Accelerating Interpolation-based Model-Checking

Nicolas Caniart, Emmanuel Fleury, Jérôme Leroux and Marc Zeitoun

LaBRI, Université Bordeaux - CNRS UMR 5800,
351 cours de la Libération, F-33405 Talence CEDEX France.
{caniart,fleury,leroux,mz}@labri.fr

Abstract. *Interpolation-based model-checking* and *acceleration* techniques have been widely proved successful and efficient for reachability checking. Surprisingly, these two techniques have never been combined to strengthen each other. Intuitively, acceleration provides under-approximation of the reachability set by computing the exact effect of some control-flow cycles and combining them with other transitions. On the other hand, interpolation-based model-checking is refining an over-approximation of the reachable states based on spurious error-traces. The goal of this paper is to combine acceleration techniques with interpolation-based model-checking at the refinement stage. Our method, called "*interpolant acceleration*", helps to refine the abstraction, ruling out not only a single spurious error-trace but a possibly infinite set of error-traces obtained by any unrolling of its cycles. Interpolant acceleration is also proved to strictly enlarge the set of transformations that can be usually handled by acceleration techniques.

1 Introduction

Counterexample-guided abstraction refinement (CEGAR) paradigm [6] makes it possible to perform efficient verification of real-life software. In this approach (see Fig.1), an initial coarse predicate abstraction [10] of the concrete model is first derived and explored by a model-checker for reachability of error states. If no error path is found, the system is said to be '*safe*'. If an abstract error-trace is found, it is checked against the concrete model. When the error also exists in the concrete model, the system is said to be '*unsafe*' and a concrete error path is provided to the operator. Finally, when the error is found to be spurious, a proof of the spuriousness of the trace is used to build a refinement of the abstraction.

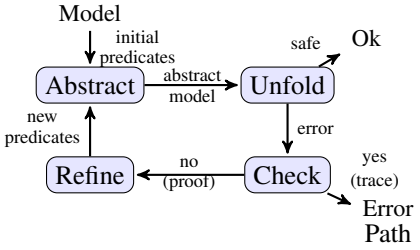


Fig. 1. Interpolant-based Model-Checking.

Interpolation-based model-checking [14, 15] is a CEGAR framework where checking the error-trace is performed using decision procedures for various logics and refinement is produced by computing an interpolant, which provides a set of predicates needed to invalidate the considered spurious error-trace in the abstraction. Interpolation-based model-checking technique has been proved robust and efficient but, as other

CEGAR frameworks, cannot easily handle numerous cycles or infinite behaviors which tend to generate a lot (possibly an infinity) of predicates, while another, better chosen predicate could have captured the whole behavior of the cycle at once. Recently, a '*lazy*' [12] approach of this method has been introduced [16], allowing it to deal with infinite systems. Still, the interpolation-based model-checking technique suffers from a lack of good strategies to efficiently handle infinite behaviors of the input model. As an illustration, consider the example shown on Fig. 2 taken from [13], and well-known in the CEGAR framework [11]. On such a (correct) program, an interpolant-based model-checker might never stop while deriving the predicates to refine the abstraction because an infinity of values of i will have to be checked.

```
void foo(int i, int j) {
  x = i; y = j;
  while (x != 0)
    { x--; y--; }
  if (i == j)
    assert (y == 0);
}
```

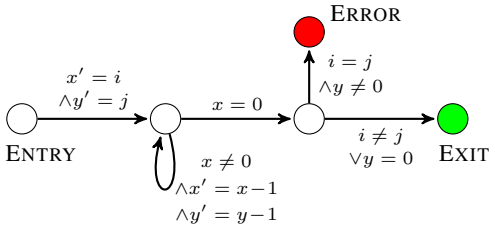


Fig. 2. An example of CEGAR divergence (C code and Control-Flow Automaton (CFA)).

On the other hand, *acceleration techniques* [3, 5, 1] make it possible to check for reachability of infinite systems thanks to a symbolic representation of configurations. Basically, given some suitable control-flow cycle σ fulfilling some properties and a set of states X, acceleration tries to compute the infinite union of all $\sigma^n(X)$. Such a set is called the σ^*-*acceleration set*. It captures the reachable states from X through any unrolling of the cycle σ. Acceleration model-checking is usually performed by adding meta-transitions σ^* to the original model in order to create 'shortcuts' allowing to explore arbitrary iterations of a cycle in one single step, and thus computing the reachable states even for infinite sequences of transitions. For example, systems such as the one presented in Fig. 2 are quite easy to accelerate. Unfortunately, acceleration techniques do not scale up to large systems and termination cannot be ensured.

Intuitively, interpolant-based model-checking focuses on large and simple systems (large number of states, few predicates), where acceleration techniques focus more on small and complex systems. Therefore, our idea is to combine interpolation-based model-checking and acceleration techniques. Interpolation-based model-checking offers a quite helpful automatic abstraction/refinement scheme which can discard unnecessary parts of the system, helping the acceleration technique to deal with smaller chunks to accelerate. Similarly, the acceleration technique can help the interpolation-based model-checking to deal with complex behaviors. We propose here three ways to combine them together:

– **Static Acceleration**: one simply performs static-analysis on the abstract model to detect interesting cycles and adding the corresponding meta-transitions σ^* to the model. This method is quite simple but probably also extremely inefficient because we possibly have to deal with large systems at this stage of the CEGAR for which acceleration would not scale.

– **On-the-fly Acceleration**: While exploring and thus unfolding the abstract model, paths can be processed on-the-fly to detect control loops and check for their conformance to acceleration requirements. Acceleration can then be used to fasten the state-space exploration. This simple method is expected to have a better efficiency as it does not require exhaustive cycle detection. Still, its complexity overhead can be high since many cycles might be found during the unfolding. Heuristics can at last be added to decide whether it is relevant or not to compute an acceleration.
– **Interpolant Acceleration**: Last but not least, we believe this method to be the most promising one, though it can only be applied to lazy interpolant-based model-checking and not to any CEGAR scheme as the previous ones. It takes place at the refining stage, just after identifying an error-trace as '*spurious*' and when computing an interpolant for this trace. Suppose that some suitable cycles σ are found to be such that any unrolling of them are also proved to be spurious. Then, computing the σ^*-acceleration and extracting the interpolant will capture an enlarged set of spurious counter-examples, thus yielding a better abstraction refinement.

We focus here on "*interpolant acceleration*" which reveals to be both theoretically interesting and with room for improvements. We first extend the notion of *interpolant on a path* [16] with the notion of *error-pattern* and *accelerated interpolant* and prove that if an error-pattern is spurious, then there is an accelerated interpolant that will witness every error-trace matching the error-pattern (section 2). We then identify two classes of computable accelerated interpolants: *Presburger* accelerated interpolants (section 4) and *poly-bounded* ones (sections 5 and 6). The first one makes it possible to assess the spuriousness for error-patterns labeled by Presburger transformations, using standard acceleration techniques. The second one allows us to compute interpolants for error-patterns labeled by transformations whose iteration has polynomial, and not only linear, behaviors (*i.e.* which are of the form $\boldsymbol{x}' = M\boldsymbol{x} + \boldsymbol{v}$ where $\boldsymbol{v} \in \mathbb{Z}^n$ and $M \in \mathcal{M}_n(\mathbb{Z})$ is such that the coefficients of its ℓ-th power are bounded by a polynomial in ℓ). Our proof is constructive and can be translated into a (non-optimized) algorithm.

Our work is related to the framework recently presented in [2]. The approach of [2] is to extract "*path programs*", which are sub-graphs of the program leading to errors. In our framework, we aim at capturing a characteristic unfolding of the program leading to the error trace. Where *path programs* can be extremely complex and difficult to exploit for acceleration techniques, *error patterns* tend to be simpler in the way loops interleave and easier to process. On the other hand, *path programs* can capture much more behaviors than *error patterns*.

The remainder of the paper is organized as follows: in Section 2 we recall the notion of interpolant, introduce '*accelerated interpolant*' and relate it to set separability. We recall basics on linear algebra and characterize the class of transformations that our method can handle in Section 3. In Section 4, we rephrase the problem of computing an accelerated interpolant for Presburger sets in more suitable terms. We then reduce this latter problem in Section 5. Finally, using these intermediate results, Section 6 describes how to compute an accelerated interpolant for our class of linear transformations and two Presburger definable sets, one of which is finite. At last, we show that the finiteness condition for one of the Presburger sets cannot be dropped.

2 Introducing Accelerated Interpolants

The need for *interpolants* in the CEGAR loop of *interpolation-based model-checking* arises during the refinement step. More precisely, if we assume the input program of the CEGAR loop to be given as a *control-flow automaton* [12] (CFA), an abstraction of this one will be unfolded and explored to find an *error-trace*. In case one is found, the algorithm tries to checks if it witnesses a real error-path or appears as a side effect of a too coarse abstraction. In the latter case the trace is said *spurious*. Finally, if proved spurious, abstraction is refined to rule out the spurious error-trace thanks to the computation of an interpolant capturing this trace.

Formally a *CFA* is a tuple $G = (Q, q_{ini}, q_{err}, \mathbb{D}, T)$ where Q is the finite set of *control-states*, $q_{ini} \in Q$ is the *initial state*, $q_{err} \in Q$ is the *error state*, $\mathbb{D}$ is a possibly infinite set representing the data domain, and T is a finite set of transitions $t = (q, r_t, q')$ with $q, q' \in Q$ and $r_t \subseteq \mathbb{D} \times \mathbb{D}$. Intuitively, the binary relations over $\mathbb{D} \times \mathbb{D}$ can be used either to encode guards, or to encode updates (see for instance Example 4.2). A *trace* $\pi = t_0 \cdots t_k$ is a word of transitions $t_i \in T$ such that there exist $q_0, \ldots, q_{k+1} \in Q$ and $r_0, \ldots, r_k \subseteq \mathbb{D} \times \mathbb{D}$ with $t_i = (q_i, r_i, q_{i+1})$ for $0 \leq i \leq k$. Such a trace is also denoted $\pi = q_0 \xrightarrow{r_0} q_1 \cdots \xrightarrow{r_k} q_{k+1}$, or just $q_0 \xrightarrow{r_\pi} q_{k+1}$ with $r_\pi = r_0 \cdots r_k$[1] . It is called an *error-trace* if $q_0 = q_{ini}$ and $q_{k+1} = q_{err}$. It is a *cycle* if $q_{k+1} = q_0$. We denote by $r^* = \bigcup_{\ell \in \mathbb{N}} r^\ell$ the reflexive and transitive closure of a binary relation $r \subseteq \mathbb{D} \times \mathbb{D}$ where r^ℓ denotes the ℓ-th power of r.

Semantically, a CFA defines a *labeled transition system* given by the set of *configurations* $Q \times \mathbb{D}$ and the binary relations $\xrightarrow{r}$ over the set of configurations by $(q, d) \xrightarrow{r} (q', d')$ if $q \xrightarrow{r} q'$ and $(d, d') \in r$. A *path* is an alternating sequence of configurations and binary relations $(q_0, d_0) \xrightarrow{r_0} (q_1, d_1) \cdots \xrightarrow{r_k} (q_{k+1}, d_{k+1})$. A *concretization* of a trace $q_0 \xrightarrow{r_0} q_1 \cdots \xrightarrow{r_k} q_{k+1}$ is a path of the form $(q_0, d_0) \xrightarrow{r_0} (q_1, d_1) \cdots \xrightarrow{r_k} (q_{k+1}, d_{k+1})$, unambiguously abusing the $\xrightarrow{r_k}$ notation, for the sake of simplicity.

Definition 2.1. *An error-trace is said* spurious *if it does not have a concretization.*

By definition, the existence of a concretization is sufficient to certify that an error-trace is not spurious. Let us now recall why a sequence of sets $X_0, \ldots, X_k$ called an *interpolant* can certify that an error-trace *is* spurious. Let introduce few notations, given $X, X' \subseteq \mathbb{D}$ and $r \subseteq \mathbb{D} \times \mathbb{D}$, let $\mathsf{post}_r(X) = \{d' \mid \exists\, d\, (d, d') \in r \wedge d \in X\}$ and $\mathsf{wpre}_r(X') = \{d \mid \forall\, d'\, (d, d') \in r \Rightarrow d' \in X'\}$. Recall that $(\mathsf{post}_r(), \mathsf{wpre}_r())$ forms a *Galois connection*, since clearly $\mathsf{post}_r(X) \subseteq X'$ iff $X \subseteq \mathsf{wpre}_r(X')$. If these inclusions hold true, we write $X \xrightarrow{r} X'$. Moreover if $X = X'$ then X is called an r-invariant.

Definition 2.2. *A sequence $X_0, \ldots, X_{k+1}$ of subsets of $\mathbb{D}$ is called an* interpolant *for a decomposition $\pi_0, \ldots, \pi_k$ of an error-trace $\pi = \pi_0 \ldots \pi_k$ if:*

$$\mathbb{D} = X_0 \xrightarrow{r_{\pi_0}} X_1 \cdots X_k \xrightarrow{r_{\pi_k}} X_{k+1} = \varnothing$$

Thus, the existence of an interpolant witnesses the spuriousness of an error-trace. Conversely, we would like to establish that if an error-trace is spurious, then there exists an interpolant. This immediately follows from [7, Propositions 1&2].

[1] By convention r_π is the identity binary relation if π is the empty word of T^*.

Proposition 2.3 ([7, Propositions 1&2]). *An error-trace $\pi_0 \cdots \pi_k$ is spurious if and only if there exists an interpolant $(X_i)_{0\leq i\leq k+1}$. In this case $(\mathsf{post}_{r_{\pi_0}\ldots r_{\pi_{i-1}}}(\mathbb{D}))_{0\leq i\leq k+1}$ and $(\mathsf{wpre}_{r_{\pi_i}\ldots r_{\pi_k}}(\varnothing))_{0\leq i\leq k+1}$ are interpolants and we have:*

$$\forall\, 0 \leq i \leq k+1 \quad \mathsf{post}_{r_{\pi_0}\ldots r_{\pi_{i-1}}}(\mathbb{D}) \subseteq X_i \subseteq \mathsf{wpre}_{r_{\pi_i}\ldots r_{\pi_k}}(\varnothing).$$

Thus, an error-trace is spurious iff one can find an interpolant witnessing its spuriousness. Unfortunately, using this property, the classical CEGAR scheme may only discard error-traces one by one. Consider the case where a trace contains cycles forming *error-patterns*. We would like then to discard *every* error-traces matching a pattern *at once* (whatever is the number of iterations along each cycle). That is, we would like to prove that an error-pattern is spurious, not only a single error-trace. More formally:

Definition 2.4. *An* error-pattern *is a sequence $(\pi_0, \theta_1, \pi_1, \ldots, \theta_k, \pi_k)$ where each π_i is a trace and each θ_i is a cycle, of the following form:*

$$q_{ini} = q_0 \xrightarrow{r_{\pi_0}} \underset{\circlearrowright\, r_{\theta_1}}{q_1} \xrightarrow{r_{\pi_1}} \underset{\circlearrowright\, r_{\theta_2}}{q_2} \quad \cdots \quad \underset{\circlearrowright\, r_{\theta_k}}{q_k} \xrightarrow{r_{\pi_k}} q_{k+1} = q_{err}$$

Note that, by extension, an error-pattern $(\pi_0, \theta_1, \pi_1 \ldots, \theta_k, \pi_k)$ is said *spurious* if all error-traces in $\pi_0\theta_1^*\pi_1 \ldots \theta_k^*\pi_k$ are spurious.

Definition 2.5 (Accelerated Interpolant). *A sequence $X_0, \ldots, X_{k+1}$ of subsets of $\mathbb{D}$ is called an* accelerated interpolant *for an error-pattern $(\pi_0, \theta_1, \pi_1, \ldots, \theta_k, \pi_k)$ if:*

$$\mathbb{D} = X_0 \xrightarrow{r_{\pi_0}} \underset{\circlearrowright\, r_{\theta_1}}{X_1} \xrightarrow{r_{\pi_1}} \underset{\circlearrowright\, r_{\theta_2}}{X_2} \quad \cdots \quad \underset{\circlearrowright\, r_{\theta_k}}{X_k} \xrightarrow{r_{\pi_k}} X_{k+1} = \varnothing$$

That is, in order for an interpolant $X_0, \ldots, X_{k+1}$ for $(\pi_0, \pi_1, \ldots, \pi_k)$ to be an accelerated interpolant for the error-pattern $(\pi_0, \theta_1, \pi_1, \ldots, \theta_k, \pi_k)$, we require in addition that each X_i is an r_{θ_i}-invariant, for $1 \leq i \leq k$. Once again, it is easy to check that accelerated interpolants characterize spurious error-patterns.

Lemma 2.6. *Let $(\pi_0, \theta_1, \pi_1, \ldots, \theta_k, \pi_k)$ be an error-pattern, $r_{\theta_0} = r_{\theta_{k+1}}$ be the identity relation on $\mathbb{D}$, and $p_i, s_i \subseteq \mathbb{D} \times \mathbb{D}$ defined by $p_i = r_{\theta_0}^* r_{\pi_0} r_{\theta_1}^* \cdots r_{\pi_{i-1}} r_{\theta_i}^*$, $s_i = r_{\theta_i}^* r_{\pi_i} \cdots r_{\theta_k}^* r_{\pi_k}$. The error-pattern is spurious if and only if there exists an accelerated interpolant $(X_i)_{0\leq i\leq k+1}$. Moreover, in this case both $(\mathsf{post}_{p_i}(\mathbb{D}))_{0\leq i\leq k+1}$ and $(\mathsf{wpre}_{s_i}(\varnothing))_{0\leq i\leq k+1}$ are accelerated interpolants such that:*

$$\forall\, 0 \leq i \leq k+1 \quad \mathsf{post}_{p_i}(\mathbb{D}) \subseteq X_i \subseteq \mathsf{wpre}_{s_i}(\varnothing).$$

Corollary 2.7. *An error-pattern is spurious iff there exists an accelerated interpolant.*

We now investigate the computation of accelerated interpolants for error-patterns containing one single cycle. We show that the accelerated interpolation problem for such an error-pattern (π_0, θ, π_1) reduces to a separation problem.

Definition 2.8. *Given a binary relation r, a set X is called a* r-separator *for a pair of sets (E, F) if $X \xrightarrow{r} X$, $E \subseteq X$ and $X \cap F = \varnothing$. If such a set exists, (E, F) is said* r-separable*.*

In fact, observe that $(\mathbb{D}, X, \varnothing)$ is an accelerated interpolant for (π_0, θ, π_1) iff X is an r_θ-separator for (E, F) where $E = \mathsf{post}_{r_{\pi_0}}(\mathbb{D})$ and $F = \mathsf{wpre}_{r_{\pi_1}}(\varnothing)$.

We have shown that if an error-pattern is spurious, then there exist an interpolant that will witness it spuriousness. But, to find an interpolant for a given a error-pattern, we need to be able to compute or approximate the relations $r^*_{\theta_i}$. Considering the fact that the set of error-traces matching a pattern may be infinite, it is obvious that this is not possible in general. This is the question addressed in the next sections.

3 Some Notes on Linear Algebra

The method we present in the next sections computes accelerated interpolants for an error-pattern with one single cycle θ whose associated binary relation is $\boldsymbol{x}\ r_\theta\ \boldsymbol{y}$ if and only if $\boldsymbol{y} = M\boldsymbol{x} + \boldsymbol{v}$, where $\boldsymbol{v} \in \mathbb{Z}^n$ and $M \in \mathcal{M}_n(\mathbb{Z})$ is such that the coefficients of its ℓ-th power are bounded by a polynomial in ℓ. In this section, we first briefly recall some material about matrices, and then characterize these integer matrices whose ℓ-th power is polynomially bounded in ℓ.

Considering $\mathbb{K} \in \{\mathbb{C}, \mathbb{R}, \mathbb{Q}, \mathbb{Z}\}$, we denote by $\mathcal{M}_n(\mathbb{K})$ the set of n-dim *square matrices* with coefficients in $\mathbb{K}$. The n-dim *identity matrix* and the *zero matrix* are respectively denoted by I_n and 0_n. The *inverse* of an *invertible matrix* P is denoted by P^{-1}. The matrix M^ℓ, where $\ell \in \mathbb{N}$, denotes the ℓ-th power of M. The multiplicative monoid $\{M^\ell \mid \ell \in \mathbb{N}\}$ is denoted by M^*. Given $S \subseteq \mathbb{K}^n$, we let $MS = \{M\boldsymbol{x} \mid \boldsymbol{x} \in S\}$ and $M^*S = \bigcup_{\ell=0}^{\infty} M^\ell S$. Two matrices M_1, M_2 *commute* if $M_1M_2 = M_2M_1$. An n-dim matrix Δ is said *diagonal* if $\Delta_{ij} = 0$ whenever $i \neq j$. A matrix D is said *diagonalizable* if there exists a diagonal matrix $\Delta \in \mathcal{M}_n(\mathbb{C})$ and an invertible matrix $P \in \mathcal{M}_n(\mathbb{C})$ such that $D = P\Delta P^{-1}$. A matrix N is said *nilpotent* if there exists $\ell \in \mathbb{N} \setminus \{0\}$ such that $N^\ell = 0_n$. Remember that $N^n = 0_n$. A set $S \subseteq \mathbb{K}^n$ is called an M*-invariant* if $MS \subseteq S$. An M-invariant S is called an M*-attractor* for a vector $\boldsymbol{x} \in \mathbb{K}^n$ if there exists $\ell_0 \in \mathbb{N}$ such that $M^{\ell_0}\boldsymbol{x} \in S$. Observe that $M^\ell \boldsymbol{x} \in S$ for any $\ell \geq \ell_0$ since S is an M-invariant.

Let $L_m(X) = \frac{1}{!m} X \cdots (X - m + 1)$ be the m-th *Lagrange polynomial*. The *binomial theorem* states that for every pair (M_1, M_2) of commuting matrices and for every $\ell \in \mathbb{N}$:

$$(M_1 + M_2)^\ell = \sum_{m=0}^{\ell} L_m(\ell) M_1^{\ell-m} M_2^m$$

Observe that a matrix $M \in \mathcal{M}_n(\mathbb{Z})$ generates a finite monoid M^* if and only if the coefficients of M^ℓ are bounded independently of ℓ. And the finiteness of M^* is decidable in polynomial time [3]. We are going to show that the Dunford decomposition algorithmically characterizes the set of matrices $M \in \mathcal{M}_n(\mathbb{Z})$ such that the coefficients of M^ℓ are *polynomially bounded* in ℓ. Recall that the *Dunford decomposition theorem* proves that any matrix M can be uniquely decomposed into a pair (D, N) of commuting matrices of $\mathcal{M}_n(\mathbb{C})$ such that $M = D + N$, where D is diagonalizable and

N is nilpotent. Moreover if $M \in \mathcal{M}_n(\mathbb{Q})$ then $D, N \in \mathcal{M}_n(\mathbb{Q})$ are effectively computable in polynomial time[2]. In particular, we can decide in polynomial time if a matrix $M \in \mathcal{M}_n(\mathbb{Q})$ is diagonalizable. In fact, a matrix M is diagonalizable if and only if its Dunford decomposition (D, N) satisfies $D = M$ and $N = 0_n$.

Definition 3.1. *A matrix M is* poly-bounded *if all the coefficients of M^ℓ are polynomially bounded in ℓ.*

Proposition 3.2. *A matrix $M \in \mathcal{M}_n(\mathbb{Z})$ is poly-bounded if and only if the Dunford decomposition (D, N) of M is such that D^* is finite.*

Example 3.3. Below is a poly-bounded matrix, and the transition system it encodes.

$$M = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} \text{ is poly-bounded as } M^\ell = \begin{pmatrix} 1 & \ell & \frac{\ell(\ell-1)}{2} \\ 0 & 1 & \ell \\ 0 & 0 & 1 \end{pmatrix}$$

$$\begin{array}{l} x_1' = x_1 + x_2 \\ x_2' = x_2 + x_3 \\ x_3' = x_3 \end{array}$$

4 Presburger Accelerated Interpolants

In this section, we focus on the expressive power of the Presburger logic for effectively computing accelerated interpolants.

Presburger logic [17] is a first-order additive arithmetic theory over the integers. This decidable logic is used in a large range of applications such as compiler optimization, program analysis and model-checking. A set $Z \subseteq \mathbb{Z}^n$ that can be encoded by a formula $\phi(\boldsymbol{x})$ in this logic is called a *Presburger set*. In this paper, we use two geometrical characterizations of the Presburger sets respectively based on *linear sets* and *linear constraints*. A *linear set* Z is a set of the form $Z = \boldsymbol{a} + P^*$ where $\boldsymbol{a} \in \mathbb{Z}^n$, P is a finite subset of $\mathbb{Z}^n$ and P^* denotes the set of finite sums $\sum_{i=1}^{k} \boldsymbol{p_i}$ with $\boldsymbol{p_1}, \ldots, \boldsymbol{p_k} \in P$ and $k \in \mathbb{N}$. Recall that a set is Presburger if and only if it is equal to a finite union of linear sets [9]. A *linear constraint* is either an inequality constraint $\langle \boldsymbol{\alpha}, \boldsymbol{x} \rangle \leq c$ or a modular constraint $\langle \boldsymbol{\alpha}, \boldsymbol{x} \rangle \equiv_b c$ where $\boldsymbol{\alpha} \in \mathbb{Z}^n$, $b \in \mathbb{N} \setminus \{0\}$, $c \in \mathbb{Z}$ and where $\langle \boldsymbol{\alpha}, \boldsymbol{x} \rangle = \sum_{i=1}^{n} \alpha_i x_i$ denotes the *dot product* of $\boldsymbol{\alpha}$ and $\boldsymbol{x}$, and $\equiv_b$ denotes the *equivalence binary relation* over $\mathbb{Z}$ satisfying $z_1 \equiv_b z_2$ if and only if b divides $z_1 - z_2$. A *quantification elimination* shows that a set $Z \subseteq \mathbb{Z}^n$ is Presburger if and only if it can be encoded by a propositional formula of linear constraints (*i.e.*, a quantifier-free Presburger formula).

A CFA $G = (Q, q_{ini}, q_{err}, \mathbb{Z}^n, T)$ is said *Presburger* if $\boldsymbol{x}\ r_t\ \boldsymbol{x}'$ is encoded by a Presburger formula $\phi_t(\boldsymbol{x}, \boldsymbol{x}')$ for any transition $t \in T$. We say that an interpolant (resp. accelerated interpolant) $X_0, \ldots, X_{k+1}$ is *Presburger* if the sets $X_0, \ldots, X_{k+1}$ are Presburger. Since Presburger logic is decidable, observe that the spuriousness problem for error-traces of Presburger CFA is decidable and that we can effectively compute a Presburger interpolant.

Concerning the computation of Presburger accelerated interpolants, observe that the reachability problem for *Minsky machines* can be reduced to the spuriousness problem

[2] A possible algorithm consists in computing $P = \chi_M / \gcd(\chi_M, \chi_M')$, where χ_M denotes the characteristic polynomial of M, and the sequence defined by $D_0 = M$, and $D_{k+1} = D_k - P(D_k) \circ (P'(D_k))^{-1}$, which is well-defined and stabilizes to D after $O(\log n)$ iterations.

of an error-pattern of the form (π_0, θ, π_1) where π_0 intuitively initialized the Minsky machine, π_1 tests if the final state is reached and θ encodes the one step reachability relation of the machine. This reduction shows that the spuriousness problem for error-patterns of Presburger CFA is undecidable. However, observe that if there exists a Presburger accelerated interpolant, such an interpolant can be effectively computed with an enumerative approach. In fact, the set of Presburger accelerated interpolants is recursively enumerable since it is sufficient to fairly enumerate the sequences of Presburger formulas $\phi_0(\boldsymbol{x}), \ldots, \phi_{k+1}(\boldsymbol{x})$ and checks if such a sequence effectively encodes an accelerated interpolant.

Naturally, such an enumerative algorithm has no practical interest. This explains why we focus on error-pattern classes admitting Presburger accelerated interpolants based on a non enumerative algorithm. Acceleration techniques provide such a class. The following theorem shows that if θ is a control-flow cycle such that $\boldsymbol{x}\ r_\theta\ \boldsymbol{y}$ iff $\boldsymbol{y} = M\boldsymbol{x} + \boldsymbol{v}$ where $\boldsymbol{v} \in \mathbb{Z}^n$ and $M \in \mathcal{M}_n(\mathbb{Z})$ has a finite monoid M^*, then we can effectively compute a Presburger formula encoding the binary relation r_θ^*. Thus, if the cycles of an error-pattern satisfy the finite monoid condition, we can effectively decide the spuriousness, and in this case we can effectively compute a Presburger accelerated interpolant. Observe that the obtained interpolant does not use the fact that r_θ^* can be approximated, whereas the definition of accelerated interpolants does not require a precise computation of this relation.

Proposition 4.1 (Acceleration [4, 8]). *A binary relation r over $\mathbb{Z}^n$ such that $\boldsymbol{x}\ r\ \boldsymbol{y}$ if and only if $\boldsymbol{y} = M\boldsymbol{x} + \boldsymbol{v}$ where $\boldsymbol{v} \in \mathbb{Z}^n$ and $M \in \mathcal{M}_n(\mathbb{Z})$ satisfies r^* is Presburger if and only if M^* is finite. Moreover, in this case we can compute a Presburger formula $\phi(\boldsymbol{x}, \boldsymbol{y})$ encoding $\boldsymbol{x}\ r^*\ \boldsymbol{y}$.*

Here is an example of spurious error-pattern with no Presburger accelerated interpolant.

Example 4.2. Let $n = 2$ and consider the CFA G_1 depicted in Fig 3. Intuitively, t_{ini} reset two integer variables x_1 and x_2, t_1 and t_2 are two deterministic loops such that r_{t_2} performs the inverse of r_{t_1}, t does not modify the variables, and t_{err} tests if $x_1 = 0$ and $x_2 > 0$. More formally $G_1 = (Q, q_{ini}, q_{err}, \mathbb{Z}^2, T)$ where $Q = \{q_{ini}, q_1, q_2, q_{err}\}$ and where $T = \{t_{ini}, t_1, t, t_2, t_{err}\}$ is defined by :

$$
\begin{array}{lll}
t_{ini} = (\ q_{ini}\ ,\ r_{ini}\ ,\ q_1\ ) & \text{with } (\boldsymbol{x}, \boldsymbol{x}') \in r_{ini} & \text{iff } x_1' = 0 \wedge x_2' = 0 \\
t_1 = (\ q_1\ ,\ r_1\ ,\ q_1\ ) & \text{with } (\boldsymbol{x}, \boldsymbol{x}') \in r_1 & \text{iff } x_1' = x_1 + 1 \wedge x_2' = x_2 + x_1 \\
t = (\ q_1\ ,\ r\ ,\ q_2\ ) & \text{with } (\boldsymbol{x}, \boldsymbol{x}') \in r & \text{iff } x_1' = x_1 \wedge x_2' = x_2 \\
t_2 = (\ q_2\ ,\ r_2\ ,\ q_2\ ) & \text{with } (\boldsymbol{x}, \boldsymbol{x}') \in r_2 & \text{iff } x_1 = x_1' + 1 \wedge x_2 = x_2' + x_1' \\
t_{err} = (\ q_2\ ,\ r_{err}\ ,\ q_{err}\ ) & \text{with } (\boldsymbol{x}, \boldsymbol{x}') \in r_{err} & \text{iff } x_1 = 0 \wedge x_2 > 0
\end{array}
$$

Observe that $(\mathbb{Z}^2, X, X, \varnothing)$ where $X = \{\boldsymbol{x} \in \mathbb{Z}^2 \mid x_2 = \frac{x_1(x_1-1)}{2}\}$ is an accelerated interpolant for the error-pattern $(t_{ini}, t_1, t, t_2, t_{err})$. In particular this error-pattern is spurious. Unfortunately X is not a Presburger set. Actually, the following lemma shows that it is hopeless to try computing a Presburger accelerated interpolant.

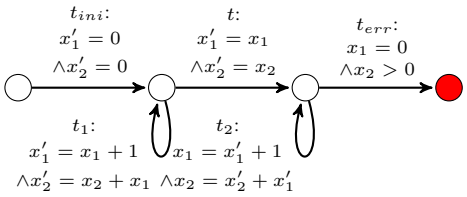


Fig. 3. The CFA G_1

Lemma 4.3. *There does not exist a Presburger accelerated interpolant for the spurious error-pattern* $(t_{ini}, t_1, t, t_2, t_{err})$.

Proof. Let us consider an accelerated interpolant $(\mathbb{Z}^2, X_1, X_2, \varnothing)$ for the spurious error-pattern $(t_{ini}, t_1, t, t_2, t_{err})$ and assume by contradiction that X_1 is a Presburger set. By replacing X_1 by $X_1 \cap \mathbb{N}^2$, we can assume without loss of generality that $X_1 \subseteq \mathbb{N}^2$. Let us consider a Presburger formula $\phi_1(\boldsymbol{x})$ encoding X_1. An immediate induction proves that $\mathsf{post}_{r_{ini} r_1^*}(\mathbb{Z}^2)$ is equal to $X' = \{\boldsymbol{x} \in \mathbb{N}^2 \mid x_2 = \frac{x_1(x_1-1)}{2}\}$. As $(\mathbb{Z}^2, X_1, X_2, \varnothing)$ is an interpolant, we deduce that $X' \subseteq X_1$. Note that if $X_1 \cap \{\boldsymbol{x} \in \mathbb{N}^2 \mid x_2 > \frac{x_1(x_1-1)}{2}\}$ is empty then X' is encoded by the Presburger formula $\phi'(\boldsymbol{x}') := \phi_1(\boldsymbol{x}') \wedge \forall x_2 \phi_1(x_1', x_2) \implies x_2 \leq x_2'$. As X' is not a Presburger set we deduce that this intersection is not empty. Thus, there exists $\boldsymbol{x} \in X_1$ such that $x_2 > \frac{x_1(x_1-1)}{2}$. Now, just observe that $\mathsf{post}_{r r_2^{x_2} r_{err}}(\{\boldsymbol{x}\})$ shows that $\mathsf{post}_{r r_2^* r_{err}}(X_1) \neq \varnothing$ which is in contradiction with the fact that $(\mathbb{Z}^2, X_1, X_2, \varnothing)$ is an accelerated interpolant. □

5 Half-Space Attractors

In this section we provide an algorithm for solving the following convergence decision problem: given the *half-space*

$$H(\boldsymbol{\alpha}, c) = \{\boldsymbol{x} \in \mathbb{R}^n \mid \langle \boldsymbol{\alpha}, \boldsymbol{x} \rangle \geq c\}$$

with $\boldsymbol{\alpha} \in \mathbb{Z}^n$ and $c \in \mathbb{Z}$, and a matrix $M \in \mathcal{M}_n(\mathbb{R})$ such that $N = M - I_n$ is nilpotent, decide whether a vector $\boldsymbol{x} \in \mathbb{Z}^n$ satisfies $M^\ell \boldsymbol{x} \in H(\boldsymbol{\alpha}, c)$ for some $\ell \in \mathbb{N}$. This algorithm will be crucial for computing accelerated interpolants, in the next section.

We first show that the following two sets can be decomposed into an effectively computable Boolean combination of half-spaces:

$$E_-(\boldsymbol{\alpha}, c) = \{\boldsymbol{x} \in \mathbb{R}^n \mid \exists \ell_0 \in \mathbb{N}, \forall \ell \geq \ell_0, M^\ell \boldsymbol{x} \notin H(\boldsymbol{\alpha}, c)\},$$
$$E_+(\boldsymbol{\alpha}, c) = \{\boldsymbol{x} \in \mathbb{R}^n \mid \exists \ell_0 \in \mathbb{N}, \forall \ell \geq \ell_0, M^\ell \boldsymbol{x} \in H(\boldsymbol{\alpha}, c)\}.$$

It is clear that $E_-(\boldsymbol{\alpha}, c)$ and $E_+(\boldsymbol{\alpha}, c)$ are disjoint (the decomposition proof will show in addition that $\mathbb{R}^n = E_-(\boldsymbol{\alpha}, c) \cup E_+(\boldsymbol{\alpha}, c)$). Recall that $L_m(X)$ denotes the Lagrange polynomial. Since $N^n = 0_n$ and $L_m(\ell) = 0$ for any $m > \ell$, the binomial theorem applied to the commutative matrices I_n and N yields:

$$\langle \boldsymbol{\alpha}, M^\ell \boldsymbol{x} \rangle = \sum_{m=0}^{n-1} L_m(\ell) \langle \boldsymbol{\alpha}, N^m \boldsymbol{x} \rangle \tag{1}$$

We introduce the sets $Z_k(\boldsymbol{\alpha})$ for $k \in \mathbb{Z}$. First, $Z_0(\boldsymbol{\alpha}) = \{\boldsymbol{x} \in \mathbb{R}^n \mid \bigwedge_{j \geq 1} \langle \boldsymbol{\alpha}, N^j \boldsymbol{x} \rangle = 0\}$ and, for $\varepsilon \in \{-1, +1\}$ and $m \in \mathbb{N} \setminus \{0\}$:

$$Z_{\varepsilon.m}(\boldsymbol{\alpha}) = \{\boldsymbol{x} \in \mathbb{R}^n \mid \varepsilon.\langle \boldsymbol{\alpha}, N^m \boldsymbol{x} \rangle > 0 \wedge \bigwedge_{j > m} \langle \boldsymbol{\alpha}, N^j \boldsymbol{x} \rangle = 0\}.$$

Clearly, the $Z_k(\boldsymbol{\alpha})$ are pairwise disjoint, $Z_k(\boldsymbol{\alpha}) = \varnothing$ if $|k| \geq n$, and $\bigcup_{k \in \mathbb{Z}} Z_k(\boldsymbol{\alpha}) = \mathbb{R}^n$.

Lemma 5.1. *Let $\boldsymbol{\alpha}, \boldsymbol{x} \in \mathbb{R}^n$. We have:*

$$\lim_{\ell \to +\infty} \langle \boldsymbol{\alpha}, M^\ell \boldsymbol{x} \rangle = \begin{cases} +\infty & \text{if } \boldsymbol{x} \in \bigcup_{k \geq 1} Z_k(\boldsymbol{\alpha}), \\ \langle \boldsymbol{\alpha}, \boldsymbol{x} \rangle & \text{if } \boldsymbol{x} \in Z_0(\boldsymbol{\alpha}), \\ -\infty & \text{if } \boldsymbol{x} \in \bigcup_{k \leq -1} Z_k(\boldsymbol{\alpha}). \end{cases}$$

From the previous lemma, we deduce the expression of $E_-(\boldsymbol{\alpha}, c)$ and $E_+(\boldsymbol{\alpha}, c)$:

$$E_-(\boldsymbol{\alpha}, c) = (Z_0(\boldsymbol{\alpha}) \setminus H(\boldsymbol{\alpha}, c)) \cup \bigcup_{k \leq -1} Z_k(\boldsymbol{\alpha}), \tag{2}$$

$$E_+(\boldsymbol{\alpha}, c) = (Z_0(\boldsymbol{\alpha}) \cap H(\boldsymbol{\alpha}, c)) \cup \bigcup_{k \geq 1} Z_k(\boldsymbol{\alpha}). \tag{3}$$

Naturally, if $\boldsymbol{x} \in E_+(\boldsymbol{\alpha}, c)$ we can conclude that there exists $\ell \in \mathbb{N}$ such that $M^\ell \boldsymbol{x} \in H(\boldsymbol{\alpha}, c)$. On the other hand, if $\boldsymbol{x} \in E_-(\boldsymbol{\alpha}, c)$ we cannot conclude that $M^\ell \boldsymbol{x} \notin H(\boldsymbol{\alpha}, c)$ for all $\ell \in \mathbb{N}$. We are going to characterize a set $X_-(\boldsymbol{\alpha}, c)$ with an empty intersection with $H(\boldsymbol{\alpha}, c)$ that is an M-attractor for any vector $\boldsymbol{x} \in E_-(\boldsymbol{\alpha}, c)$. Thus, if $\boldsymbol{x} \in E_-(\boldsymbol{\alpha}, c)$, it suffices to compute the beginning of the sequence $M^\ell \boldsymbol{x}$ until we discover ℓ such that $M^\ell \boldsymbol{x}$ is in $H(\boldsymbol{\alpha}, c)$ or $X_-(\boldsymbol{\alpha}, c)$. In the first case there must be an ℓ such that $M^\ell \boldsymbol{x} \in H(\boldsymbol{\alpha}, c)$ and in the second case we can tell that $M^\ell \boldsymbol{x} \notin H(\boldsymbol{\alpha}, c)$ for every $\ell \in \mathbb{N}$. The situation is shown in Fig. 4.

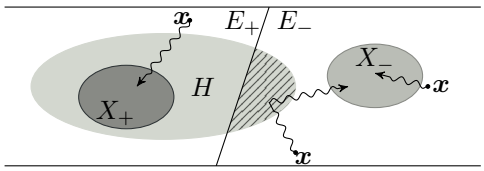


Fig. 4. Likely trajectories of $M^\ell \boldsymbol{x}$, omitting $(\boldsymbol{\alpha}, c)$

We define the two sets $X_-(\boldsymbol{\alpha}, c)$ and $X_+(\boldsymbol{\alpha}, c)$ as follows:

$$X_-(\boldsymbol{\alpha}, c) = \{\boldsymbol{x} \notin H(\boldsymbol{\alpha}, c) \mid \bigwedge_{j \geq 1} \langle \boldsymbol{\alpha}, N^j \boldsymbol{x} \rangle \leq 0\},$$

$$X_+(\boldsymbol{\alpha}, c) = \{\boldsymbol{x} \in H(\boldsymbol{\alpha}, c) \mid \bigwedge_{j \geq 1} \langle \boldsymbol{\alpha}, N^j \boldsymbol{x} \rangle \geq 0\}.$$

Proposition 5.2. *(a) $X_-(\boldsymbol{\alpha}, c)$ is an M-attractor for every $\boldsymbol{x} \in E_-(\boldsymbol{\alpha}, c)$, and (b) $X_+(\boldsymbol{\alpha}, c)$ is an M-attractor for every $\boldsymbol{x} \in E_+(\boldsymbol{\alpha}, c)$.*

Proof. We only prove (a) since (b) is symmetrical.

We first show that $X_-(\boldsymbol{\alpha}, c)$ is an M-invariant. Consider $\boldsymbol{x} \in X_-(\boldsymbol{\alpha}, c)$. Since $M = I_n + N$, we have $\langle \boldsymbol{\alpha}, M\boldsymbol{x} \rangle = \langle \boldsymbol{\alpha}, \boldsymbol{x} \rangle + \langle \boldsymbol{\alpha}, N\boldsymbol{x} \rangle$. From $\boldsymbol{x} \notin H(\boldsymbol{\alpha}, c)$, we get $\langle \boldsymbol{\alpha}, \boldsymbol{x} \rangle < c$ and since $\boldsymbol{x} \in X_-(\boldsymbol{\alpha}, c)$, we deduce $\langle \boldsymbol{\alpha}, N\boldsymbol{x} \rangle \leq 0$. Therefore $\langle \boldsymbol{\alpha}, M\boldsymbol{x} \rangle < c$ and we have proved that $M\boldsymbol{x} \notin H(\boldsymbol{\alpha}, c)$. Moreover, given $j \geq 1$, observe that $\langle \boldsymbol{\alpha}, N^j M\boldsymbol{x} \rangle = \langle \boldsymbol{\alpha}, N^j \boldsymbol{x} \rangle + \langle \boldsymbol{\alpha}, N^{j+1} \boldsymbol{x} \rangle$. From $\boldsymbol{x} \in X_-(\boldsymbol{\alpha}, c)$ we get $\langle \boldsymbol{\alpha}, N^j \boldsymbol{x} \rangle \leq 0$ and $\langle \boldsymbol{\alpha}, N^{j+1} \boldsymbol{x} \rangle \leq 0$. We deduce that $\langle \boldsymbol{\alpha}, N^j M\boldsymbol{x} \rangle \leq 0$ for any $j \geq 1$. We have proved that $X_-(\boldsymbol{\alpha}, c)$ is an M-invariant.

It remains to show that for $\boldsymbol{x} \in E_-(\boldsymbol{\alpha}, c)$, there exists ℓ such that $M^\ell \boldsymbol{x} \in X_-(\boldsymbol{\alpha}, c)$. We use the expression (2) of $E_-(\boldsymbol{\alpha}, c)$. The case $\boldsymbol{x} \in Z_0(\boldsymbol{\alpha}) \setminus H(\boldsymbol{\alpha}, c)$ is immediate

since it implies $\boldsymbol{x} \in X_{-}(\boldsymbol{\alpha}, c)$. Thus, we can assume that there exists $m \in \mathbb{N} \setminus \{0\}$ such that $\boldsymbol{x} \in Z_{-m}(\boldsymbol{\alpha})$. By Lemma 5.1, there exists ℓ_0 such that $\langle \boldsymbol{\alpha}, M^{\ell}\boldsymbol{x}\rangle < c$ for any $\ell \geq \ell_0$. Let $j \geq 1$ and let us prove that there exists $\ell_j \in \mathbb{N}$ such that $\langle \boldsymbol{\alpha}, N^j M^{\ell}\boldsymbol{x}\rangle \leq 0$ for any $\ell \geq \ell_j$. Since M and N commute, we deduce that $\langle \boldsymbol{\alpha}, N^j M^{\ell}\boldsymbol{x}\rangle = \langle \boldsymbol{\alpha}, M^{\ell} N^j \boldsymbol{x}\rangle$. From equation (1) we get:

$$\langle \boldsymbol{\alpha}, M^{\ell} N^j \boldsymbol{x}\rangle = \sum_{i=0}^{n-1} L_i(\ell) \langle \boldsymbol{\alpha}, N^{i+j}\boldsymbol{x}\rangle$$

Thus $\ell \mapsto \langle \boldsymbol{\alpha}, N^j M^{\ell}\boldsymbol{x}\rangle$ is a polynomial in ℓ. If this polynomial is equal to zero then $\langle \boldsymbol{\alpha}, N^j M^{\ell}\boldsymbol{x}\rangle \leq 0$ for any $\ell \geq 0$. Otherwise, we get $j \leq m$ by definition of $Z_{-m}(\boldsymbol{\alpha})$, and the leading coefficient of this polynomial is equal to $\frac{\langle \boldsymbol{\alpha}, N^m \boldsymbol{x}\rangle}{!(m-j)}$. Now $\langle \boldsymbol{\alpha}, N^m\boldsymbol{x}\rangle < 0$ again by definition of $Z_{-m}(\boldsymbol{\alpha})$, and we deduce that $\lim_{\ell \to +\infty} \langle \boldsymbol{\alpha}, N^j M^{\ell}\boldsymbol{x}\rangle = -\infty$. Therefore there exists $\ell_j \in \mathbb{N}$ such that $\langle \boldsymbol{\alpha}, N^j M^{\ell}\boldsymbol{x}\rangle \leq 0$ for all $\ell \geq \ell_j$. Now, just observe that $M^{\ell}\boldsymbol{x} \in X_{-}(\boldsymbol{\alpha}, c)$ if $\ell = \max\{\ell_0, \ldots, \ell_{n-1}\}$. □

6 Computing Presburger Accelerated Interpolants

This section focus on the computation of a Presburger r-separator for a pair (E, F) of r-separable Presburger sets. Observe that this is equivalent to the Presburger accelerated interpolation problem for a spurious error-pattern with a unique cycle. We assume that the relation r satisfies $\boldsymbol{x}\, r\, \boldsymbol{y}$ iff $\boldsymbol{y} = M\boldsymbol{x} + \boldsymbol{v}$ where $\boldsymbol{v} \in \mathbb{Z}^n$ and $M \in \mathcal{M}_n(\mathbb{Z})$ is a poly-bounded matrix. Note that this condition strictly extend the finite monoid M^* condition required in acceleration techniques (see Theorem 4.1). We prove that if (E, F) is r-separable, then there exists a constructible Presburger r-separator for (E, F).

Remark 6.1. The unique cycle restriction is motivated by Example 4.2. In fact, this example exhibits a spurious error-pattern $(t_{ini}, t_1, t, t_2, t_{err})$ such that the cycles t_1 and t_2 satisfy the condition presented above. However, let us recall that this error-pattern does not admit a Presburger accelerated interpolant.

In the sequel, the Presburger sets E and F are decomposed into sets (E_i, F_j) following the half-space attractors introduced in the previous section. Note that a Presburger r-separator for (E, F) can be obtained as a combination of the Presburger r-separators for (E_i, F_j) thanks to the following straightforward Lemma 6.2.

Lemma 6.2 (Stability by union).

(a) *If X_i r-separates (E_i, F) for $1 \leq i \leq p$, then* $\bigcup_{i=1}^{p} X_i$ *r-separates* $(\bigcup_{i=1}^{p} E_i, F)$.

(b) *If X_j r-separates (E, F_j) for $1 \leq j \leq m$, then* $\bigcap_{j=1}^{m} X_j$ *r-separates* $(E, \bigcup_{j=1}^{m} F_j)$.

Now, we reduce the r-separability problem to the uniform case $\boldsymbol{v} = \mathbf{0}$. As expected, this reduction is obtained by adding an extra component that remains equal to 1. More precisely, consider the pair (E', F') of Presburger sets defined by $E' = E \times \{1\}$ and $F' = F \times \{1\}$ and the binary relation r' over $\mathbb{Z}^{n+1}$ defined by $((\boldsymbol{x}, x_{n+1}), (\boldsymbol{y}, y_{n+1})) \in r'$ iff $\boldsymbol{y} = M\boldsymbol{x} + \boldsymbol{v}x_{n+1}$ and $y_{n+1} = x_{n+1}$. Note that the matrix $M' = [[M\ \boldsymbol{v}][0, 1]] \in$

$\mathcal{M}_{n+1}(\mathbb{Z})$ is poly-bounded. Moreover (E, F) is r-separable if and only if (E', F') is r'-separable. From a Presburger r'-separator X' of (E', F') we deduce a Presburger r-separator for (E, F) by considering $X = \{\boldsymbol{x} \in \mathbb{Z}^n \mid (\boldsymbol{x}, 1) \in X'\}$. Note that under the condition $\boldsymbol{v} = \mathbf{0}$, a pair (E, F) of sets is r-separable if and only if $M^*E \cap F = \varnothing$ and a set X is a r-separator if and only if X is an M-invariant such that $E \subseteq X$ and $X \cap F = \varnothing$. Such a pair (E, F) is said *M-separable* and X is called a *M-separator*.

Next, the M-separability problem is reduced to a poly-bounded matrix $M = I_n + N$ where $N \in \mathcal{M}_n(\mathbb{Z})$ is a nilpotent matrix.

Lemma 6.3. *Let $M \in \mathcal{M}_n(\mathbb{Z})$ be a poly-bounded matrix. Let (D, N) be the Dunford decomposition of M. There exists an integer $d \in \mathbb{N} \setminus \{0\}$ such that the matrix $D' = D^d$ satisfies $D'D' = D'$. In this case $N' = M^d - D'$ is a nilpotent matrix of $\mathcal{M}_n(\mathbb{Z})$ and $M' = I_n + N'$ satisfies $M^{d\ell}M^{dn} = (M')^\ell M^{dn}$ for any ℓ.*

A pair (E, F) is M-separable if and only if (E', F') with $E' = \bigcup_{\ell=0}^{d-1} M^{dn+\ell}E$ and $F' = F$ is M'-separable. Moreover, given an M'-separator X' for (E', F'), the following set X is an M-separator for (E, F).

$$X = E \cup \ldots \cup M^{dn-1}E \cup (\bigcap_{\ell=0}^{d-1} \{\boldsymbol{x} \in M^{dn}\mathbb{Z}^n \mid M^\ell \boldsymbol{x} \in X'\})$$

Finally, denoting by $b > 0$ an integer extracted from the modular constraints defining the Presburger set F, the following lemma shows that by replacing $(I_n + N)$ by one of its powers $I_n + N' = (I_n + N)^d$, we can assume that $M \equiv_b I_n$.

Lemma 6.4. *For any matrix $M \in \mathcal{M}_n(\mathbb{Z})$ such that $M = I_n + N$ and for any integer $d > 0$ we have $M^d = I_n + N'$ where N' is a nilpotent matrix. Moreover, for any integer $b > 0$ there exists an integer $d > 0$ such that $M^d \equiv_b I_n$.*

A pair (E, F) is M-separable if and only if the pair (E', F') with $E' = \bigcup_{\ell=0}^{d-1} M^\ell E$ and $F' = F$ is M^d-separable. Moreover, given an M^d-separator for (E', F'), the following set X is an M-separator for (E, F).

$$X = \bigcap_{\ell=0}^{d-1} \{\boldsymbol{x} \in \mathbb{Z}^n \mid M^\ell \boldsymbol{x} \in X'\}$$

We can now provide the proof of our main Presburger separability theorem.

Theorem 6.5. *Let r be a binary relation over $\mathbb{Z}^n$ such that $\boldsymbol{x}\ r\ \boldsymbol{y}$ iff $\boldsymbol{y} = M\boldsymbol{x} + \boldsymbol{v}$ where $\boldsymbol{v} \in \mathbb{Z}^n$ and $M \in \mathcal{M}_n(\mathbb{Z})$ is poly-bounded. A pair (E, F) of Presburger sets, with either E or F finite, is r-separable if and only if it is Presburger r-separable. Moreover in this case we can effectively compute a Presburger r-separator.*

Proof. We have previously provided the reduction to the uniform case $\boldsymbol{v} = 0$. Let (E, F) be a pair of r-separable Presburger sets. Recall that this condition is equivalent to $M^*E \cap F = \varnothing$. From the reduction given in Lemma 6.3, we can assume that $M = (I_n + N)$ where $N \in \mathcal{M}_n(\mathbb{Z})$ is nilpotent. We have to find a Presburger r-separator X for (E, F) *i.e.*, an M-invariant X such that $E \subseteq X$ and $X \cap F = \varnothing$.

Since the condition $M^*E \cap F = \varnothing$ is equivalent to $(M^{-1})^*F \cap E = \varnothing$, and since by hypothesis, either E or F is finite, it suffices by symmetry to handle the case where E is finite. Since F is a Presburger set, it is defined by a propositional formula of linear

constraints, and one can effectively compute an integer $b \in \mathbb{N} \setminus \{0\}$ and an expression $F = \bigcup_{j=1}^{m}(C_j \cap \bigcap_{i=1}^{q_j} H(\boldsymbol{\alpha}_{i,j}, c_{i,j}))$, where for all $\boldsymbol{x} \in C_j$ and $\boldsymbol{y} \in \mathbb{Z}^n$, $\boldsymbol{x} \equiv_b \boldsymbol{y}$ implies $\boldsymbol{y} \in C_j$. By the reduction given in Lemma 6.4 one can assume that $M\boldsymbol{x} \equiv_b \boldsymbol{x}$ for all $\boldsymbol{x} \in \mathbb{Z}^n$. Notice that this implies that both C_j and $\mathbb{Z}^n \setminus C_j$ are M-invariant. By Lemma 6.2 (b), one can assume without loss of generality that F is of the form $C \cap \bigcap_{i=1}^{q} H(\boldsymbol{\alpha}_i, c_i)$.

Let $\boldsymbol{x} \in E$. Assume that $\boldsymbol{x} \in \bigcap_{i=1}^{q} E_+(\boldsymbol{\alpha}_i, c_i) \cap C$. Then $M^*\boldsymbol{x} \cap X_+(\boldsymbol{\alpha}_i, c_i) \neq \varnothing$ for $1 \leq i \leq q$ by Proposition 5.2(b). Since $X_+(\boldsymbol{\alpha}_i, c_i)$ is M-invariant, one would have $M^*\boldsymbol{x} \cap \bigcap_{i=1}^{r} X_+(\boldsymbol{\alpha}_i, c_i) \neq \varnothing$. Since $X_+(\boldsymbol{\alpha}_i, c_i) \subseteq H(\boldsymbol{\alpha}_i, c_i)$, and since $\boldsymbol{x}$ also belongs to C which is M-invariant, one would finally get $M^*\boldsymbol{x} \cap F \neq \varnothing$, contradicting the hypothesis $M^*E \cap F = \varnothing$. Therefore, $E \subseteq \bigcup_{i=1}^{q} E_-(\boldsymbol{\alpha}_i, c_i) \cup (\mathbb{Z}^n \setminus C)$, so that

$$E = \left[\bigcup_{i=1}^{q} E_-(\boldsymbol{\alpha}_i, c_i) \cap C \cap E\right] \cup [(\mathbb{Z}^n \setminus C) \cap E] \tag{4}$$

Again by Lemma 6.2 (a), it suffices to treat two cases

$$(a) \quad E \subseteq E_-(\boldsymbol{\alpha}_i, c_i) \cap C, \text{ and} \qquad (b) \quad E \subseteq \mathbb{Z}^n \setminus C.$$

In case (a), Proposition 5.2 shows that for every $\boldsymbol{x} \in E$ there exists ℓ such that $M^\ell \boldsymbol{x} \in X_-(\boldsymbol{\alpha}_i, c_i)$. Since E is finite and $X_-(\boldsymbol{\alpha}_i, c_i)$ is an invariant, there exists ℓ such that $M^\ell E \subseteq X_-(\boldsymbol{\alpha}_i, c_i)$. Furthermore, one can compute such an integer ℓ, just by computing successive images of E by M. Therefore, $X = \{M^k E \mid k \leq \ell\} \cup X_-(\boldsymbol{\alpha}_i, c_i)$ is an M-separator for (E, F).

In case (b), where $E \subseteq \mathbb{Z}^n \setminus C$, it suffices to choose $X = \mathbb{Z}^n \setminus C$, which is a Presburger M-invariant set such that $E \subseteq X$ and $X \cap F = \varnothing$. □

We finally prove that finiteness of either E or F is necessary to entail r-separability.

Proposition 6.6. *Consider $E = (1,1)\mathbb{N}$ and $F = \{\boldsymbol{x} \in \mathbb{Z}^2 \mid x_2 < x_1 \wedge x_1 < 0\}$. Let $r \subseteq \mathbb{Z}^2 \times \mathbb{Z}^2$ defined by $\boldsymbol{x}\ r\ \boldsymbol{y}$ if $y_1 = x_1 + x_2 - 2$ and $y_2 = x_2 - 2$. Then, the pair (E, F) is r-separable, but it is not Presburger r-separable.*

Proof. Computing $r^\ell E = -(\ell(\ell+1), 2\ell) + (\ell+1, 1)\mathbb{N}$ shows that $r^*E \cap F = \varnothing$, whence r^*E is an r-separator for (E, F). Assume by contradiction that there is a Presburger r-separator X for (E, F). For $t \in \mathbb{Z}$ and $t \geq -1$, let $D_t = (t, t) + (t, -1)\mathbb{N}$. This linear set is located on the line $(\Delta_t) : x_1 + tx_2 = t + t^2$. Figure 6 (a) depicts the set E, its successive images under r, and F. Figure 6 (b) displays the sets D_t and the lines Δ_t. An easy computation gives $(t, t) + k(t, -1) = r^k(t+k, t+k) \in r^*E$, so $\bigcup_{t \geq -1} D_t \subseteq r^*E$.

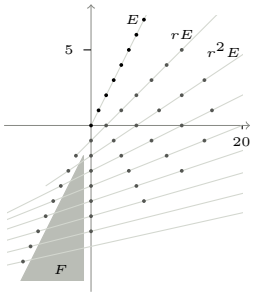
Let $R_t \subseteq \mathbb{Z}^2$ be the set of points between Δ_{t-1} and Δ_t in the half space $x_1 \geq x_2$.

$$R_t = \big\{(x_1, x_2) \mid x_1 + tx_2 < t + t^2, \quad x_1 + (t-1)x_2 > (t-1) + (t-1)^2, \text{ and } x_1 \geq x_2\big\}.$$

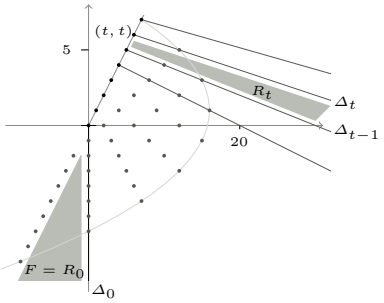
This is a Presburger set, and $F = R_0$. One easily checks that $rR_t \subseteq R_{t-1}$. We claim that there exists t such that $\varnothing \neq R_t \cap X$. This will yield the contradiction, since then $\varnothing \neq r(R_t \cap X) \subseteq r(R_t) \cap r(X) \subseteq R_{t-1} \cap X$, and by induction, $\varnothing \neq R_0 \cap X = F \cap X$, contradicting the assumption that X is an r-separator for (E, F).

Choose an expression of the Presburger set X as a finite union of linear sets, and let $N \in \mathbb{N}$ be greater than all the norms of the periods appearing in this expression. Then,

every point of X but a finite number is at distance at most N of another element of X. Choose $\boldsymbol{x} \in D_N \subseteq X$, with $\boldsymbol{x}_1$ large enough so that the distance from $\boldsymbol{x}$ to both D_{N-1} and D_{N+1} is greater than N. There are infinitely many such $\boldsymbol{x}$, since D_N is neither parallel to D_{N-1} nor to D_{N+1}. Now, any two points of $\Delta_N \cap \mathbb{Z}^2$ are at distance at least N. By the choice of $\boldsymbol{x}$ and the definition of N, there must be an element in $X \cap R_N$ or $X \cap R_{N+1}$. This proves the claim and concludes the proof of the proposition. □



(a) r^*E as an infinite union of $r^\ell E$



(b) r^*E as an infinite union of D_t

7 Conclusion & Further Work

The main idea of this paper is to combine *interpolation-based model-checking*, which works well on large and simple systems, and *acceleration techniques*, which prefers small and complex ones. We explored a track to combine them, named *interpolant acceleration*. in which we see a fair trade-off between the lack of scalability of acceleration, by applying it locally, and CEGAR inability to deal with of infinite behaviors. We also strongly believe this paper to open a new field of investigation, and to offer interesting research perspectives for future work.

We introduced the notion of *error-pattern* and *accelerated interpolant*. We identified two classes of computable accelerated interpolants: 'Presburger' accelerated interpolants and 'poly-bounded' accelerated interpolants. The second one allows to compute interpolants for error-patterns labeled by transformations which strictly enlarge usual classes used in acceleration techniques. This method is applicable for programs with a finite set of initializations or with a finite set of errors, and this condition is necessary due to Proposition 6.6. It would be interesting to extend the class of transformations, and to find finer conditions for such interpolants to be computable. One can extract straight from our constructive proof a rough algorithm. We would like to make it explicit, to compute its theoretical complexity, and to test how it behaves in practice.

Indeed, we would like to find efficient algorithms to compute accelerated interpolants as the one we provide here through the proof is brute-force. One possible track is to compute them from symbolic (*e.g.* automata based) set representations, and then build an effective implementation of a CEGAR loop using accelerated interpolants. Next, the full potential of accelerated interpolants in the refinement remains to be explored. From a more theoretical point of view, there are also many possible extensions: among others, we would like to be able to handle transitions with explicit guards, or check for some extensions of the class of transformations for which we can compute accelerated interpolants. A full study of these classes would allow us to clearly delimit

what is the frontier between programs that can be handled by accelerated interpolants and others. Finally, another track would be to investigate the influence of some structural properties of the CFA (*e.g.* nested cycles) and how to deal with spurious error-traces whose proof does not hold after some unrolling.

References

[1] S. Bardin, A. Finkel, J. Leroux, and P. Schnoebelen. Flat Acceleration in Symbolic Model-Checking. In *Proc. of 3rd Symp. on Automated Technology for Verification and Analysis (ATVA'05)*, volume 3707 of *LNCS*, pages 474–488. Springer, 2005.

[2] D. Beyer, T. A. Henzinger, R. Majumdar, and A. Rybalchenko. Path Invariants. In *Proc. of the ACM SIGPLAN'07 Conference on Programming Language Design and Implementation (PLDI'07)*, pages 300–309. ACM Press, 2007.

[3] B. Boigelot. *Symbolic Methods for Exploring Infinite State Spaces*. PhD thesis, Faculté des Sciences Appliquées de l'Université de Liège, 1999.

[4] B. Boigelot. On Iterating Linear Transformations Over Recognizable Sets of Integers. *Theoret. Comput. Sci.*, 309(1-3):413–468, 2003.

[5] A. Bouajjani, B. Jonsson, M. Nilsson, and T. Touili. Regular Model-Checking. In *Proc. of 12th Conf. on Computer Aided Verification (CAV'00)*, volume 1855 of *LNCS*, pages 403–418, 2000.

[6] E. M. Clarke, O. Grumberg, S. Jha, Y. Lu, and H. Veith. CounterExample-Guided Abstraction Refinement for Symbolic Model Checking. *J. ACM*, 50(5):752–794, 2003.

[7] J. Esparza, S. Kiefer, and S. Schwoon. Abstraction Refinement with Craig Interpolation and Symbolic Pushdown Systems. In *Proc. of 12th Conf. on Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2006)*, volume 3920 of *LNCS*, pages 489–503. Springer, 2006.

[8] A. Finkel and J. Leroux. How to Compose Presburger-Accelerations: Applications to Broadcast Protocols. In *Proc. of Conf. on Foundation of Software Technology and Theoretical Computer Science (FSTTCS'02)*, volume 2556, pages 145–156. Springer, 2002.

[9] S. Ginsburg and E. H. Spanier. Semigroups, Presburger Formulas and Languages. *Pacific Journal of Mathematics*, 16(2):285–296, 1966.

[10] S. Graf and H. Saïdi. Construction of Abstract State Graphs with PVS. In *Proc. of 9th Conf. on Computer Aided Verification (CAV'97)*, volume 1254 of *LNCS*, pages 72–83, 1997.

[11] B. Gulavani, T. A. Henzinger, Y. Kannan, A. Nori, and S. K. Rajamani. Synergy: A New Algorithm for Property Checking. In *Proc. of 14th Symp. on Foundations of Software Engineering (FSE'06)*, pages 117–127. ACM Press, 2006.

[12] T. A. Henzinger, R. Jhala, R. Majumbar, and G. Sutre. Lazy Abstraction. In *Proc. of 29th Symp. on Principles of Programming Languages (POPL'02)*, pages 58–70, 2002.

[13] R. Jhala and K. L. McMillan. A Practical and Complete Approach to Predicate Refinement. In *Proc. of 12th Conf. on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'06)*, volume 3920 of *LNCS*, pages 459–473. Springer, 2006.

[14] K. L. McMillan. Interpolation and SAT-Based Model Checking. In *Proc. of 15th Conf. on Computer Aided Verification (CAV'03)*, volume 2725 of *LNCS*, pages 1–13. Springer, 2003.

[15] K. L. McMillan. An Interpolating Theorem Prover. *Journal of Theoritical Computer Science*, 345(1):101–121, 2005.

[16] K. L. McMillan. Lazy Abstraction with Interpolants. In *Proc. of 18th Conf. on Computer Aided Verification (CAV'06)*, volume 4144 of *LNCS*, pages 123–136. Springer, 2006.

[17] M. Presburger. Über die Vollständigkeit eines gewissen Systems der Arithmetik ganzer Zahlen, in welchem die Addition als einzige Operation hervortritt. In *Comptes Rendus du 1er congrès de Mathématiciens des Pays Slaves*, pages 92–101, 1929.

A The algorithm to compute Presburger separators

Let $F = \bigcup_{j=1}^{m} F_j$, with $F_j = \bigcap_{i=1}^{q_j} H(\boldsymbol{\alpha}_{i,j}, c_{i,j}) \cap C_j$, where C_j denotes the modular constraints of F_j. The algorithm obtained from the constructive proof is written in Fig. 5. Its purpose is to highlight the structure of the proof, not to provide an implementable algorithm. The functions (a) and (b) are justified by Lemma 6.3. The first one reduces the matrix to one of the form $I_n + N$, where N is nilpotent. The second one makes sure that M acts as the identity with respect to modular constraints appearing in F. The functions SEPARATE3 and SEPARATE4 correspond to reductions made in Theorem 6.5.

(a) **Reduction to $M - I_n$ nilpotent**

SEPARATE (M, E, F):
  $D, N \leftarrow$ DUNFORD(M)
  Compute d such that $D^d = D^{2d}$
  $E' \leftarrow \bigcup_{\ell=0}^{d-1} M^\ell E$
  $M' \leftarrow I_n + M^d - D^d$
  $X' \leftarrow$ SEPARATE2(M', E', F)
  Return $\bigcup_{\ell=0}^{dn-1} M^\ell E \cup$
    $\bigcap_{\ell=0}^{d-1} \{\boldsymbol{x} \in M^{dn}\mathbb{Z}^n \mid M^\ell \boldsymbol{x} \in X'\}$

(b) **Reduction for stability wrt, mod-b constraints**

SEPARATE2 (M, E, F):
  $b \leftarrow \operatorname{lcm}_{1 \le i \le m}$ (modular constraints of C_i)
  $d \leftarrow b^{\lfloor \log n + 1 \rfloor}$
  $E' \leftarrow \bigcup_{\ell=0}^{d-1} \mathsf{post}_{M^\ell}(E)$
  $X =$SEPARATE3 (M^d, E', F)
  Return $\bigcap_{\ell=0}^{d-1} \mathsf{wpre}_{M^\ell}(X)$

(c) **Eliminating unions and mod-b constraints from F**

SEPARATE3 (M, E, F):
  Return $\bigcap_{j=1}^{m}$ SEPARATE4$(M, E, F_j) \cup (E \setminus C_j)$

(d) **Reduction on E**

SEPARATE4 (M, E, F):
  // Assume F of the form $\bigcup_{i=1}^{q} H(\boldsymbol{\alpha}_i, c_i)$
  // $M - I_n$ nilpotent, and $M \equiv_b I_n$.

  for i in $1, \ldots, q$:
    $E_i = E \cap E_-(\boldsymbol{\alpha}_i, c_i)$
    Compute ℓ_i s.t. $M^{\ell_i} E_i \subseteq X_-(\boldsymbol{\alpha}_i, c_i)$
    $X'_i = \bigcup_{\ell=0}^{\ell_i} M^\ell E \quad \cup X_-(\boldsymbol{\alpha}_i, c_i)$
  Return $\bigcup X'_i$

Fig. 5. Algorithm for computing a Presburger separator

B An Example of Non-uniform Attractor

Example B.1. Note that even if X is an M-attractor for every $\boldsymbol{x} \in E$, there might exist no ℓ such that $M^\ell E \subseteq X$. For instance, let $X = \{(1,0)\} \cup]-\infty,-1] \times]-\infty,0]$, $E = \{1\} \times [-1,0]$ and $M = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$. First, $M(1,0) = (1,0) \in X$. Next, for $\lambda \in [-1,0[$. We have $M^\ell = \begin{pmatrix} 1 & \ell \\ 0 & 1 \end{pmatrix}$ and thus we have $M^\ell(1,\lambda) = (1+\ell\lambda, \lambda)$ which also belongs to X for ℓ large enough. On the other hand, $M^\ell E$ contains $M^\ell(1,-1/\ell) = (0,-1/\ell)$ which does not belong to X.

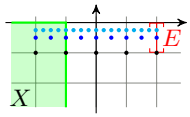


Fig. 6. X is not an M-attractor for E

C Proof of Proposition 3.2

Remember that a matrix $M \in \mathcal{M}_n(\mathbb{Z})$ is said *poly-bounded* if all the coefficients of M^ℓ are polynomially bounded in ℓ.

Proposition 3.1. *A matrix $M \in \mathcal{M}_n(\mathbb{Z})$ is poly-bounded if and only if the Dunford decomposition (D, N) of M is such that D^* is finite.*

Proof. Assume first that D^* is finite. The binomial theorem applied to the commuting matrices D and N proves that the coefficients of M^ℓ are polynomials in ℓ, just by observing that $N^n = 0_n$.

Conversely, assume that the coefficients of M^ℓ are polynomially bounded in ℓ. The binomial theorem applied to M and $-N$ shows that the coefficients of D^ℓ are polynomially bounded in ℓ. Since D is diagonalizable, there exists a diagonal matrix $\Delta \in \mathcal{M}_n(\mathbb{C})$ and an invertible matrix $P \in \mathcal{M}_n(\mathbb{C})$ such that $D = P^{-1}\Delta P$. Denoting by $\lambda_1, \ldots, \lambda_n \in \mathbb{C}$ the eigenvalues of D, we deduce that $\lambda_1^\ell, \ldots, \lambda_n^\ell$ are the diagonal entries of Δ^ℓ. From $\Delta^\ell = PD^\ell P^{-1}$, we deduce that $|\lambda_i|^\ell$ is polynomially bounded in ℓ. Thus $|\lambda_i| \leq 1$. From $D^\ell = P^{-1}\Delta^\ell P$, we deduce that the coefficients of D^ℓ are bounded, independently of ℓ.

Choose an integer $k > 0$ large enough so that $kN^i \in \mathcal{M}_n(\mathbb{Z})$ for all $0 \leq i \leq n-1$. Observe that the binomial theorem applied to M and $-N$ also provides $kD^\ell \in \mathcal{M}_n(\mathbb{Z})$ for every $\ell \in \mathbb{N}$ (recall that $L_i(\ell) \in \mathbb{N}$). Moreover, as $k.D^\ell$ is bounded, we deduce that $\{kD^\ell \mid \ell \in \mathbb{N}\}$ is included in a finite set of matrices. Thus D^* is finite. □

D Proof of Lemma 5.1

Lemma 5.1. *Let $\boldsymbol{\alpha}, \boldsymbol{x} \in \mathbb{R}^n$. We have:*

$$\lim_{\ell \to +\infty} \langle \boldsymbol{\alpha}, M^\ell \boldsymbol{x} \rangle = \begin{cases} +\infty & \text{if } \boldsymbol{x} \in \bigcup_{k \geq 1} Z_k(\boldsymbol{\alpha}), \\ \langle \boldsymbol{\alpha}, \boldsymbol{x} \rangle & \text{if } \boldsymbol{x} \in Z_0(\boldsymbol{\alpha}), \\ -\infty & \text{if } \boldsymbol{x} \in \bigcup_{k \leq -1} Z_k(\boldsymbol{\alpha}). \end{cases}$$

Proof. Let $\boldsymbol{x} \in Z_{\varepsilon.m}(\boldsymbol{\alpha})$ where $\varepsilon \in \{-1, 1\}$ and $m \in \mathbb{N} \setminus \{0\}$. Equation (1) shows that $\ell \mapsto \langle \boldsymbol{\alpha}, M^\ell \boldsymbol{x} \rangle$ is a non-constant polynomial in ℓ whose leading coefficient is $\frac{1}{!m} \langle \boldsymbol{\alpha}, N^m \boldsymbol{x} \rangle$. Since $\boldsymbol{x} \in Z_{\varepsilon.m}(\boldsymbol{\alpha})$, we have $\varepsilon \langle \boldsymbol{\alpha}, N^m \boldsymbol{x} \rangle > 0$, which implies the result if $\boldsymbol{x} \in \bigcup_{k \neq 0} Z_k(\boldsymbol{\alpha})$. Finally, if $\boldsymbol{x} \in Z_0(\boldsymbol{\alpha})$, then $\langle \boldsymbol{\alpha}, M^\ell \boldsymbol{x} \rangle$ is constant, equal to $\langle \boldsymbol{\alpha}, \boldsymbol{x} \rangle$.

□

E Proofs of Lemmas 6.3 and 6.4

Let us first provide the following technical lemma that will be useful when replacing X by a nilpotent matrix $N \in \mathcal{M}_n(\mathbb{Z})$.

Lemma E.1. *Let $b \in \mathbb{N} \setminus \{0\}$. Then, for any $k \in \mathbb{N}$, there exist $P(X) \in \mathbb{Z}/b\mathbb{Z}[X]$ such that, in $\mathbb{Z}/b\mathbb{Z}[X]$:*

$$(1+X)^{b^k} = 1 + X^{2^k} P(X). \tag{5}$$

Proof. By induction on k. For $k = 0$, take $P(X) = 1$. Assume now that (5) holds. Then we have, in $\mathbb{Z}/b\mathbb{Z}[X]$:

$$(1+X)^{b^{k+1}} = \left(1 + X^{2^k} P(X)\right)^b = \sum_{i=0}^{b} C_b^i . \left(X^{2^k} P(X)\right)^i = 1 + X^{2^{k+1}} Q(X)$$

with $Q(X) = \sum_{i=2}^{b} C_b^i . \left[X^{2^k}\right]^{i-2} P(X)^i$. We have obtained (5) at rank $k+1$. □

Corollary E.2. *Let $M \in \mathcal{M}_n(\mathbb{Z})$ be a matrix such that its Dunford decomposition (D, N) satisfies $D^2 = D$. Then (D, N) is a pair of matrices in $\mathcal{M}_n(\mathbb{Z})$.*

Proof. As $D, N \in \mathcal{M}_n(\mathbb{Q})$, there exists an integer $d \geq 2$ large enough so that $dDN^\ell \in \mathcal{M}_n(\mathbb{Z})$ for every $0 \leq \ell \leq n-1$. As $N^n = 0_n$ we deduce that $dDN^\ell \in \mathcal{M}_n(\mathbb{Z})$ for any ℓ. Let us consider $k \in \mathbb{N}$ such that $2^k \geq n$ and let $m = d^k$. In particular $m \geq n$. Lemma E.1 proves that the there exists a polynomial $Q(X) \in d\mathbb{Z}[X]$ and a polynomial $P(X) \in \mathbb{Z}[X]$ such that $(1+X)^m = 1 + Q(X) + X^{2^k} P(X)$. Replacing X by N provides $(I_n + N)^m = I_n + Q(N)$. Thus $D(I_n + N)^m = D + DQ(N)$. Note that $DQ(N) \in \mathcal{M}_n(\mathbb{Z})$ since $dDN^\ell \in \mathcal{M}_n(\mathbb{Z})$ for any ℓ. The binomial theorem applied to the commuting matrices D and N provides $(D+N)^m = \sum_{i=0}^{n-1} L_i(m) D^{m-i} N^i$. As $D^2 = D$ we deduce that $M^m = D(\sum_{i=0}^{n-1} L_i(m) N^i)$. The binomial theorem applied to I_n and N also provides $(I_n + N)^m = (\sum_{i=0}^{n-1} L_i(m) N^i)$. We deduce that $M^m = D(I_n + N)^m$. We have proved that $M^m = D(I_n + Q(N))$. Since M^m and $DQ(N)$ are both in $\mathcal{M}_n(\mathbb{Z})$ we deduce that $D \in \mathcal{M}_n(\mathbb{Z})$. From $N = M - D$ we also obtain $N \in \mathcal{M}_n(\mathbb{Z})$. □

Lemma 6.3. *Let $M \in \mathcal{M}_n(\mathbb{Z})$ be a poly-bounded matrix. Let (D, N) be the Dunford decomposition of M. There exists an integer $d \in \mathbb{N} \setminus \{0\}$ such that the matrix $D' = D^d$ satisfies $D'D' = D'$. In this case $N' = M^d - D'$ is a nilpotent matrix of $\mathcal{M}_n(\mathbb{Z})$ and $M' = I_n + N'$ satisfies $M^{d\ell} M^{dn} = (M')^\ell M^{dn}$ for any ℓ.*

A pair (E, F) is M-separable if and only if the pair (E', F') with $E' = \bigcup_{\ell=0}^{d-1} M^{dn+\ell} E$ and $F' = F$ is M'-separable. Moreover, given an M'-separator X' for (E', F'), the following set X is an M-separator for (E, F).

$$X = E \cup \ldots \cup M^{dn-1} E \cup \left(\bigcap_{\ell=0}^{d-1} \{\boldsymbol{x} \in M^{dn}\mathbb{Z}^n \mid M^\ell \boldsymbol{x} \in X'\}\right)$$

Proof. As M is poly-bounded, we deduce that D^* is finite and in particular there exists an integer $d > 0$ such that $D^{2d} = D^d$. Let $D' = D^d$ and $N' = M^d - D'$. The binomial theorem applied to the commuting matrices M and D shows that N' is a nilpotent matrix. Observe that (D', N') is the Dunford decomposition of $M^d \in \mathcal{M}_n(\mathbb{Z})$. As $D'D' = D'$, Corollary E.2 proves that $N', D' \in \mathcal{M}_n(\mathbb{Z})$.

Let us show that $D'M^{dn} = M^{dn}$. The binomial theorem applied on the commuting matrices D' and N' provides $(D' + N')^n = \sum_{m=0}^{n-1} L_m(n)(D')^{n-m}(N')^m = D'(\sum_{m=0}^{n-1} L_m(n)(N')^m)$ thanks to $D'D' = D'$. We have proved that $D'(D'+N')^n = (D' + N')^n$ and thus $D'M^{dn} = M^{dn}$.

We now show that $M^{d\ell}M^{dn} = (M')^\ell M^{dn}$ for any ℓ. The binomial theorem applied to D' and N' provides $(D'+N')^\ell = \sum_{m=0}^{\ell} L_m(\ell)(N')^m(D')^{\ell-m}$. By multiplying by D', we get $(D'+N')^\ell D' = (\sum_{m=0}^{\ell} L_m(\ell)(N')^m)D'$. The binomial theorem applied to I_n and N' also provides $\sum_{m=0}^{\ell} L_m(\ell)(N')^m = (I_n + N')^\ell$. Combining this equality with the previous one provides $(D' + N')^\ell D' = (I_n + N')^\ell D'$. From the previous paragraph, we get $M^{d\ell}M^{dn} = (M')^\ell M^{dn}$.

Finally, let (E, F) be a pair of sets. From the previous equality we get $M^*E = (M')^*E'$. Thus (E, F) is M-separable if and only if (E', F') is M'-separable. Let us consider a M'-separator X' for (E', F') and let us prove that X is an M-separator for (E, F). First of all, note that $E \subseteq X$. Moreover, as $X \subseteq M^*E \cup X'$, $M^*E \cap F = \varnothing$ and $X' \cap F = \varnothing$ we get $X \cap F = \varnothing$. Now, let us show that X is an M-invariant. Note that $M^{dn+\ell}E \subseteq E' \subseteq X'$ for any $0 \leq \ell \leq d-1$ and thus $M^{dn}E \subseteq X$. Thus, it is sufficient to prove that $\bigcap_{\ell=0}^{d-1}\{\boldsymbol{x} \in M^{dn}\mathbb{Z}^n \mid M^\ell \boldsymbol{x} \in X'\}$ is an M-invariant. Let us consider $\boldsymbol{x} \in M^{dn}\mathbb{Z}^n$ such that $M^\ell \boldsymbol{x} \in X'$ for any $0 \leq \ell \leq d-1$ and let us prove that $M\boldsymbol{x}$ satisfies the same conditions. Observe that $M\boldsymbol{x} \in M^{dn}\mathbb{Z}^n$ and $M^\ell(M\boldsymbol{x}) \in X'$ for any $0 \leq \ell < d-1$. As $\boldsymbol{x} \in M^{dn}\mathbb{Z}^n$, there exists a vector $\boldsymbol{z} \in \mathbb{Z}^n$ such that $\boldsymbol{x} = M^{dn}\boldsymbol{z}$. As X' is an M'-invariant and $\boldsymbol{x} \in X'$ we deduce that $M'\boldsymbol{x} \in X'$. Thus $M'M^{dn}\boldsymbol{z} \in X'$. From the equality $M^{d\ell}M^{dn} = (M')^\ell M^{dn}$ established in the previous paragraph, we get $M'M^{dn}\boldsymbol{z} = M^dM^{dn}\boldsymbol{z} = M^d\boldsymbol{x}$. Thus $M^{d-1}(M\boldsymbol{x}) \in X'$ and we have proved that $M^\ell(M\boldsymbol{x}) \in X'$ for any $0 \leq \ell \leq d-1$. We deduce that X is an M-separator for (E, F). □

Lemma 6.4. *For any matrix $M \in \mathcal{M}_n(\mathbb{Z})$ such that $M = I_n + N$ and for any integer $d > 0$ we have $M^d = I_n + N'$ where N' is a nilpotent matrix. Moreover, for any integer $b > 0$ there exists an integer $d > 0$ such that $M^d \equiv_b I_n$.*

A pair (E, F) is M-separable if and only if the pair (E', F') with $E' = \bigcup_{\ell=0}^{d-1} M^\ell E$ and $F' = F$ is M^d-separable. Moreover, given an M^d-separator for (E', F'), the following set X is an M-separator for (E, F).

$$X = \bigcap_{\ell=0}^{d-1}\{\boldsymbol{x} \in \mathbb{Z}^n \mid M^\ell \boldsymbol{x} \in X'\}$$

Proof. The binomial theorem applied to I_n and N shows that $N' = M^d - I_n$ is a nilpotent matrix. Let us consider an integer $k \in \mathbb{N}$ such that $2^k \geq n$. Lemma E.1 shows that there exists a polynomial $Q(X) \in b\mathbb{Z}[X]$ and a polynomial $P(X) \in \mathbb{Z}[X]$ such

that $(1+X)^d = 1 + Q(X) + X^{2^k} P(X)$ with $d = b^k$. Replacing X by N provides $M^d = I_n + Q(N)$. As $Q(X) \in b\mathbb{Z}[X]$, we deduce that $M^d \equiv_b I_n$.

Let (E, F) be a pair of sets. As $M^*E = (M^d)^*E'$, we deduce that (E, F) is M-separable if and only if (E', F') is M^d-separable. Let us consider a M^d-separator X' for (E', F') and let us prove that X is an M-separator for (E, F). First of all, note that $E \subseteq X$ and from $X' \cap F = \varnothing$ and $X \subseteq X'$ we deduce that $X \cap F = \varnothing$. It sufficient to prove that X is an M-invariant. Let $\boldsymbol{x} \in X$. We deduce that $M^\ell \boldsymbol{x} \in X'$ for any $0 \leq \ell \leq d-1$. In particular $M^\ell(M\boldsymbol{x}) \in X'$ for any $0 \leq \ell < d-1$. As X' is an M^d-invariant and $\boldsymbol{x} \in X'$, we deduce that $M^d \boldsymbol{x} \in X'$. We have proved that $M^\ell(M\boldsymbol{x}) \in X'$ for any $0 \leq \ell \leq d-1$. Therefore $M\boldsymbol{x} \in X$ and we have proved that X is an M-invariant. □