}$' denotes the null vector in $\mathbb{N}^d$. In case of a fork rule $q \to q_1 \wedge q_2$, Environment then chooses which branch of the deduction tree to explore, i.e. chooses between $(q_1, \mathbf{v})$ and $(q_2, \mathbf{v})$ as the next configuration. Various winning conditions on such plays $(q_0, \mathbf{v}_0), (q_1, \mathbf{v}_1), \ldots$ can then be envisioned, and correspond to conditions that must be satisfied by all the branches of a deduction tree. As shown by Abdulla et al. [3], such asymmetric games are closely related to *multi-dimensional energy games* [7, 5], see Sec. 5.

2.2 Decision Problems and Complexity

We assume when deriving complexity bounds a binary encoding of vectors in $\mathbb{Z}^d$. That is, letting $\|\mathbf{u}\|_\infty \stackrel{\text{def}}{=} \max_{1 \leq i \leq d} |\mathbf{u}(i)|$ denote the norm of the vector $\mathbf{u}$ and defining $\|T_u\|_\infty \stackrel{\text{def}}{=} \max_{(q,\mathbf{u},q') \in T_u} \|\mathbf{u}\|_\infty$, then the size of an AVASS $\langle Q, d, T_u, T_f \rangle$ depends polynomially on the *bitsize* $\log(\|T_u\|_\infty + 1)$. Note that we can reduce by standard techniques all our decision problems to work with a set of unary rules T'_u with effects $\mathbf{u} = \mathbf{e}_i$ or $\mathbf{u} = -\mathbf{e}_i$—where '$\mathbf{e}_i$' is the unit vector with '1' in coordinate i and '0' everywhere else—, but this comes at the expense of an increase in the dimension by a factor of $\log(\|T_u\|_\infty + 1)$.

Reachability. The decision problem that originally motivated the definition of AVASS by Lincoln et al. [13] is *reachability*: given an AVASS $\langle Q, d, T_u, T_f \rangle$ and two states q_r and q_ℓ in Q, does there exist a deduction tree with root labelled by $(q_r, \mathbf{0})$ and every leaf labelled by $(q_\ell, \mathbf{0})$? Equivalently, does Controller have a strategy that ensures that a play starting in $(q_r, \mathbf{0})$ eventually visits $(q_\ell, \mathbf{0})$?

Fact 2.1 (Lincoln et al. **[13]).** *Reachability in AVASS is undecidable.*

State Reachability. Our main problem of interest in this paper is *(control) state reachability* (aka *leaf coverability*): given as before an AVASS $\langle Q, d, T_u, T_f \rangle$ and two states q_r and q_ℓ in Q, we ask now whether there exists a deduction tree with root labelled by $(q_r, \mathbf{0})$ and every leaf label in $\{q_\ell\} \times \mathbb{N}^d$. Equivalently, does Controller have a strategy that ensures that a play starting in $(q_r, \mathbf{0})$ eventually visits $(q_\ell, \mathbf{v})$ for some $\mathbf{v}$ in $\mathbb{N}^d$? We prove in this paper that state reachability is 2-ExpTime-complete, see Thm. 3.1 and Thm. 4.1.

Non-termination. A second problem of interest is *non-termination*: given an AVASS $\langle Q, d, T_u, T_f \rangle$ and an initial state q_r in Q, does there exist a deduction tree where every branch is infinite? Equivalently, does Controller have a strategy to ensure that a play starting in $(q_r, \mathbf{0})$ never stops?

Brázdil, Jančar, and Kučera [5] show in the context of Z-reachability games that this problem is ExpSpace-hard, and in $(d-1)$-ExpTime when the dimension d is fixed. Our 2-ExpTime lower bound in Thm. 4.1 is the best known lower bound for this problem, leaving a large complexity gap.

We discuss a few other decision problems related to energy games in Sec. 5 and to regular VASS simulations in Sec. 6.

3 Complexity Upper Bounds

The state reachability problem asks about the existence of a deduction tree with root $(q_r, \mathbf{0})$ and leaves labels in $\{q_\ell\} \times \mathbb{N}^d$, which describes when using the game semantics a winning strategy for Controller. More generally, we are interested in deduction trees with root label $(q, \mathbf{v})$ and leaves in $\{q_\ell\} \times \mathbb{N}^d$, which we call *witnesses* for $(q, \mathbf{v})$. Let us write $\mathcal{A}, q_\ell \triangleright q, \mathbf{v}$ if such a witness exists in an AVASS $\mathcal{A}$; then the state reachability problem asks whether $\mathcal{A}, q_\ell \triangleright q_r, \mathbf{0}$.

Following Rackoff [15], the main idea to prove a 2-ExpTime upper bound on the state reachability problem is to prove a doubly exponential upper bound on the height of witnesses, by induction on the dimension d; see Sec. 3.1. But let us first make a useful observation: if $\mathcal{A}, q_\ell \triangleright q, \mathbf{v}$ and $(q', \mathbf{v}') \geq (q, \mathbf{v})$ for the product ordering over $Q \times \mathbb{N}^d$, i.e. if $q = q'$ and $\mathbf{v}'(i) \geq \mathbf{v}(i)$ for all $1 \leq i \leq d$, then $\mathcal{A}, q_\ell \triangleright q', \mathbf{v}'$. This means that the set of root labels that ensure reaching q_ℓ is *upward-closed*, and since $(Q \times \mathbb{N}^d, \leq)$ is a well partial order, it has a finite set of minimal elements called its *Pareto frontier*:

$$\text{Pareto}(\mathcal{A}, q_\ell) \stackrel{\text{def}}{=} \min\{(q, \mathbf{v}) \in Q \times \mathbb{N}^d \mid \mathcal{A}, q_\ell \triangleright q, \mathbf{v}\} \ . \tag{1}$$

We use in Sec. 3.2 the bounds on the size of witnesses to show that Pareto frontiers can be computed in doubly exponential time, which in turn proves:

Theorem 3.1. *State reachability in AVASS is in* 2-ExpTime. *It is in* ExpTime *when the dimension is fixed, and in* PTime *when furthermore the bitsize is fixed.*

Note that the PTime bound in the case of a fixed dimension and fixed bitsize, is not trivial, since it still allows for infinite arenas. In essence it shows one can add a fixed number of counters to a reachability game 'for free.'

3.1 Small Witnesses

Let us fix an instance $\langle \mathcal{A}, q_r, q_\ell \rangle$ of the state reachability problem with $\mathcal{A} = \langle Q, d, T_u, T_f \rangle$ and write $[d] \stackrel{\text{def}}{=} \{1, \ldots, d\}$ for its set of components. For a subset $I \subseteq [d]$ of the components of $\mathcal{A}$, we write $\mathbf{u}_{\restriction I}$ for the projection of a vector $\mathbf{u}$ on I, and define the *projection* $\mathcal{A}_{\restriction I} \stackrel{\text{def}}{=} \langle Q, |I|, T_{u\restriction I}, T_f \rangle$ of $\mathcal{A}$ on I as the AVASS with unary rules $T_{u\restriction I} \stackrel{\text{def}}{=} \{(q, \mathbf{u}_{\restriction I}, q') \mid (q, \mathbf{u}, q') \in T_u\}$. Let $W_I \stackrel{\text{def}}{=} \{(q, \mathbf{v}) \in Q \times \mathbb{N}^{|I|} \mid \mathcal{A}_{\restriction I}, q_\ell \triangleright q, \mathbf{v}\}$ be the set of witness roots in $\mathcal{A}_{\restriction I}$. We are interested in bounding the height $h(t)$ of *minimal* witnesses t in $\mathcal{A}_{\restriction I}$:

$$H_I \stackrel{\text{def}}{=} \sup_{(q,\mathbf{v}) \in W_I} \min\{h(t) \mid t \text{ witnesses } (q, \mathbf{v})\} \ , \tag{2}$$

where implicitly $H_I = 0$ if no witness exists.

A last remark before we proceed is that, if a label $(q, \mathbf{v})$ appears twice along a branch of a witness t, i.e. if $t = C[C'[t']]$ for some context C, some non-empty context C' with root label $(q, \mathbf{v})$, and tree t' with root label $(q, \mathbf{v})$, then the *shortening* $C[t']$ of t, obtained by replacing $C'[t']$ by t' in t, is also a witness.

Assume that there exists a witness for some root label $(q, \mathbf{v})$. We bound H_I by induction on $|I|$: for the base case where $I = \emptyset$, by repeated shortenings we see that no branch of a minimal witness can have the same state twice, thus

$$H_\emptyset \leq |Q| . \tag{3}$$

Consider now some non-empty set I and a minimal witness t for $(q, \mathbf{v})$. We would like to bound H_I, assuming by induction hypothesis that we are able to bound H_J for all $J \subsetneq I$ by some value $H_{\subsetneq I} = \max_{J \subsetneq I} H_J$. Define for this a large value $B_I \stackrel{\text{def}}{=} \|T_u\|_\infty \cdot H_{\subsetneq I}$ and consider along each branch of t the first occurrence (starting from the root) of a node with some vector value $\geq B_I$ if one exists. Let n be the number of such first occurrences in t; then t can be written as $C[t_1, \ldots, t_n]$ where C is a context where all the vector values are $< B_I$, and each t_j witnesses $\mathcal{A}_I, q_\ell \triangleright q_j, \mathbf{v}_j$ where $\mathbf{v}_j(i_j) \geq B_I$ for some i_j in I.

1. By repeated shortenings, we can bound the height of C by $|Q| \cdot B_I^{|I|}$.
2. For each j, let $I_j \stackrel{\text{def}}{=} I \setminus \{i_j\}$. Then t_j is also a witness for $\mathcal{A}_{\restriction I_j}, q_\ell \triangleright q_j, \mathbf{v}_{j\restriction I_j}$, and we can replace it by a witness t'_j of height at most H_{I_j}. Then t'_j also witnesses $\mathcal{A}_I, q_\ell \triangleright q_j, \mathbf{v}_j$ because B_I bounds the maximal total decrease that can occur along a branch of a deduction tree of height H_{I_j}.

Hence $t' \stackrel{\text{def}}{=} C[t'_1, \ldots, t'_n]$ is a witness for $(q, \mathbf{v})$ and

$$H_I \leq h(t') \leq |Q| \cdot B_I^{|I|} + H_{\subsetneq I} = |Q| \cdot (\|T_u\|_\infty \cdot H_{\subsetneq I})^{|I|} + H_{\subsetneq I} . \tag{4}$$

Combining (3) with (4), we obtain by induction over d that

$$H_{[d]} \leq (|Q| \cdot (\|T_u\|_\infty + 1) + 1)^{(3d)!} . \tag{5}$$

Observe that this bound is doubly exponential in d, but only exponential in the bitsize $\log(\|T_u\|_\infty + 1)$, and polynomial in the number of states $|Q|$.

3.2 Pareto Frontier

Equation (5) yields an algorithm in AExpSpace = 2-ExpTime to decide given $(q, \mathbf{v})$ in $Q \times \mathbb{N}^d$ whether $\mathcal{A}, q_\ell \triangleright q, \mathbf{v}$: it suffices to look for a minimal witness of height at most $H_{[d]}$, and the vector values in such a witness are bounded by $H_{[d]} \cdot \|T_u\|_\infty$.

Furthermore, as observed by Yen and Chen [18], a bound like (5) that does not depend on the initial configuration $(q, \mathbf{v})$ can be exploited to compute the Pareto frontier: if $(q, \mathbf{v})$ belongs to Pareto$(\mathcal{A}, q_\ell)$, then $\|\mathbf{v}\|_\infty \leq H_{[d]} \cdot \|T_u\|_\infty$. Thus the Pareto frontier can be computed by running the previous algorithm on at most $|Q| \cdot (1 + H_{[d]} \cdot \|T_u\|_\infty)^d$ candidates $(q, \mathbf{v})$:

Proposition 3.2. *Let $\mathcal{A} = \langle Q, d, T_u, T_f \rangle$ be an AVASS and q_ℓ be a state in Q. Then the Pareto frontier* $\text{Pareto}(\mathcal{A}, q_\ell)$ *can be computed in doubly exponential time. If d is fixed it can be computed in exponential time, and if $\|T_u\|_\infty$ is also fixed it can be computed in polynomial time.*

4 Complexity Lower Bounds

In this section, we match the 2-ExpTime upper bound of Thm. 3.1 for state reachability in AVASS (Sec. 4.1). Regarding the fixed dimensional cases, we also show in Sec. 4.2 that our ExpTime upper bound is optimal—note that the case where both the dimension and the bitsize are fixed is trivially PTime-hard by reduction from the emptiness problem for tree automata. These lower bounds on decision problems also entail that our bounds in Thm. 3.2 for the complexity of computing Pareto frontiers are optimal.

4.1 A General Lower Bound

We extend the classical ExpSpace-hardness proof of Lipton [14] for state reachability in VASS to the AVASS case. Instead of reducing from the halting problem for Minsky machines with counter valuations bounded by 2^{2^n}, we reduce instead from the same problem for *alternating* Minsky machines.

More precisely, a Minsky machine can be defined as a VASS with additional *zero-test* rules T_z of the form $q \xrightarrow{i?=0} q'$ for $1 \leq i \leq d$ with deduction semantics

$$\frac{q, \mathbf{v} \quad \mathbf{v}(i) = 0}{q', \mathbf{v}} \text{ (zero-test)}$$

An *alternating* Minsky machine $\langle Q, d, T_u, T_f, T_z \rangle$ can similarly be defined by allowing fork rules. By adapting the usual encoding of Turing machines into Minsky machines to the alternating case, the halting problem for alternating Minsky machines with counter values bounded by 2^{2^n} is hard for AExpSpace = 2-ExpTime. With this in mind, the necessary adaptations of Lipton's reduction are straightforward; see the full paper for details.

Proposition 4.1. *State reachability and non-termination in AVASS are hard for* 2-ExpTime.

Proposition 4.1 was implicit in the 2-ExpTime lower bound proofs of [8, 4] for similar questions. Reducing instead from AVASS would simplify these proofs by separating the extension of Lipton's arguments from the actual reduction.

4.2 Fixed Dimension

Similarly to Thm. 4.1, proving an ExpTime lower bound in the case where the dimension d is fixed is rather easy: Rosier and Yen [17, Thm. 3.1] show indeed that the *boundedness* problem for VASS of dimension $d \geq 4$ is PSpace-hard by reducing from the acceptance problem in linear bounded automata (LBA). Their proof easily extends to the state reachability and non-termination problems for

VASS, and for AVASS by reducing instead from alternating LBA; see the full paper for details.

Proposition 4.2. *State reachability and non-termination in AVASS of fixed dimension $d \geq 4$ are* ExpTime*-hard.*

5 Energy Games

The asymmetric game semantics described in Sec. 2.1 is easily seen to be equivalent to one-sided VASS games as defined in [16, 3]. Such a game is played on a VASS with a partitioned state space $Q = Q_\Diamond \uplus Q_\Box$, where Controller owns the states in $Q_\Diamond$ and can freely manipulate the current vector value, while Environment owns the states in $Q_\Box$ and can only change the current state: if $q_\Box \xrightarrow{\mathbf{u}} q'$ is a rule in T_u and $q_\Box \in Q_\Box$, then $\mathbf{u} = \mathbf{0}$; these restricted Environment rules correspond to AVASS fork rules.

Abdulla et al. [3] have shown the equivalence of AVASS games with the *(multi-dimensional) energy games* of Brázdil et al. [5] and Chatterjee et al. [7], where the asymmetry between Controller and Environment is not enforced in the structure of the AVASS or in restricted unary rules for Environment: in such a game, Environment can use arbitrary unary rules. This would lead to an undecidable state reachability game when played on the $Q \times \mathbb{N}^d$ arena [2], but energy games are played instead over $Q \times \mathbb{Z}^d$—which means that unary rules can be applied even if they yield some negative vector components.

Asymmetry appears instead in the winning conditions for Controller. In addition to a winning condition $\mathit{Win} \subseteq Q^\omega \cup Q^*$ on the sequence of states $q_0, q_1, \ldots$ appearing during the play, Controller must also ensure that all the components of the vectors $\mathbf{v}_0, \mathbf{v}_1, \ldots$ remain non-negative (positive in [5]). Such games are motivated by the synthesis of controllers able to ensure that quantitative values (represented by the integer vectors) are maintained above some critical values.

Various regular winning conditions Win can be employed in this setting: the simplest one is (state) reachability, i.e. $\mathit{Win} = Q^*\{q_\ell\}$, which is in 2-ExpTime by Thm. 3.1. Non-termination, i.e. $\mathit{Win} = Q^\omega$, is shown to be in Tower, i.e. iterated exponential time, by Brázdil et al. [5]. Finally, parity is shown decidable by Abdulla et al. [3]. Theorem 4.1 furthermore entails that state reachability and non-termination (and thus parity) multi-dimensional energy games are 2-ExpTime-hard.

6 Regular Simulations

Jančar and Moller [9] proved in 1995 that the two *regular VASS simulation problems* VASS $\preceq$ FS and FS $\preceq$ VASS, which ask whether a VASS is simulated by a finite-state system (FS) and vice versa, are decidable. They relied however on well quasi orders in their proofs and no complexity upper bounds have been published since. Regarding lower bounds, no improvement has appeared in the general case over the easy ExpSpace-hardness one can derive by reductions from the state reachability and non-termination problems for VASS and the lower bounds of Lipton [14] for these. In the particular case where we restrict ourselves to *basic parallel processes* (BPP) instead of VASS, Kučera and Mayr [11] proved that FS $\preceq$ BPP is PSPACE-hard and BPP $\preceq$ FS is co-NPTIME-hard, and both bounds were later improved to EXPSPACE-hardness by Lasota [12].

By presenting reductions to and from the state reachability and non termination problems in AVASS, we improve on all these results:

– BPP $\preceq$ FS and VASS $\preceq$ FS are both 2-EXPTIME-complete by Thm. 4.1 and Thm. 3.1, and
– FS $\preceq$ BPP and FS $\preceq$ VASS are both 2-EXPTIME-hard by Thm. 4.1 and in TOWER by the results of Brázdil et al. [5].

Abdulla et al. [1] independently showed similar connections between on the one hand the (undecidable) simulation problem PDS $\preceq$ VASS between pushdown systems (PDS) and VASS, and on the other hand energy games played on infinite pushdown graphs. They show that these problems become decidable when the PDS has a singleton stack alphabet and the VASS is 1-dimensional.

6.1 Transition Systems and Simulations

Labelled Transition Systems. Operational semantics are often defined through *labelled transition systems* (LTS) $\mathcal{S} = \langle S, \Sigma, \rightarrow \rangle$ where S is a set of states, Σ is a set of actions, and $\rightarrow \subseteq S \times \Sigma \times S$ is a labelled transition relation, with elements denoted by '$s_1 \xrightarrow{a} s_2$.' When S is finite we call $\mathcal{S}$ a *finite-state system* (FS).

For instance, the operational semantics of a VASS $\mathcal{V} = \langle Q, d, T_u \rangle$ along with a labelling $\sigma : T_u \rightarrow \Sigma$ using a set of actions Σ is the LTS $\mathcal{S}_{\mathcal{V}} \stackrel{\text{def}}{=} \langle Q \times \mathbb{N}^d, \Sigma, \rightarrow \rangle$ with transitions $(q, \mathbf{v}) \xrightarrow{a} (q', \mathbf{v} + \mathbf{u})$ whenever $r = q \xrightarrow{u} q'$ is a unary rule in T_u with label $\sigma(r) = a$ (which we write more simply $q \xrightarrow{u,a} q'$ in the following).

Simulations. Given two LTS $\langle S_1, \Sigma, \rightarrow_1 \rangle$ and $\langle S_2, \Sigma, \rightarrow_2 \rangle$, a *simulation* is a relation $R \subseteq S_1 \times S_2$ such that, whenever (s_1, s_2) belongs to R then for each action a in Σ, if there exists s_1' in S_1 with $s_1 \xrightarrow{a}_1 s_1'$, then there also exists s_2' in S_2 such that $s_2 \xrightarrow{a}_2 s_2'$ and (s_1', s_2') is also in R. A state s_1 is *simulated* by a state s_2, written $s_1 \preceq s_2$, if there exists a simulation R such that (s_1, s_2) is in R.

Simulations can also be characterised by two-player turn-based *simulation games* between 'Spoiler', who wishes to disprove simulation, and 'Duplicator', who aims to establish its existence, played over the arena $S_1 \times S_2$. In a position (s_1, s_2), Spoiler first chooses a transition $s_1 \xrightarrow{a}_1 s_1'$ in $\mathcal{S}_1$, and Duplicator must answer with a transition $s_2 \xrightarrow{a}_2 s_2'$ with the same label a, and the game then proceeds from (s_1', s_2'). A player loses if during one of its turns no suitable transition can be found, otherwise the play is infinite and Duplicator wins. Then $s_1 \preceq s_2$ if and only if Duplicator has a winning strategy starting from (s_1, s_2).

Given two classes of (finitely-presented) systems $\mathbf{A}$ and $\mathbf{B}$, the *simulation problem* $\mathbf{A} \preceq \mathbf{B}$ takes as input two systems A in $\mathbf{A}$ and B in $\mathbf{B}$ with operational semantics $\mathcal{S}_A$ and $\mathcal{S}_B$, and two initial states s_A from $\mathcal{S}_A$ and s_B from $\mathcal{S}_B$, and asks whether $s_A \preceq s_B$. In the following we focus on *regular VASS simulations*, where one of $\mathbf{A}$ and $\mathbf{B}$ is the class of labelled VASS and the other the class FS.

6.2 From Regular VASS Simulations to AVASS

Our two reductions from regular VASS simulations essentially implement the simulation game as an AVASS game. Given a finite set of actions Σ, a labelled VASS defined by $\mathcal{V} = \langle Q, d, T_u \rangle$ and $\sigma\colon T_u \to \Sigma$, a finite-state system $\mathcal{A} = \langle S, \Sigma, \rightarrow_{\mathcal{A}} \rangle$, and a pair of states (q_0, s_0) from $Q \times S$, we construct in both cases a state space $Q' \stackrel{\text{def}}{=} (Q \times S) \uplus (Q \times S \times \Sigma)$ for our AVASS. For convenience we allow forks of arbitrary finite arity $q \to q_1 \wedge \cdots \wedge q_r$.

VASS $\preceq$ FS. We actually reduce in this case from the complement problem VASS $\not\preceq$ FS to AVASS state reachability from (q_0, s_0). Controller plays the role of Spoiler, owns the states in $Q \times S$, and tries to reach the distinguished state q_ℓ. Environment plays the role of Duplicator and owns the states in $Q \times S \times \Sigma$. The rules of the AVASS are then:

$$(q, s) \xrightarrow{\mathbf{u}} (q', s, a) \qquad \text{whenever } q \xrightarrow{\mathbf{u}, a} q' \in T_u \tag{6}$$

$$(q', s, a) \to q_\ell \wedge \bigwedge_{s \xrightarrow{a}_{\mathcal{A}} s'} (q', s') \ . \tag{7}$$

Observe that Spoiler has a winning strategy from (q_0, s_0) in the simulation game if and only if it can force Duplicator into a deadlock, i.e. a state s and an action a where no transition $s \xrightarrow{a}_{\mathcal{A}} s'$ exists. This occurs if and only if Environment can be forced into going to q_ℓ in (7) in the AVASS game starting from (q_0, s_0).

Proposition 6.1. *There is a logarithmic space reduction from* VASS $\not\preceq$ FS *to AVASS state reachability.*

FS $\preceq$ VASS. This direction is actually a particular case of [3, Thm. 5], who show the decidability of *weak simulation* by reducing it to a parity energy game. Environment now plays the role of Spoiler and owns the states in $S \times Q$. Controller now plays the role of Duplicator, owns the states in $S \times Q \times \Sigma$, and attempts to force an infinite play. The rules of the AVASS are then:

$$(s, q) \to \bigwedge_{s \xrightarrow{a}_{\mathcal{A}} s'} (s', q, a) \ , \tag{8}$$

$$(s', q, a) \xrightarrow{\mathbf{u}} (s', q') \qquad \text{whenever } q \xrightarrow{\mathbf{u}, a} q' \in T_u \ . \tag{9}$$

Then, Duplicator has a winning strategy in the simulation game from (q_0, s_0) if and only if Controller has a winning strategy for non-termination in the AVASS game starting in (q_0, s_0):

Proposition 6.2. *There is a logarithmic space reduction from* FS $\preceq$ VASS *to AVASS non-termination.*

6.3 From AVASS to Regular VASS Simulations

Basic Parallel Processes. As announced at the beginning of the section, we prove our lower bounds on the more restricted BPP rather than VASS. Formally, a *BPP net* is a Petri net $\mathcal{N} = \langle P, T, W \rangle$ where P and T are finite sets of places and

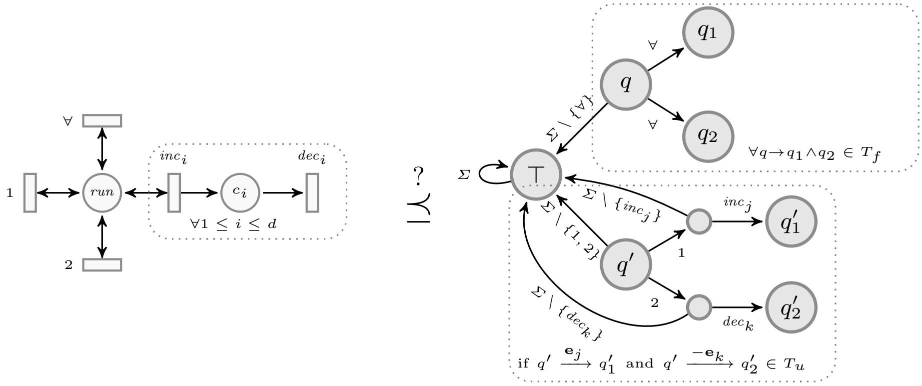

Fig. 1. Reducing AVASS state reachability to a simulation BPP $\not\preceq$ FS

transitions and $W: (P \times T) \cup (T \times P) \to \mathbb{N}$ is the weighted flow, where additionally for all transitions t in T there is exactly one place p in P with $W(p,t) = 1$ and for all $p' \neq p$, $W(p',t) = 0$. Given a labelling function $\sigma: T \to \Sigma$, its semantics is defined by the LTS $\mathcal{S}_\mathcal{N} \stackrel{\text{def}}{=} \langle \mathbb{N}^{|P|}, \Sigma, \to_\mathcal{N} \rangle$ where $m \xrightarrow{a}_\mathcal{N} m - W(P,t) + W(t,P)$ if and only if there exists t with $\sigma(t) = a$ and $m \geq W(P,t)$. In figures we represent places as circles, transitions as rectangles, and positive flows as arrows.

In both our reductions, we want to implement an AVASS game as a simulation game where the FS is in charge of maintaining the state information and the BPP is in charge of maintaining the vector values. We assume we are given an AVASS $\langle Q, d, T_u, T_f \rangle$ in *ordinary form*, i.e. where the only updates vectors in T_u are unit vectors, and in *binary form*, i.e. for each state q of Q, either there is a fork $q \to q_1 \wedge q_2$ (and we call q an *universal state*), or there are exactly two unary rules $q \xrightarrow{\mathbf{u}_1} q_1$ and $q \xrightarrow{\mathbf{u}_2} q_2$ with origin q (and we call it an *existential state*), or there are no applicable rules at all (and we call it a *deadlock state*). We can ensure these two conditions through logarithmic space reductions. Our action alphabet is then defined as $\Sigma \stackrel{\text{def}}{=} \{\forall, \exists, 1, 2\} \cup \{inc_i, dec_i \mid 1 \leq i \leq d\}$.

BPP $\preceq$ FS. We actually reduce AVASS state reachability to BPP $\not\preceq$ FS and assume wlog. that the target state q_ℓ is a deadlock state, and even the only deadlock state by adding rules $q_d \to q_d \wedge q_d$ for the other deadlock states q_d. We construct a BPP net for Spoiler with places $P \stackrel{\text{def}}{=} \{run\} \cup \{c_i \mid 1 \leq i \leq d\}$ where run contains a single token at all times and the c_i's encode the current vector value of the AVASS. Its transitions, labels and flows are depicted on the left of Fig. 1. Its purpose is to force Duplicator, which is playing on the FS depicted on the right of Fig. 1, into state q_ℓ. Because q_ℓ is a deadlock state and Spoiler can always fire transitions (e.g. $\forall$), it then wins the simulation game.

Duplicator plays the role of Environment in the original AVASS game and maintains the AVASS state using its state space, which contains Q. When in a universal state it can choose the following state, but when in an existential state Spoiler chooses instead the branch by firing transition 1 or 2. Duplicator

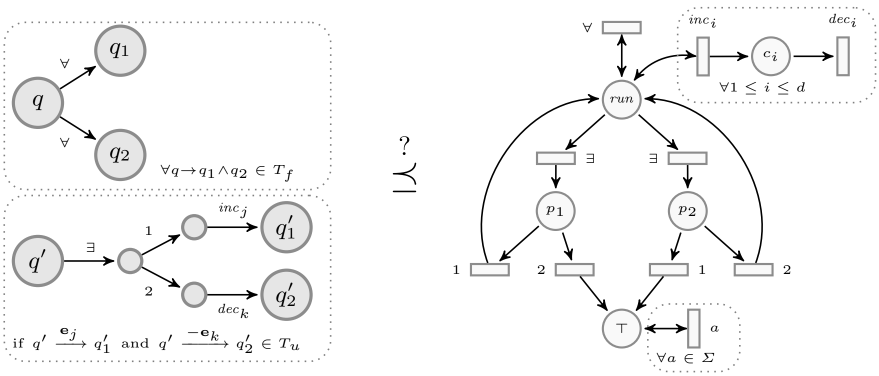

Fig. 2. Reducing AVASS non-termination to a simulation FS $\preceq$ BPP

ensures that the sequence of transitions of Spoiler is indeed valid in the original AVASS, by punishing invalid transitions by entering state '⊤,' where it can play any symbol and thus win the simulation game.

Proposition 6.3. *There is a logarithmic space reduction from AVASS state reachability to* BPP $\not\preceq$ FS*.*

FS $\preceq$ BPP. In this direction we reduce from the non-termination problem. Spoiler now plays in an FS depicted on the left of Fig. 1 and plays for Environment in the original AVASS game. It still maintains the current state of the AVASS in its state space.

Duplicator now plays on a BPP depicted on the right of Fig. 1. It plays the role of Controller in the original VASS game and maintains the vector values in its places c_i as before. We rely on *Duplicator's choice*: using the '∃' label in existential states, Spoiler leaves the choice to Duplicator, who can punish Spoiler—if it does not comply with its choice between actions '1' and '2'—by putting a token in place '⊤', from where it wins.

Proposition 6.4. *There is a logarithmic space reduction from AVASS non-termination to* FS $\preceq$ BPP*.*

7 Concluding Remarks

Alternating VASS provide a unified formalism to reason about VASS games, along with simple complexity arguments for state reachability objectives. This allows us to improve on all the previously known complexity bounds for regular VASS simulations, and show in particular that VASS $\preceq$ FS is 2-ExpTime-complete.

The main open question at this point is whether the upper bounds for non-termination and parity objectives on AVASS could be lowered to 2-ExpTime, and thus to close the gap between 2-ExpTime-hardness and Tower for FS $\preceq$ VASS. A first step to this end could be to extend the PTime upper bound of Chaloupka [6] for the fixed bitsize and unknown initial credit case from dimension two to arbitrary fixed dimensions. However, quoting Chaloupka, 'since the presented results about 2-dimensional VASS are relatively complicated, we suspect this problem is difficult.'

Acknowledgements. The authors thank Stefan Göller who drew our attention to [9, 12] and to the fact that the exact complexities of the two regular simulation problems were unknown. This work also benefited from discussions with Sławomir Lasota, Ranko Lazić, Arnaud Sangnier, and Patrick Totzke.

References

1. Abdulla, P.A., Atig, M.F., Hofman, P., Mayr, R., Kumar, K.N., Totzke, P.: Infinite-state energy games. CSL-LICS 2014. ACM (2014), to appear
2. Abdulla, P.A., Bouajjani, A., d'Orso, J.: Deciding monotonic games. CSL 2003. LNCS, vol. 2803, pp. 1–14. Springer (2003)
3. Abdulla, P.A., Mayr, R., Sangnier, A., Sproston, J.: Solving parity games on integer vectors. Concur 2013. LNCS, vol. 8052, pp. 106–120. Springer (2013)
4. Bérard, B., Haddad, S., Sassolas, M., Sznajder, N.: Concurrent games on VASS with inhibition. Concur 2012. LNCS, vol. 7454, pp. 39–52. Springer (2012)
5. Brázdil, T., Jančar, P., Kučera, A.: Reachability games on extended vector addition systems with states. ICALP 2010. LNCS, vol. 6199, pp. 478–489. Springer (2010)
6. Chaloupka, J.: Z-reachability problem for games on 2-dimensional vector addition systems with states is in P. Fund. Inform. 123(1), 15–42 (2013)
7. Chatterjee, K., Doyen, L., Henzinger, T.A., Raskin, J.F.: Generalized mean-payoff and energy games. FSTTCS 2010. LIPIcs, vol. 8, pp. 505–516. LZI (2010)
8. Demri, S., Jurdziński, M., Lachish, O., Lazić, R.: The covering and boundedness problems for branching vector addition systems. J. Comput. Syst. Sci. 79(1), 23–38 (2012)
9. Jančar, P., Moller, F.: Checking regular properties of Petri nets. Concur '95. LNCS, vol. 962, pp. 348–362. Springer (1995)
10. Kanovich, M.I.: Petri nets, Horn programs, linear logic and vector games. Ann. Pure App. Logic 75(1–2), 107–135 (1995)
11. Kučera, A., Mayr, R.: Simulation preorder over simple process algebras. Inform. and Comput. 173(2), 184–198 (2002)
12. Lasota, S.: ExpSpace lower bounds for the simulation preorder between a communication-free Petri net and a finite-state system. Inf. Process. Lett. 109(15), 850–855 (2009)
13. Lincoln, P., Mitchell, J., Scedrov, A., Shankar, N.: Decision problems for propositional linear logic. Ann. Pure App. Logic 56(1–3), 239–311 (1992)
14. Lipton, R.: The reachability problem requires exponential space. Tech. Rep. 62, Yale University (1976)
15. Rackoff, C.: The covering and boundedness problems for vector addition systems. Theor. Comput. Sci. 6(2), 223–231 (1978)
16. Raskin, J.F., Samuelides, M., Begin, L.V.: Games for counting abstractions. AVoCS 2004. ENTCS, vol. 128(6), pp. 69–85. Elsevier (2005)
17. Rosier, L., Yen, H.C.: A multiparameter analysis of the boundedness problem for vector addition systems. J. Comput. Syst. Sci. 32(1), 105–135 (1986)
18. Yen, H.C., Chen, C.L.: On minimal elements of upward-closed sets. Theor. Comput. Sci. 410(24–25), 2442–2452 (2009)