Ramsey Quantifiers in Linear Arithmetics

PASCAL BERGSTRÄSSER, University of Kaiserslautern-Landau, Germany
MOSES GANARDI, MPI-SWS, Germany
ANTHONY W. LIN, University of Kaiserslautern-Landau, Germany and MPI-SWS, Germany
GEORG ZETZSCHE, MPI-SWS, Germany

We study Satisfiability Modulo Theories (SMT) enriched with the so-called Ramsey quantifiers, which assert the existence of cliques (complete graphs) in the graph induced by some formulas. The extended framework is known to have applications in proving program termination (in particular, whether a transitive binary predicate is well-founded), and monadic decomposability of SMT formulas. Our main result is a new algorithm for eliminating Ramsey quantifiers from three common SMT theories: Linear Integer Arithmetic (LIA), Linear Real Arithmetic (LRA), and Linear Integer Real Arithmetic (LIRA). In particular, if we work only with existentially quantified formulas, then our algorithm runs in polynomial time and produces a formula of linear size. One immediate consequence is that checking well-foundedness of a given formula in the aforementioned theory defining a transitive predicate can be straightforwardly handled by highly optimized SMT-solvers. We show also how this provides a uniform semi-algorithm for verifying termination and liveness with completeness guarantee (in fact, with an optimal computational complexity) for several well-known classes of infinite-state systems, which include succinct timed systems, one-counter systems, and monotonic counter systems. Another immediate consequence is a solution to an open problem on checking monadic decomposability of a given relation in quantifier-free fragments of LRA and LIRA, which is an important problem in automated reasoning and constraint databases. Our result immediately implies decidability of this problem with an optimal complexity (coNP-complete) and enables exploitation of SMT-solvers. It also provides a termination guarantee for the generic monadic decomposition algorithm of Veanes et al. for LIA, LRA, and LIRA. We report encouraging experimental results on a prototype implementation of our algorithms on micro-benchmarks.

CCS Concepts: • **Theory of computation** → **Logic and verification**; **Automated reasoning**; **Program verification**; *Complexity classes*; *Verification by model checking*; Program analysis.

Additional Key Words and Phrases: Ramsey Quantifiers, Satisfiability Modulo Theories, Linear Integer Arithmetic, Linear Real Arithmetic, Monadic Decomposability, Liveness, Termination, Infinite Chains, Infinite Cliques

ACM Reference Format:
Pascal Bergsträßer, Moses Ganardi, Anthony W. Lin, and Georg Zetzsche. 2024. Ramsey Quantifiers in Linear Arithmetics. *Proc. ACM Program. Lang.* 8, POPL, Article 1 (January 2024), 32 pages. https://doi.org/10.1145/3632843

1 INTRODUCTION

The last two decades have witnessed significant advances in software verification [Jhala and Majumdar 2009]. One prominent and fruitful approach to software verification is that of *deductive verification* and *program logics* [Leino 2023; Nelson and Oppen 1980; Shostak 1984], whereby one

Authors' addresses: Pascal Bergsträßer, University of Kaiserslautern-Landau, Kaiserslautern, Germany, bergstraesser@cs.uni-kl.de; Moses Ganardi, MPI-SWS, Kaiserslautern, Germany, ganardi@mpi-sws.org; Anthony W. Lin, University of Kaiserslautern-Landau, Kaiserslautern, Germany and MPI-SWS, Kaiserslautern, Germany, awlin@mpi-sws.org; Georg Zetzsche, MPI-SWS, Kaiserslautern, Germany, georg@mpi-sws.org.

ACM 2475-1421/2024/1-ART1
https://doi.org/10.1145/3632843

models a specification of a program P as a formula φ_P over some logical theory, usually in first-order logic (FO), or a fragment thereof (e.g. quantifier-free formulas or existential formulas). That way, the original problem is reduced to satisfiability of the formula φ_P (i.e. whether it has a solution). One decisive factor of the success of this software verification approach is that solvers for satisfiability of boolean formulas and extensions to quantifier-free and existential theories (a.k.a. SAT-solvers and SMT-solvers, respectively) have made an enormous stride forward in the last decades to the extent that they are now capable of solving practical industrial instances. The cornerstone theories in the SMT framework include the theory of Linear Integer Arithmetic (LIA), the theory of Linear Real Arithmetic (LRA), and the mixed theory of Linear Integer Real Arithmetic (LIRA). Among others, these can be used to naturally model numeric programs [Hague and Lin 2011], programs with clocks [Boigelot and Herbreteau 2006; Dang 2001; Dang et al. 2000, 2001; Hague and Lin 2011], linear hybrid systems [Boigelot and Herbreteau 2006], and numeric abstractions of programs that manipulate lists and arrays [Bouajjani et al. 2011; Hague and Lin 2011].

Most program logics in software verification can be formulated directly within FO. For example, to specify a *safety* property, a programmer may provide a formula *Inv* asserting a desired invariant for the program. In turn, the property that *Inv* is an invariant is definable in FO. In fact, if we stay within the quantifier-free fragment of FO, this check can be easily and efficiently verified by SMT-solvers. Some program logics, however, require us to go beyond FO. Most notably, when verifying that a program terminates, a programmer must provide well-founded relations (or a finite disjunction thereof) and prove that this covers the transitive closure of the program (e.g. see [Podelski and Rybalchenko 2004]). [Some techniques realize the proof rules of [Podelski and Rybalchenko 2004] (e.g. see [Cook et al. 2011]) by constructing relations that are guaranteed to be well-founded by construction, but these limit the shapes of the well-founded relations that can be constructed.] In case such a relation is synthesized and not guaranteed to be well-founded, one may want to check the well-foundedness property automatically. Since well-foundedness of a transitive predicate is in general not a first-order property (e.g. see Problem 1.4.1 of [Chang and Keisler 1990]), an extension of FO is required to be able to reason about well-foundedness. One solution is to simply enrich FO with an ad-hoc condition for checking well-foundedness of a relation [Beyene et al. 2013]. A more general solution is to extend FO with *Ramsey quantifiers* [Bergsträßer et al. 2022] (see also Chapter VII of [Barwise and Feferman 1985]) and study elimination of such quantifiers in the logical theory under consideration. This latter solution is known [Bergsträßer et al. 2022] to also provide an approach to analyze variable dependencies (a.k.a. *monadic decomposability*) in a first-order formula, which has applications in formal verification [Veanes et al. 2017] and query optimization in constraint databases [Grumbach et al. 2001; Kuper et al. 2000].

SMT with Ramsey quantifiers. In a nutshell, a Ramsey quantifier asserts the existence of an infinite sequence of elements forming a clique (i.e. a complete graph) in the graph induced by a given formula. [There are in fact two flavors of Ramsey quantifiers, of which one asserts the existence of an undirected clique (e.g. see Chapter VII of [Barwise and Feferman 1985]), and the other of a *directed* clique [Bergsträßer et al. 2022]. In the sequel, we will only deal with the latter because of the applications to reasoning about liveness and variable dependencies.] More precisely, if $\varphi(\boldsymbol{x},\boldsymbol{y})$ is a formula over a structure $\mathfrak{A}$ with universe D and $\boldsymbol{x}$, $\boldsymbol{y}$ are k-tuples of variables, the formula $\exists^{\mathsf{ram}}\boldsymbol{x},\boldsymbol{y}\colon \varphi(\boldsymbol{x},\boldsymbol{y})$ asserts the existence of an *infinite (directed) φ-clique*, i.e. a sequence $\boldsymbol{v}_1,\boldsymbol{v}_2,\ldots$ of pairwise distinct k-tuples in D^k such that $\mathfrak{A}\models\varphi(\boldsymbol{v}_i,\boldsymbol{v}_j)$ for all $i<j$. For example, in the theory $T=\langle\mathbb{R};+,<,1,0\rangle$ of Linear Real Arithmetic, we have $T\models\exists^{\mathsf{ram}}x,y\colon(x<y\wedge x>99\wedge y<100)$ because there are infinitely many numbers between 99 and 100.

How do Ramsey quantifiers connect to proving termination/liveness? Let us take a proof rule for termination/liveness from [Podelski and Rybalchenko 2004], which concerns covering the transitive closure R^+ of a relation R by well-founded relations (or a finite disjunction thereof). At its simplest form, we obtain a verification condition of asserting

$$R \subseteq T \quad \wedge \quad T \circ R \subseteq T \quad \wedge \quad T \text{ is well-founded.} \tag{1}$$

Such a T satisfying the first two conjuncts is said to be an *inductive relation* [Podelski and Rybalchenko 2004]. The disjunctive well-founded version can be stated similarly, but with a disjunction of relations T_i instead of just a single T. Here, one defines a relation to be *well-founded* if there is no infinite T-chain, i.e., $s_1, s_2, \ldots$ such that $(s_i, s_{i+1}) \in T$ for each i. Clearly, if T is well-founded, then there is no T-loop (i.e. x with $T(x, x)$) and no infinite T-clique. Thus, T also satisfies

$$R \subseteq T \quad \wedge \quad T \circ R \subseteq T \quad \wedge \quad \text{no } T\text{-loop} \quad \wedge \quad \text{no infinite } T\text{-clique.} \tag{2}$$

Note that (2) also implies termination of R, despite imposing a weaker requirement on T. However, the conditions in (2) are easily expressed with the Ramsey quantifier: The absence of T-loops is a first-order property (i.e. $\neg\exists x\colon T(x, x)$) and the absence of an infinite T-clique is definable with the help of a Ramsey quantifier $\neg\exists^{\mathsf{ram}} x, y\colon T(x, y)$. As an example for a covering that satisfies (2) but not (1), consider the well-founded relation $R = \{(i + 1, i) \mid i \in \mathbb{N}\}$ and the covering $T = \{(i, j) \mid i, j \in \mathbb{N},\ i > j\} \cup \{(i, i - 1) \mid i \leq 0\}$. Then $R^+ \subseteq T$ and T is loop-free and contains no (directed) infinite clique, hence (2) proves termination of R. However, T is not well-founded.

Most techniques for handling Ramsey quantifiers proceed by eliminating them. In the early 1980s, Schmerl and Simpson [1982] showed that in LIA, Ramsey quantifiers $\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y}$ can be eliminated if $\boldsymbol{x}$ and $\boldsymbol{y}$ are single variables (hence, it is about cliques of numbers, not vectors). [Actually, their result concerns only undirected cliques, but the proof easily generalizes to directed cliques.] At the turn of the 21st centry, Dang and Ibarra [2002] provided a procedure to decide whether a given relation R described in LIRA admits an infinite directed clique. Their proof yields that general Ramsey quantifiers (i.e. about vectors) can be eliminated in LIRA: The procedure transforms the input formula into a LIRA formula that holds if and only if R admits an infinite directed clique. However, the procedure of Dang and Ibarra [2002] (i) requires the input formula to be quantifier-free (this is also the case for Schmerl and Simpson [1982]) and (ii) yields a formula with several quantifier alternations. Because of (ii), the algorithm needs to then decide the truth of a LIRA formula with quantifier alternations, for which Dang and Ibarra [2002, p. 924] provide (based on Weispfenning [1999]) a doubly exponential time bound of $2^{L^{n^c}}$ for a constant c, where L is the length of the input formula and n is the number of variables. Because of (i), applying the algorithm to an existential LIRA formula φ for R necessitates an elimination of the existential quantifiers from φ. If φ is of length ℓ with q quantified variables, then according to Weispfenning [1999, Theorem 5.1], this results in a quantifier-free formula of size $2^{\ell^{q^d}}$ for some constant d. Plugging this into the construction of Dang and Ibarra [2002] yields a triply exponential time bound of $2^{2^{n^c \cdot \ell^{q^d}}}$.

More recent results [Bergsträßer et al. 2022; To and Libkin 2008] on eliminating Ramsey quantifiers over *theories of string (resp. tree) automatic structures* are also worth mentioning. These are rich classes of logical structures whose domains/relations can be encoded using string/tree automata [Benedikt et al. 2003; Blumensath and Grädel 2000], and subsume various arithmetic theories including LIA and Skolem Arithmetic (i.e. $\langle \mathbb{Z}; \times, \leq, 1, 0\rangle$). Among others, this gives rise to a decision procedure for LIA with Ramsey quantifiers, which runs in exponential time. The main problem with the decision procedures given in [Bergsträßer et al. 2022; To and Libkin 2008] is that they cannot be implemented directly on top of an existing (and highly optimized) SMT-solver, and their complexity is rather high. Secondly, it does not yield algorithms for LRA and LIRA. In fact, the common extension of LIRA and automatic structures are the so-called ω-automatic structures, for which eliminability of Ramsey quantifiers is a long-standing open problem [Kuske 2010].

Monadic decomposability. Another application of Ramsey quantifiers is the analysis of variable dependencies (i.e. monadic decomposability [Veanes et al. 2017]) of formulas. Loosely speaking, a formula $\varphi(x_1, \ldots, x_n)$ is *monadically decomposable* in the theory $\mathcal{T}$ if it is equivalent (over $\mathcal{T}$) to a boolean combination of $\mathcal{T}$-formulas of the form $\varphi(x_i)$, i.e., with at most one x_i as a free variable. This boolean combination of monadic formulas is a monadic decomposition of φ. For example, the formula $x_1 + x_2 \geq 2 \wedge x_1 \geq 0 \wedge x_2 \geq 0$ is monadically decomposable in LIA as it is equivalent to

$$(x_1 \geq 2 \wedge x_2 \geq 0) \vee (x_1 \geq 1 \wedge x_2 \geq 1) \vee (x_1 \geq 0 \wedge x_2 \geq 0).$$

Monadic decompositions have numerous applications in formal verification including string analysis [Hague et al. 2020; Veanes et al. 2017] and query optimization in constraint databases [Grumbach et al. 2001; Kuper et al. 2000]. Veanes et al. [2017] gave a generic semi-algorithm that is guaranteed to output a monadic decomposition of SMT formulas, if such a decomposition exists. To make this semi-algorithm terminating, one may incorporate a monadic decomposability check, which exists for numerous theories [Barceló et al. 2019; Bergsträßer et al. 2022; Hague et al. 2020; Libkin 2003; Veanes et al. 2017]. However, most of these algorithms have very high computational complexity, and for some theories the precise computational complexity is still an open problem. Recently, Hague et al. [2020] have shown that monadic decomposability of quantifier-free LIA formulas is coNP-complete, in contrast to the previously known double exponential-time algorithm [Libkin 2003]. In case of quantifier-free LRA and LIRA the precise complexity is still open. Although both of which can be shown to be decidable in PSPACE [Bergsträßer and Ganardi 2023a].

Contributions. The main contribution of our paper is new algorithms for eliminating Ramsey quantifiers for three common SMT theories: Linear Integer Arithmetic (LIA), Linear Real Arithmetic (LRA), and Linear Integer Real Arithmetic (LIRA). If we restrict to existential fragments, the algorithms run in polynomial time and produce formulas of linear size. [Here, in the definition of size we assume that every variable occurrence has length one.] As a consequence, SMT over these theories can be extended with Ramsey quantifiers only with a small overhead on SMT-solvers. Our results substantially improve the complexity of the elimination procedures of Ramsey quantifiers from [Dang and Ibarra 2002; Schmerl and Simpson 1982], which run in at least double exponential time. This has direct applications in proving program termination (especially, connected to well-foundedness checks) and monadic decomposability (including, the precise complexity for LIA/LRA/LIRA). We detail our contributions below.

Key novel ingredients. We circumvent the high complexities of [Dang and Ibarra 2002; Schmerl and Simpson 1982] as follows. The *first key ingredient* is a procedure to eliminate existential quantifiers *in the context of Ramsey quantifiers*: We prove that any formula

$$\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \exists \boldsymbol{w} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{w}) \tag{3}$$

with some quantifier-free φ and quantifier block $\exists \boldsymbol{w}$ is equivalent to

$$\exists^{\mathsf{ram}} (\boldsymbol{x}, \boldsymbol{r}, \boldsymbol{s}), (\boldsymbol{y}, \boldsymbol{t}, \boldsymbol{u}) \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{s} + \boldsymbol{t}). \tag{4}$$

Note that the formula (3) says that there exists a sequence $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$ of vectors such that for any $i < j$, there exists a $\boldsymbol{b}_{i,j}$ with $\varphi(\boldsymbol{a}_i, \boldsymbol{a}_j, \boldsymbol{b}_{i,j})$. The equivalence says that if such $\boldsymbol{b}_{i,j}$ exist, then there are $\boldsymbol{b}_i, \boldsymbol{b}'_i$ such that one can choose $\boldsymbol{b}_{i,j} := \boldsymbol{b}'_i + \boldsymbol{b}_j$ to satisfy φ. This comes as a surprise, because instead of needing to choose a vector for each *edge* of an infinite clique, it suffices to merely choose an additional vector at each *node*. This non-obvious structural result about infinite cliques yields an algorithmically extremly simple elimination of quantifiers, which just replaces (3) with (4).

Our *second key ingredient* allows us to express the existence of an infinite directed clique in an *existential* formula. Very roughly speaking, Dang and Ibarra [2002] express the existence of an infinite directed clique by saying that for each k, there exists a vector $\boldsymbol{a}_k$ for which some measure is at least k. By choosing appropriate measures, this ensures that the sequence $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$ has certain unboundedness or convergence properties. In contrast, our formula expresses the existence of vectors $\boldsymbol{a}$, $\boldsymbol{d}_c$, and $\boldsymbol{d}_\infty$, such that the sequence

$$\boldsymbol{a}_k = \boldsymbol{a} - \tfrac{1}{k}\boldsymbol{d}_c + k\boldsymbol{d}_\infty \tag{5}$$

has an infinite clique as a subsequence. Here, the vector $\boldsymbol{d}_c$ is used to enable convergence behavior (and is not needed in the case of LIA). Note that this is possible despite the fact that there are formulas φ for which no infinite φ-clique can be written in the form above. However, one can always find an infinite φ-clique as a subsequence of a sequence as in (5).

Proving termination/non-termination. Since well-foundedness of an inductive relation can be expressed by means of an SMT formula with a Ramsey quantifier, our quantifier elimination procedure yields a formula ψ without Ramsey quantifiers, whose size is linear in the size of the original formula φ expressing the verification condition. This means that we can prove termination by simply checking satisfiability of ψ, which can be checked easily by an SMT-solver. Similarly, if we provide an *underapproximation* $T \subseteq R^+$, we may use this to prove non-termination of R by simply checking satisfiability of $T(\boldsymbol{x}, \boldsymbol{x}) \vee \exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y}\colon T(\boldsymbol{x}, \boldsymbol{y})$. By the same token, the Ramsey quantifier can be eliminated using our algorithm, resulting in a formula of linear size that can be easily handled by SMT-solvers.

In fact, one can combine our results with various semi-algorithms for computing approximations of reachability relations (e.g. [Bardin et al. 2008, 2005; Boigelot and Herbreteau 2006; Boigelot et al. 2003; Legay 2008]), yielding a semi-algorithm for deciding termination/non-termination with completeness guarantee for many classes of infinite-state systems operating over integer and real variables. These include classes of hybrid systems and timed systems (e.g. timed pushdown systems), reversal-bounded counter systems, and continuous vector addition systems with states. For these, we also obtain tight computational complexity for the problem.

Monadic decomposition. Our procedure reduces a monadic decomposability check for a LIA, LRA, or LIRA formula to linearly many unsatisfiability queries over the same theory. As before, the resulting formulas without Ramsey quantifiers is of linear size, which can be handled easily by SMT-solvers. This reduction also shows that monadic decomposability for LIA/LRA/LIRA is in coNP, which can be shown to be the precise complexity for the problems. The coNP complexity for LIA was shown already by Hague et al. [2020], but with a completely different reduction (and no experimental validation). The coNP complexity of monadic decomposability for LRA/LIRA is new and answers the open questions posed by Veanes et al. [2017] and Bergsträßer and Ganardi [2023a].

Implementation. We have implemented a prototype of our elimination algorithms for LIA, LRA, and LIRA and tested it on two sets of micro-benchmarks. The first benchmarks contain examples where a single Ramsey quantifier has to be eliminated. Such formulas can for example be derived from program (non-)termination. With the second benchmarks we use our algorithms to check monadic decomposability as described above. Here, we compare our algorithm to the ones in [Veanes et al. 2017] and [Markgraf et al. 2021]. For both sets of benchmarks we obtain promising results. The implementation is available at [Bergsträßer et al. 2023b] and the full version of this paper at [Bergsträßer et al. 2023a].

2 MORE DETAILED EXAMPLES

In this section, we give concrete examples illustrating the problems of proving termination and non-termination, and how these give rise to verification conditions involving Ramsey quantifiers. We then discuss how our algorithms eliminate Ramsey quantifiers from these verification conditions.

Proving termination. We consider a simplified version of McCarthy 91 program [Manna and Pnueli 1970]. The program has two integer variables n, m and applies the following rules till termination:

(1) $n := n - 1$ and $m := m - 1$, if $n > 0$ and $m \geq 0$.
(2) $n := n + 1$ and $m := m + 2$, if $n > 0$ and $m < 0$.

The interesting case in this termination proof is when $-1 \leq m \leq 1$. For simplicity, we will only deal with this. In the sequel, we will write $R \subseteq \mathbb{N}^2 \times \mathbb{N}^2$ to denote the relation generated by the program restricted to $-1 \leq m \leq 1$.

To prove termination, we will need to annotate the program with an inductive relation $T \subseteq \mathbb{N}^2 \times \mathbb{N}^2$ that is well-founded. Define T as the conjunction of $n > 0 \wedge -1 \leq m, m' \leq 1 \wedge n' \geq 0$ and disjunctions of relations $T_1, \ldots, T_6$ as specified below.

$$T_1 := m' = 0 \wedge n' = n \wedge m = -1 \wedge n = 1 \qquad T_2 := m' = 1 \wedge n' = n + 1 \wedge m = -1 \wedge n \geq 1$$
$$T_3 := m' > m \wedge n' = n \wedge n \geq 2 \qquad T_4 := n' < n \wedge n' \geq 0 \wedge m \leq 0$$
$$T_5 := n' < n \wedge n' \geq 0 \wedge m = 1 \wedge m' \geq 0 \qquad T_6 := n' < n - 1 \wedge n' \geq 0 \wedge m = 1 \wedge m' = -1$$

The condition that T is inductive is easily phrased as satisfiability of a quantifier-free LIA formula:

$$[R(n, m, n', m') \wedge \neg T(n, m, n', m')] \vee [T(n, m, n', m') \wedge R(n', m', n'', m'') \wedge \neg T(n, m, n'', m'')].$$

We need to prove unsatisfiability of this formula, which can be easily checked using a LIA solver, which is supported by major SMT-solvers (e.g. Z3 [de Moura and Bjørner 2008]). To prove well-foundedness of T, we consider two cases. The looping case also easily translates into a LIA formula:

$$T(n, m, n', m') \wedge T(n', m', n', m').$$

Again, we need to prove that this is unsatisfiable. The non-looping case is the one that requires a Ramsey quantifier where we need to prove unsatisfiability of:

$$\exists^{\mathsf{ram}}(n, m), (n', m')\colon T(n, m, n', m').$$

Proving non-termination. Let us now present an example of a program (see Figure 1) where Ramsey quantifiers can be used to prove non-termination.

```
real x1 ← input-real();
int x2 ← input-int();
assert x1 > 0;
while x2 > 0 do
    real t1 ← input-real();
    assert t1 ≥ 0.5x1 + 0.5;
    x1 ← t1;
    int t2 ← input-int();
    assert t2 ≥ 0;
    x2 ← x2 − ⌊x1⌋ − t2;
end
```

Fig. 1. Example of a non-terminating program.

The reachability relation $\to_1$ for x_1 and x_2 after the first iteration of the while-loop is $(x_1, x_2) \to_1 (y_1, y_2)$ such that $x_1 > 0 \wedge x_2 > 0 \wedge y_1 \geq 0.5x_1 + 0.5 \wedge y_2 \leq x_2 - \lfloor x_1 \rfloor$. It turns out that $\to_1$ is already an under-approximation of the reachability relation $\to^+$ after at least one iteration. Thus, to show non-termination, it suffices to show that $\to_1$ and therefore $\to^+$ has an infinite clique. For example we find the clique $(a_i, b_i)_{i \geq 1}$ with $a_1 = 0.5$, $a_{i+1} = 0.5a_i + 0.5$, and $b_i = 1$ for all $i \geq 1$ which corresponds to choosing $x_1 = 0.5$ and $x_2 = 1$ at the beginning and

$t_1 = 0.5x_1 + 0.5$ and $t_2 = 0$ in each iteration. Here we can see that $(a_i)_{i\geq 1}$ converges against 1 but never reaches 1, which means that $\lfloor a_i \rfloor$ is always 0 for all $i \geq 1$.

Illustration of how to remove a Ramsey quantifier. Let us see an example of Ramsey quantifier elimination. Consider the following formula:

$$\varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}) \quad = \quad x_1 + \frac{z_1 - x_1}{2} \leq y_1 \leq z_1 \quad \wedge \quad x_2 + \frac{z_2 - x_2}{2} \leq y_2 \leq z_2 \quad \wedge \quad y_2 = \lfloor y_1 \rfloor,$$

in which $\boldsymbol{x} = (x_1, x_2)$, $\boldsymbol{y} = (y_1, y_2)$, and $\boldsymbol{z} = (z_1, z_2)$. Now we claim that $\exists^{\mathrm{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ is equivalent to

$$z_2 = \lfloor z_1 \rfloor \quad \vee \quad (z_1 = \lfloor z_1 \rfloor \wedge z_2 = z_1 - 1).$$

Note that $\exists^{\mathrm{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ expresses that there exists a sequence $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots \in \mathbb{R}^2$ such that the first components converge from below against z_1: The first conjunct in φ requires the first component of $\boldsymbol{a}_j$ to have at most half the distance to z_1 as the first component of $\boldsymbol{a}_i$, for every $i < j$. Similarly, the second conjunct forces the second components to converge against z_2.

Furthermore, the third conjunct requires the second components of $\boldsymbol{a}_i$ to be the floor of the first component of $\boldsymbol{a}_i$, for every i. Thus, if z_1 is not an integer, then the first components of $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$ will eventually be between $\lfloor z_1 \rfloor$ and z_1. Thus, the second components must eventually be equal to $\lfloor z_1 \rfloor$ and thus also their limit: $z_2 = \lfloor z_1 \rfloor$. However, if z_1 is an integer, then there is another option: The first components can all be strictly smaller than $z_1 = \lfloor z_1 \rfloor$. But then the second components must eventually be equal to $\lfloor z_1 \rfloor - 1$, and thus $z_2 = z_1 - 1$.

3 PRELIMINARIES

We denote a vector with components $(x_1, \ldots, x_k)$ of dimension k with a boldface letter $\boldsymbol{x}$ and for numbers n we write $\boldsymbol{n}$ for a vector $(n, \ldots, n)$ of appropriate dimension. On vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ of dimension k we define the usual pointwise partial order $\boldsymbol{x} \leq \boldsymbol{y}$ such that $x_i \leq y_i$ for all $1 \leq i \leq k$. Moreover, we define $\boldsymbol{x} \ll \boldsymbol{y}$ if $x_i < y_i$ for *every* i.

To reduce the usage of parentheses, we assume the binding strengths of logical operators to be $\neg, \wedge, \vee, \rightarrow$ in decreasing order and quantifiers bind the weakest.

We define the *size* of a formula by the length of its usual encoding where we assume that every variable occurrence has length one. In the following we formally only define formulas with constants 0 and 1, but we will also use arbitrary constants that, when encoded in binary, can be eliminated with only a linear blow-up in the above size definition. Note that for implementations, it would also make sense to measure the length of writing the formula using a fixed alphabet, which would incur a logarithmic-length string per variable occurrence.

3.1 Linear Integer Arithmetic

Linear Integer Arithmetic (LIA) is defined as the first-order theory with the structure $\langle \mathbb{Z}; +, <, 1, 0 \rangle$. LIA is also called *Presburger arithmetic* and we will use these terms interchangeably. We will only work on the existential fragment of LIA, i.e., formulas of the form $\exists \boldsymbol{x} \colon \varphi(\boldsymbol{x}, \boldsymbol{z})$ where the variables in $\boldsymbol{x}$ are bound by the existence quantifier and $\boldsymbol{z}$ is a vector of free variables.

Proposition 3.1 ([Borosh and Treybig 1976]). *Satisfiability of existential formulas in LIA is* NP*-complete.*

To admit quantifier elimination, one has to enrich the structure $\langle \mathbb{Z}; +, <, 1, 0 \rangle$ by modulo constraints. A modulo constraint is a binary predicate $\equiv_e$ with $e > 0$ such that $s \equiv_e t$ is fulfilled if and only if $e | s - t$. Note that modulo constraints are definable in $\langle \mathbb{Z}; +, <, 1, 0 \rangle$ using existence quantifiers, which means that $\langle \mathbb{Z}; +, <, 0, 1, (\equiv_e)_{e>0} \rangle$ is still a structure for LIA. The following was famously shown by Presburger [1929] (see [Weispfenning 1997] for complexity considerations):

Proposition 3.2. *LIA with the structure $\langle \mathbb{Z}; +, <, 0, 1, (\equiv_e)_{e>0} \rangle$ admits quantifier elimination.*

3.2 Linear Real Arithmetic

In addition to the integers, we also consider linear arithmetic over the reals. *Linear Real Arithmetic* (LRA) is defined as the first-order theory with the structure $\langle \mathbb{R}; +, <, 1, 0 \rangle$. As for LIA, the satisfiability problem for the existential fragment of LRA is NP-complete [Sontag 1985, Corollary 3.4]:

Proposition 3.3. *Satisfiability of existential formulas in LRA is* NP*-complete.*

Moreover, quantifiers can already be eliminated over the structure $\langle \mathbb{R}; +, <, 1, 0 \rangle$. This goes back to Fourier [1826] and was rediscovered several times thereafter [Williams 1986]:

Proposition 3.4. *LRA with the structure $\langle \mathbb{R}; +, <, 1, 0 \rangle$ admits quantifier elimination.*

3.3 Linear Integer Real Arithmetic

We define *Linear Integer Real Arithmetic* (LIRA) as the first-order theory with the structure $\langle \mathbb{R}; \lfloor . \rfloor, +, <, 0, 1 \rangle$ where $\lfloor r \rfloor$ denotes the greatest integer smaller than or equal to $r \in \mathbb{R}$. In terms of full first-order logic, this logic is equally expressive as the first-order logic over the structure $\langle \mathbb{R}; \mathbb{Z}, +, <, 0, 1 \rangle$. Here, we focus on $\langle \mathbb{R}; \lfloor . \rfloor, +, <, 0, 1 \rangle$, because its existential fragment is expressively complete [Weispfenning 1999, Theorem 3.1]. Note that by using $x = \lfloor x \rfloor$, we can extend LIRA to allow two sorts of variables: real and integer variables. For a vector $\boldsymbol{x} = (x_1, \dots, x_n)$ of variables let $\boldsymbol{x}^{\mathrm{i/r}}$ denote the vector $(\boldsymbol{x}^{\mathrm{int}}, \boldsymbol{x}^{\mathrm{real}})$ where $\boldsymbol{x}^{\mathrm{int}} = (x_1^{\mathrm{int}}, \dots, x_n^{\mathrm{int}})$ is a vector of integer variables and $\boldsymbol{x}^{\mathrm{real}} = (x_1^{\mathrm{real}}, \dots, x_n^{\mathrm{real}})$ is a vector of real variables. Two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ of dimension n are said to have the same *type* if for all $i \in [1, n]$ we have that x_i and y_i are both real or integer variables. The *separation* of an existential formula $\exists x_1, \dots, x_n \colon \varphi(x_1, \dots, x_n, z_1, \dots, z_m)$ in LIRA is defined as

$$\exists \boldsymbol{x}^{\mathrm{i/r}} \colon \varphi(\boldsymbol{x}^{\mathrm{int}} + \boldsymbol{x}^{\mathrm{real}}, \boldsymbol{z}^{\mathrm{int}} + \boldsymbol{z}^{\mathrm{real}}) \wedge 0 \leq \boldsymbol{x}^{\mathrm{real}} < 1 \wedge 0 \leq \boldsymbol{z}^{\mathrm{real}} < 1$$

where $x_i^{\mathrm{int}}, z_j^{\mathrm{int}}$ are fresh integer variables and $x_i^{\mathrm{real}}, z_j^{\mathrm{real}}$ are fresh real variables that express the integer and real part of x_i and z_j. If x_i (resp. z_j) is an integer variable, we add the constraint $x_i^{\mathrm{real}} = 0$ (resp. $z_j^{\mathrm{real}} = 0$) to the separation. We say that an existential formula in LIRA is *decomposable* if its separation can be written as an existentially quantified Boolean combination of Presburger and LRA formulas (called *decomposition*).

Lemma 3.5. *Every existential formula in LIRA is decomposable. Moreover, its decomposition is of linear size and can be computed in polynomial time.*

Proof. Let $\psi = \exists x_1, \dots, x_n \colon \varphi(x_1, \dots, x_n, z_1, \dots, z_m)$ be an existential formula in LIRA. By introducing new existentially quantified variables, we can assume that every atom of φ is of one of the following forms: (i) $x = 0$, (ii) $x = 1$, (iii) $x + y = z$, (iv) $x < 0$, (v) $x = \lfloor y \rfloor$. Note that the size of the formula is still linear (even if the coefficients are given in binary). Let $\varphi'(\boldsymbol{x}^{\mathrm{i/r}}, \boldsymbol{z}^{\mathrm{i/r}})$ be the formula obtained from $\varphi(\boldsymbol{x}^{\mathrm{int}} + \boldsymbol{x}^{\mathrm{real}}, \boldsymbol{z}^{\mathrm{int}} + \boldsymbol{z}^{\mathrm{real}})$ by replacing every

- $x^{\mathrm{int}} + x^{\mathrm{real}} = 0$ by $x^{\mathrm{int}} = 0 \wedge x^{\mathrm{real}} = 0$,
- $x^{\mathrm{int}} + x^{\mathrm{real}} = 1$ by $x^{\mathrm{int}} = 1 \wedge x^{\mathrm{real}} = 0$,
- $x^{\mathrm{int}} + x^{\mathrm{real}} + y^{\mathrm{int}} + y^{\mathrm{real}} = z^{\mathrm{int}} + z^{\mathrm{real}}$ by

$$(x^{\mathrm{real}} + y^{\mathrm{real}} < 1 \rightarrow x^{\mathrm{int}} + y^{\mathrm{int}} = z^{\mathrm{int}} \wedge x^{\mathrm{real}} + y^{\mathrm{real}} = z^{\mathrm{real}}) \wedge$$
$$(x^{\mathrm{real}} + y^{\mathrm{real}} \geq 1 \rightarrow x^{\mathrm{int}} + y^{\mathrm{int}} + 1 = z^{\mathrm{int}} \wedge x^{\mathrm{real}} + y^{\mathrm{real}} - 1 = z^{\mathrm{real}}),$$

- $x^{\mathrm{int}} + x^{\mathrm{real}} < 0$ by $x^{\mathrm{int}} < 0$, and
- $x^{\mathrm{int}} + x^{\mathrm{real}} = \lfloor y^{\mathrm{int}} + y^{\mathrm{real}} \rfloor$ by $x^{\mathrm{real}} = 0 \wedge x^{\mathrm{int}} = y^{\mathrm{int}}$.

Thus, φ' is a Boolean combination of formulas that either only involve integer variables or real variables. Now the separation of ψ is equivalent to

$$\exists \boldsymbol{x}^{\mathrm{i/r}} \colon \varphi'(\boldsymbol{x}^{\mathrm{i/r}}, \boldsymbol{z}^{\mathrm{i/r}}) \wedge 0 \leq \boldsymbol{x}^{\mathrm{real}} < 1 \wedge 0 \leq \boldsymbol{z}^{\mathrm{real}} < 1$$

where we add $x_i^{\mathrm{real}} = 0$ if x_i is an integer variable and $z_j^{\mathrm{real}} = 0$ if z_j is an integer variable, which is a linear sized decomposition. □

Proposition 3.6. *Satisfiability of existential formulas in LIRA is* NP*-complete.*

Proof. The NP lower bound is inherited from the Presburger (Proposition 3.1) and LRA (Proposition 3.3) case. For the upper bound let φ be an existential formula in LIRA. We first apply Lemma 3.5 to compute a decomposition ψ of φ in polynomial time. Then we guess truth values for the Presburger and LRA subformulas of ψ and verify the guesses in NP using Propositions 3.1 and 3.3. Since φ and its decomposition ψ are equisatisfiable, it remains to check whether the truth values satisfy ψ in order to decide satisfiability of φ. □

3.4 Ramsey Quantifier

Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be two vectors of variables of the same type. For a formula φ in LIRA we define $\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ as the formula that is satisfied by a valuation $\boldsymbol{c}$ of $\boldsymbol{z}$ if and only if there exists a sequence $(\boldsymbol{a}_i)_{i \geq 1}$ of pairwise distinct valuations of $\boldsymbol{x}$ (and $\boldsymbol{y}$) such that $\varphi(\boldsymbol{a}_i, \boldsymbol{a}_j, \boldsymbol{c})$ holds for all $i < j$. The sequence $(\boldsymbol{a}_i)_{i \geq 1}$ with the above properties is called an *infinite clique* of φ with respect to $\boldsymbol{c}$. The *infinite clique problem* asks given a formula $\varphi(\boldsymbol{x}, \boldsymbol{y})$, where $\boldsymbol{x}$ and $\boldsymbol{y}$ have the same type, whether φ has an infinite clique.

The infinite version of Ramsey's theorem can be formulated over graphs as follows:

Theorem 3.7 ([Ramsey 1930]). *Any complete infinite graph whose edges are colored with finitely many colors contains an infinite monochromatic clique.*

We will often use the fact that by Ramsey's theorem $\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}) \vee \psi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ is equivalent to $(\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})) \vee (\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \psi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}))$.

4 ELIMINATING EXISTENTIAL QUANTIFIERS

The first step in our elimination of the Ramsey quantifier in $\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \psi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ is to reduce to the case where ψ is quantifier-free. In LIA and LRA, there are procedures to convert ψ into a quantifier-free equivalent (Propositions 3.2 and 3.4), but these incur a doubly exponential blow-up [Weispfenning 1997]. Instead, we will show the following (perhaps surprising) equivalence:

Theorem 4.1. *Let φ be an existential formula in LIRA. Then the formulas*

$$\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \exists \boldsymbol{w} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{w}, \boldsymbol{z}) \qquad \textit{and} \qquad \exists^{\mathsf{ram}} (\boldsymbol{x}, \boldsymbol{v}_1, \boldsymbol{v}_2), (\boldsymbol{y}, \boldsymbol{w}_1, \boldsymbol{w}_2) \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{v}_1 + \boldsymbol{w}_2, \boldsymbol{z}) \wedge \boldsymbol{x} \neq \boldsymbol{y}$$

are equivalent where $\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{w}_1, \boldsymbol{w}_2$ have the same type as $\boldsymbol{w}$.

Thus, if we write $\psi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ in prenex form as $\exists \boldsymbol{w} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{w}, \boldsymbol{z})$ with a quantifier-free φ, then Theorem 4.1 allows us to eliminate the block $\exists \boldsymbol{w}$ of quantifiers by moving $\boldsymbol{w}$ under the Ramsey quantifier. Note that both formulas express the existence of an infinite clique. The left says that for every edge $\boldsymbol{x} \to \boldsymbol{y}$ in the clique, we can choose a vector $\boldsymbol{w}$ such that $\varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{w}, \boldsymbol{z})$ is satisfied. The right formula says that $\boldsymbol{w}$ can be chosen in a specific way: It says that for each node, one can choose two vectors $(\boldsymbol{w}_1, \boldsymbol{w}_2)$ such that for each edge $\boldsymbol{x} \to \boldsymbol{y}$, the vector $\boldsymbol{w}$ can be the sum of $\boldsymbol{w}_1$ for $\boldsymbol{x}$ and $\boldsymbol{w}_2$ for $\boldsymbol{y}$. Thus, the right-hand formula clearly implies the left-hand formula. The challenging direction is to show that the left-hand formula implies the right-hand formula.

The rest of this section is devoted to proving Theorem 4.1.

4.1 Presburger Arithmetic

We start with LIA, a.k.a. Presburger arithmetic.

Theorem 4.2. *Let φ be an existential formula in Presburger arithmetic. Then the formulas*

$$\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \exists \boldsymbol{w} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{w}, \boldsymbol{z}) \quad \text{and} \quad \exists^{\mathsf{ram}} (\boldsymbol{x}, \boldsymbol{v}_1, \boldsymbol{v}_2), (\boldsymbol{y}, \boldsymbol{w}_1, \boldsymbol{w}_2) \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{v}_1 + \boldsymbol{w}_2, \boldsymbol{z}) \wedge \boldsymbol{x} \neq \boldsymbol{y}$$

are equivalent.

To prove Theorem 4.2, it suffices to prove it in case $\boldsymbol{w}$ consists of just one variable w: Then, Theorem 4.2 follows by induction.

Lemma 4.3. *Let φ be an existential formula in Presburger arithmetic. Then the formulas*

$$\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \exists w \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, w, \boldsymbol{z}) \quad \text{and} \quad \exists^{\mathsf{ram}} (\boldsymbol{x}, v_1, v_2), (\boldsymbol{y}, w_1, w_2) \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, v_1 + w_2, \boldsymbol{z}) \wedge \boldsymbol{x} \neq \boldsymbol{y} \tag{6}$$

are equivalent.

Simple Presburger formulas. Let $\boldsymbol{u}$ be a vector of variables and w be a variable. We say that a Presburger formula $\varphi(\boldsymbol{u}, w)$ is w-*simple* if it is a Boolean combination of formulas of the form $\boldsymbol{r}^\top \boldsymbol{u} + c < w$, $w < \boldsymbol{r}^\top \boldsymbol{u} + c$, and modulo constraints over $\boldsymbol{u}$ and w, where $\boldsymbol{r}$ is a vector over $\mathbb{Z}$, and $c \in \mathbb{Z}$.

Lemma 4.4. *Let $\varphi(\boldsymbol{x}, \boldsymbol{y}, w, \boldsymbol{z})$ be w-simple. Then the formulas*

$$\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \exists w \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, w, \boldsymbol{z}) \quad \text{and} \quad \exists^{\mathsf{ram}} (\boldsymbol{x}, v_1, v_2), (\boldsymbol{y}, w_1, w_2) \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, v_1 + w_2, \boldsymbol{z}) \wedge \boldsymbol{x} \neq \boldsymbol{y}$$

are equivalent.

Proof. We first move all negations in φ inwards to the atoms and possibly negate them (which for modulo constraints introduces disjunctions). We then bring φ into disjunctive normal form and move the Ramsey quantifier and existence quantifier into the disjunction. Since φ is simple, we can assume that it is a conjunction of formulas

$$\alpha_i(\boldsymbol{x}) + \beta_i(\boldsymbol{y}) + \gamma_i(\boldsymbol{z}) + h_i < w$$

for $i = 1, \ldots, n$ and

$$w < \alpha'_j(\boldsymbol{x}) + \beta'_j(\boldsymbol{y}) + \gamma'_j(\boldsymbol{z}) + h'_j$$

for $j = 1, \ldots, m$, and modulo constraints

$$\delta_i(\boldsymbol{x}, \boldsymbol{y}, w, \boldsymbol{z}) \equiv_{e_i} d_i$$

for $i = 1, \ldots, k$. Here, $\alpha_i, \beta_i, \gamma_i, \alpha'_j, \beta'_j, \gamma'_j, \delta_i$ are linear functions. Let $f_i(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}) := \alpha_i(\boldsymbol{x}) + \beta_i(\boldsymbol{y}) + \gamma_i(\boldsymbol{z}) + h_i$ for all $i \in [1, n]$ and $f'_j(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}) := \alpha'_j(\boldsymbol{x}) + \beta'_j(\boldsymbol{y}) + \gamma'_j(\boldsymbol{z}) + h'_j$ for all $j \in [1, m]$. In the following we assume that $n, m > 0$; the other cases are simpler and can be handled similarly.

Assume $\boldsymbol{c} \in \mathbb{Z}^{|\boldsymbol{z}|}$ satisfies the left-hand formula, i.e., there is an infinite sequence $(\boldsymbol{a}_i)_{i \geq 1}$ of pairwise distinct vectors over $\mathbb{Z}$ such that for all $i < j$ there exists $b_{i,j} \in \mathbb{Z}$ such that $\varphi(\boldsymbol{a}_i, \boldsymbol{a}_j, b_{i,j}, \boldsymbol{c})$ holds. By Ramsey's theorem we can take an infinite subsequence such that we can assume that $f_1(\boldsymbol{a}_i, \boldsymbol{a}_j, \boldsymbol{c}) \leq \cdots \leq f_n(\boldsymbol{a}_i, \boldsymbol{a}_j, \boldsymbol{c})$ and $f'_1(\boldsymbol{a}_i, \boldsymbol{a}_j, \boldsymbol{c}) \leq \cdots \leq f'_m(\boldsymbol{a}_i, \boldsymbol{a}_j, \boldsymbol{c})$ for all $i < j$. Thus, it suffices to consider the greatest lower bound $f_n(\boldsymbol{a}_i, \boldsymbol{a}_j, \boldsymbol{c})$ and the smallest upper bound $f'_1(\boldsymbol{a}_i, \boldsymbol{a}_j, \boldsymbol{c})$ on w. Let $f := f_n, \alpha := \alpha_n, \beta := \beta_n, \gamma := \gamma_n$, and $h := h_n$. Let $N := e_1 \cdots e_k$ be the product of all moduli where we set $N := 1$ if $k = 0$. First observe that $b_{i,j}$ can always be chosen from the interval $[f(\boldsymbol{a}_i, \boldsymbol{a}_j, \boldsymbol{c}) + 1, f(\boldsymbol{a}_i, \boldsymbol{a}_j, \boldsymbol{c}) + N]$ for all $i < j$. Since this interval has fixed length N, by Ramsey's theorem we can restrict to an infinite subsequence such that there is a constant $r \in [1, N]$ such that $f(\boldsymbol{a}_i, \boldsymbol{a}_j, \boldsymbol{c}) + r = b_{i,j}$ for all $i < j$. Now if we set $b^1_i := \alpha(\boldsymbol{a}_i)$ and $b^2_i := \beta(\boldsymbol{a}_i) + \gamma(\boldsymbol{c}) + h + r$ for $i \geq 1$, the infinite sequence $(\boldsymbol{a}_i, b^1_i, b^2_i)_{i \geq 1}$ satisfies $\varphi(\boldsymbol{a}_i, \boldsymbol{a}_j, b^1_i + b^2_j, \boldsymbol{c})$ for all $i < j$ as desired. □

General Presburger formulas. Let us now prove Lemma 4.3 for general existential Presburger formulas. Observe that if we can show equivalence of the formulas in Equation (6) for quantifier-free φ with modulo constraints, then the same follows for general φ: Since for each φ, there exists an equivalent quantifier-free φ' with modulo constraints, we can apply the equivalence in Lemma 4.3 to φ', which implies the same for φ itself. Therefore, we may assume that φ is quantifier-free, but contains modulo constraints.

We now modify φ as in the standard quantifier elimination procedure for Presburger arithmetic. To this end, we define the "w-simplification" of a quantifier-free formula $\theta(\boldsymbol{u}, w)$ with modulo constraints that has free variables $\boldsymbol{u}$ and w. This means, θ is a Boolean combination of inequalities of the form $\boldsymbol{r}^\top\boldsymbol{u} + c \sim sw$, where $\sim\, \in \{<, >\}$, and modulo constraints $\boldsymbol{r}^\top\boldsymbol{u} + sw \equiv_e c$ for some vector $\boldsymbol{r}$ and $c, s \in \mathbb{Z}$. (Note that in Presburger arithmetic equality can be expressed by a conjunction of two strict inequalities.) Let N be the least common multiple of all coefficients s of w in these constraints. We obtain θ' from θ by replacing each inequality $\boldsymbol{r}^\top\boldsymbol{u} + c \sim sw$ with $\frac{N}{s}\boldsymbol{r}^\top\boldsymbol{u} + \frac{N}{s}c \sim w$ and replacing each modulo constraint $\boldsymbol{r}^\top\boldsymbol{u} + sw \equiv_e c$ with $\frac{N}{s}\boldsymbol{r}^\top\boldsymbol{u} + w \equiv_{\frac{N}{s}e} \frac{N}{s}c$. Now the *$w$-simplification* of φ is the pair (ψ, N), where $\psi(\boldsymbol{u}, w) = \varphi'(\boldsymbol{u}, w) \wedge w \equiv_N 0$. Then clearly, ψ is w-simple and for every integer vector $\boldsymbol{a}$ and $b \in \mathbb{Z}$, we have

$$\theta(\boldsymbol{a}, b) \text{ if and only if } \psi(\boldsymbol{a}, Nb)$$

and moreover, $\psi(\boldsymbol{a}, b)$ implies that b is a multiple of N.

Now suppose $\varphi(\boldsymbol{x}, \boldsymbol{y}, w, \boldsymbol{z})$ is quantifier-free, but contains modulo constraints. Moreover, let $\psi(\boldsymbol{x}, \boldsymbol{y}, w, \boldsymbol{z})$ and N be the w-simplification of φ. To show Lemma 4.3, let us assume the left-hand formula in Equation (6) is satisfied for some integer vector $\boldsymbol{c}$. Then $\exists^{\mathsf{ram}}\boldsymbol{x}, \boldsymbol{y} \colon \exists w \colon \psi(\boldsymbol{x}, \boldsymbol{y}, w, \boldsymbol{c})$ holds, because we can multiply the witness values by N. By Lemma 4.4, this implies that

$$\exists^{\mathsf{ram}}(\boldsymbol{x}, v_1, v_2), (\boldsymbol{y}, w_1, w_2) \colon \psi(\boldsymbol{x}, \boldsymbol{y}, v_1 + w_2, \boldsymbol{c})$$

is satisfied, meaning there exists a sequence $(\boldsymbol{a}_i, b_i, b'_i)_{i\geq 1}$ where $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$ are pairwise distinct and where $\psi(\boldsymbol{a}_i, \boldsymbol{a}_j, b_i + b'_j, \boldsymbol{c})$ for every $i < j$. By construction of ψ, this implies that $b_i + b'_j$ is a multiple of N for every $i < j$ and therefore all the numbers $b_1, b_2, \ldots$ must have the same remainder modulo N, say $r \in [0, N-1]$, and all the numbers $b'_1, b'_2, \ldots$ must be congruent to $-r$ modulo N. This means, the numbers $\bar{b}_i = (b_i - r)/N$ and $\bar{b}'_i = (b'_i + r)/N$ must be integers. Then for every $i < j$, we have $\psi(\boldsymbol{a}_i, \boldsymbol{a}_j, N(\bar{b}_i + \bar{b}'_j), \boldsymbol{c})$ and hence $\varphi(\boldsymbol{a}_i, \boldsymbol{a}_j, \bar{b}_i + \bar{b}'_j, \boldsymbol{c})$. Thus, the sequence $(\boldsymbol{a}_i, \bar{b}_i, \bar{b}'_i)_{i\geq 1}$ shows that $\exists^{\mathsf{ram}}(\boldsymbol{x}, v_1, v_2), (\boldsymbol{y}, w_1, w_2) \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, v_1 + w_2, \boldsymbol{c}) \wedge \boldsymbol{x} \neq \boldsymbol{y}$ is satisfied.

4.2 Linear Real Arithmetic

We now turn to the case where φ is a formula in LRA.

Theorem 4.5. *Let φ be an existential formula in LRA. Then the formulas*

$$\exists^{\mathsf{ram}}\boldsymbol{x}, \boldsymbol{y} \colon \exists \boldsymbol{w} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{w}, \boldsymbol{z}) \quad \text{and} \quad \exists^{\mathsf{ram}}(\boldsymbol{x}, \boldsymbol{v}_1, \boldsymbol{v}_2), (\boldsymbol{y}, \boldsymbol{w}_1, \boldsymbol{w}_2) \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{v}_1 + \boldsymbol{w}_2, \boldsymbol{z}) \wedge \boldsymbol{x} \neq \boldsymbol{y}$$

are equivalent.

We may assume that $\boldsymbol{w}$ consists of just one variable w: Then Theorem 4.5 follows by induction.

Lemma 4.6. *Let φ be an existential formula in LRA. Then the formulas*

$$\exists^{\mathsf{ram}}\boldsymbol{x}, \boldsymbol{y} \colon \exists w \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, w, \boldsymbol{z}) \quad \text{and} \quad \exists^{\mathsf{ram}}(\boldsymbol{x}, v_1, v_2), (\boldsymbol{y}, w_1, w_2) \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, v_1 + w_2, \boldsymbol{z}) \wedge \boldsymbol{x} \neq \boldsymbol{y}$$

are equivalent.

Proof. Again by eliminating quantifiers in φ, bringing it into disjunctive normal form, and moving the quantifiers into the disjunction, we assume that φ is a conjunction of formulas

$$\alpha_i(\boldsymbol{x}) + \beta_i(\boldsymbol{y}) + \gamma_i(z) + h_i < w$$

for $i = 1, \ldots, n$ and

$$w < \alpha'_j(\boldsymbol{x}) + \beta'_j(\boldsymbol{y}) + \gamma'_j(z) + h'_j$$

for $j = 1, \ldots, m$, and equality constraints

$$w = \delta_i(\boldsymbol{x}) + \kappa_i(\boldsymbol{y}) + \lambda_i(z) + d_i$$

for $i = 1, \ldots, k$. Here, $\alpha_i, \beta_i, \gamma_i, \alpha'_j, \beta'_j, \gamma'_j, \delta_i, \kappa_i, \lambda_i$ are linear functions with rational coefficients and $h_i, h'_j, d_i \in \mathbb{Q}$ are constants.

Assume $\boldsymbol{c} \in \mathbb{R}^{|z|}$ satisfies the left-hand formula, i.e., there is an infinite sequence $(\boldsymbol{a}_i)_{i \geq 1}$ of pairwise distinct vectors over $\mathbb{R}$ such that for all $i < j$ there exists $b_{i,j} \in \mathbb{R}$ such that $\varphi(\boldsymbol{a}_i, \boldsymbol{a}_j, b_{i,j}, \boldsymbol{c})$ holds. Clearly, if $k > 0$, we can eliminate w by replacing it by $\delta_1(\boldsymbol{x}) + \kappa_1(\boldsymbol{y}) + \lambda_1(z) + d_1$. Thus, setting $b_i := \delta_1(\boldsymbol{a}_i)$ and $b'_i := \kappa_1(\boldsymbol{a}_i) + \lambda_1(\boldsymbol{c}) + d_1$ for $i \geq 1$, the sequence $(\boldsymbol{a}_i, b_i, b'_i)_{i \geq 1}$ satisfies $\varphi(\boldsymbol{a}_i, \boldsymbol{a}_j, b_i + b'_j, \boldsymbol{c})$ for all $i < j$. So assume $k = 0$, i.e., φ only contains lower and upper bounds on w. We further assume that $n, m > 0$ since the other cases are obvious. As in the Presburger case we can apply Ramsey's theorem so that we only have to consider the greatest lower bound $\alpha(\boldsymbol{x}) + \beta(\boldsymbol{y}) + \gamma(z) + h$ and the smallest upper bound $\alpha'(\boldsymbol{x}) + \beta'(\boldsymbol{y}) + \gamma'(z) + h'$ on w. This means that w can always be chosen to be the midpoint of this interval. Therefore, if we set $b_i := (\alpha(\boldsymbol{a}_i) + \alpha'(\boldsymbol{a}_i))/2$ and $b'_i := (\beta(\boldsymbol{a}_i) + \gamma(\boldsymbol{c}) + h + \beta'(\boldsymbol{a}_i) + \gamma'(\boldsymbol{c}) + h')/2$ for $i \geq 1$, the sequence $(\boldsymbol{a}_i, b_i, b'_i)_{i \geq 1}$ satisfies $\varphi(\boldsymbol{a}_i, \boldsymbol{a}_j, b_i + b'_j, \boldsymbol{c})$ for all $i < j$. □

4.3 Linear Integer Real Arithmetic

We are now ready to prove Theorem 4.1. As mentioned above, it suffices to prove the "only if" direction. Let $\psi(\boldsymbol{z})$ be the left-hand formula and $\boldsymbol{c}$ be a valuation of $\boldsymbol{z}$ that satisfies ψ. By Lemma 3.5 there exists a decomposition $\varphi'(\boldsymbol{x}^{\mathrm{i/r}}, \boldsymbol{y}^{\mathrm{i/r}}, \boldsymbol{w}^{\mathrm{i/r}}, \boldsymbol{z}^{\mathrm{i/r}})$ of $\varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{w}, \boldsymbol{z})$. We define the formula $\psi'(\boldsymbol{z}^{\mathrm{i/r}}) := \exists^{\mathrm{ram}} \boldsymbol{x}^{\mathrm{i/r}}, \boldsymbol{y}^{\mathrm{i/r}} \colon \exists \boldsymbol{w}^{\mathrm{i/r}} \colon \varphi'(\boldsymbol{x}^{\mathrm{i/r}}, \boldsymbol{y}^{\mathrm{i/r}}, \boldsymbol{w}^{\mathrm{i/r}}, \boldsymbol{z}^{\mathrm{i/r}})$. By definition of a decomposition, there is a valuation $\boldsymbol{c}^{\mathrm{i/r}}$ of $\boldsymbol{z}^{\mathrm{i/r}}$ with $\boldsymbol{c}^{\mathrm{int}} + \boldsymbol{c}^{\mathrm{real}} = \boldsymbol{c}$ that satisfies ψ'. We bring φ' into disjunctive normal form

$$\bigvee_{i=1}^{n} \alpha_i(\boldsymbol{x}^{\mathrm{int}}, \boldsymbol{y}^{\mathrm{int}}, \boldsymbol{w}^{\mathrm{int}}, \boldsymbol{z}^{\mathrm{int}}) \wedge \beta_i(\boldsymbol{x}^{\mathrm{real}}, \boldsymbol{y}^{\mathrm{real}}, \boldsymbol{w}^{\mathrm{real}}, \boldsymbol{z}^{\mathrm{real}})$$

where α_i is an existential Presburger formula and β_i is an existential formula in LRA. By Ramsey's theorem there exists $1 \leq i \leq n$ such that

$$\exists^{\mathrm{ram}} \boldsymbol{x}^{\mathrm{i/r}}, \boldsymbol{y}^{\mathrm{i/r}} \colon \exists \boldsymbol{w}^{\mathrm{int}} \colon \alpha_i(\boldsymbol{x}^{\mathrm{int}}, \boldsymbol{y}^{\mathrm{int}}, \boldsymbol{w}^{\mathrm{int}}, \boldsymbol{c}^{\mathrm{int}}) \wedge \exists \boldsymbol{w}^{\mathrm{real}} \colon \beta_i(\boldsymbol{x}^{\mathrm{real}}, \boldsymbol{y}^{\mathrm{real}}, \boldsymbol{w}^{\mathrm{real}}, \boldsymbol{c}^{\mathrm{real}}).$$

Note that the existentially quantified variables can be split at the conjunction into the real and integer part. To perform a similar splitting for the variables bound by the Ramsey quantifier, we have to distinct the two cases whether the vectors of the clique are pairwise distinct in the real components or in the integer components. We only show the case where the vectors of the clique are pairwise distinct in both the real and integer components. The other cases are similar by allowing that either the integer or real components do not change throughout the clique, i.e., either $\exists \boldsymbol{x}^{\mathrm{int}}, \boldsymbol{w}^{\mathrm{int}} \colon \alpha_i(\boldsymbol{x}^{\mathrm{int}}, \boldsymbol{x}^{\mathrm{int}}, \boldsymbol{w}^{\mathrm{int}}, \boldsymbol{c}^{\mathrm{int}})$ or $\exists \boldsymbol{x}^{\mathrm{real}}, \boldsymbol{w}^{\mathrm{real}} \colon \beta_i(\boldsymbol{x}^{\mathrm{real}}, \boldsymbol{x}^{\mathrm{real}}, \boldsymbol{w}^{\mathrm{real}}, \boldsymbol{c}^{\mathrm{real}})$ holds. So we assume that

$$\exists^{\mathrm{ram}} \boldsymbol{x}^{\mathrm{int}}, \boldsymbol{y}^{\mathrm{int}} \colon \exists \boldsymbol{w}^{\mathrm{int}} \colon \alpha_i(\boldsymbol{x}^{\mathrm{int}}, \boldsymbol{y}^{\mathrm{int}}, \boldsymbol{w}^{\mathrm{int}}, \boldsymbol{c}^{\mathrm{int}}) \wedge \exists^{\mathrm{ram}} \boldsymbol{x}^{\mathrm{real}}, \boldsymbol{y}^{\mathrm{real}} \colon \exists \boldsymbol{w}^{\mathrm{real}} \colon \beta_i(\boldsymbol{x}^{\mathrm{real}}, \boldsymbol{y}^{\mathrm{real}}, \boldsymbol{w}^{\mathrm{real}}, \boldsymbol{c}^{\mathrm{real}}).$$

By applying Theorem 4.2 to the first conjunct and Theorem 4.5 to the second conjunct, we get

$$\exists^{\mathsf{ram}}(\boldsymbol{x}^{\mathrm{int}},\boldsymbol{v}_1^{\mathrm{int}},\boldsymbol{v}_2^{\mathrm{int}}),(\boldsymbol{y}^{\mathrm{int}},\boldsymbol{w}_1^{\mathrm{int}},\boldsymbol{w}_2^{\mathrm{int}})\colon \alpha_i(\boldsymbol{x}^{\mathrm{int}},\boldsymbol{y}^{\mathrm{int}},\boldsymbol{v}_1^{\mathrm{int}}+\boldsymbol{w}_2^{\mathrm{int}},\boldsymbol{c}^{\mathrm{int}})\wedge \boldsymbol{x}^{\mathrm{int}}\neq \boldsymbol{y}^{\mathrm{int}}\wedge$$
$$\exists^{\mathsf{ram}}(\boldsymbol{x}^{\mathrm{real}},\boldsymbol{v}_1^{\mathrm{real}},\boldsymbol{v}_2^{\mathrm{real}}),(\boldsymbol{y}^{\mathrm{real}},\boldsymbol{w}_1^{\mathrm{real}},\boldsymbol{w}_2^{\mathrm{real}})\colon \beta_i(\boldsymbol{x}^{\mathrm{real}},\boldsymbol{y}^{\mathrm{real}},\boldsymbol{v}_1^{\mathrm{real}}+\boldsymbol{w}_2^{\mathrm{real}},\boldsymbol{c}^{\mathrm{real}})\wedge \boldsymbol{x}^{\mathrm{real}}\neq \boldsymbol{y}^{\mathrm{real}}.$$

This implies that $\boldsymbol{c}$ satisfies the right-hand formula of the theorem by adding the two cliques componentwise.

5 RAMSEY QUANTIFIERS IN PRESBURGER ARITHMETIC

In this section, we describe our procedure to eliminate the Ramsey quantifier if applied to an existential Presburger formula.

It is not difficult to construct Presburger-definable relations that have infinite cliques, but none that are definable in Presburger arithmetic. For example, consider the relation $R=\{(x,y)\in\mathbb{N}\times\mathbb{N}\mid y\geq 2x\}$. Then every infinite clique $A=\{a_0,a_1,\dots,\}$ with $a_0\leq a_1\leq a_2\leq\cdots$ must satisfy $a_i\geq 2^i\cdot a_0$ for every $i\geq 1$ and thus cannot be ultimately periodic (i.e., there is no $n,k\in\mathbb{N}$ such that for all $a\geq n$, we have $a\in A$ if and only if $a+k\in A$). Since a subset of $\mathbb{N}$ is Presburger-definable if and only if it is ultimately periodic, it follows that A is not Presburger-definable. Nevertheless, we show the following:

Theorem 5.1. *Given an existential Presburger formula $\varphi(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$, we can construct in polynomial time an existential Presburger formula of linear size that is equivalent to $\exists^{\mathsf{ram}}\boldsymbol{x},\boldsymbol{y}\colon\varphi(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$.*

We first assume that φ is a conjunction of the form

$$\bigwedge_{i=1}^{n}\boldsymbol{r}_i^\top\boldsymbol{x}<\boldsymbol{s}_i^\top\boldsymbol{y}+\boldsymbol{t}_i^\top\boldsymbol{z}+h_i\;\wedge\;\bigwedge_{j=1}^{m}\boldsymbol{u}_j^\top\boldsymbol{x}\approx_{e_j}^{j}\boldsymbol{v}_j^\top\boldsymbol{y}+\boldsymbol{w}_j^\top\boldsymbol{z}+d_j \tag{7}$$

where $\approx_{e_j}^{j}\in\{\equiv_{e_j},\not\equiv_{e_j}\}$. It should be noted that since Theorem 4.1 allows us to eliminate any existential quantifier under the Ramsey quantifier without introducing modulo constraints, it would even suffice to treat the case where φ has no modulo constraints. However, in practice it might be useful to be able to treat modulo constraints without first trading them in for existential quantifiers. For this reason, we describe the translation in the presence of modulo constraints.

5.1 Cliques in Terms of Profiles

Our goal is to construct an existential Presburger formula $\varphi'(\boldsymbol{z})$ so that $\varphi'(\boldsymbol{c})$ holds if and only if there exists an infinite sequence $\boldsymbol{a}_1,\boldsymbol{a}_2,\dots$ of pairwise distinct vectors for which $\varphi(\boldsymbol{a}_i,\boldsymbol{a}_j,\boldsymbol{c})$ for every $i<j$. As mentioned above, it is possible that such a sequence exists, but that none of them is definable in Presburger arithmetic. Therefore, our first step is to modify the condition "$\varphi(\boldsymbol{a}_i,\boldsymbol{a}_j,\boldsymbol{c})$ for $i<j$" into a different condition such that (i) the new condition is equivalent in terms of existence of a sequence and (ii) the new condition can always be satisfied by an arithmetic progression.

To illustrate the idea, suppose $\varphi(\boldsymbol{x},\boldsymbol{y},\boldsymbol{z})$ says that $y_1>2\cdot x_1\wedge\psi(\boldsymbol{x})$ for some Presburger formula ψ. As mentioned above, any directed clique for φ must grow exponentially in the first component. However, such a directed clique exists if and only if there exists a sequence $\boldsymbol{a}_1,\boldsymbol{a}_2,\dots$ such that $\psi(\boldsymbol{a}_1),\psi(\boldsymbol{a}_2),\dots$ and the sequence of numbers $a_1,a_2,\dots$ in the first components of $\boldsymbol{a}_1,\boldsymbol{a}_2,\dots$ grows unboundedly: Clearly, any directed clique for φ must satisfy this. Conversely, a sequence satisfying the unboundedness condition must have a subsequence with $a_j>2\cdot a_i$ for $i<j$.

These modified conditions on sequences are based on the notion of profiles. Essentially, a profile captures how in a sequence $\boldsymbol{a}_1,\boldsymbol{a}_2,\dots$ the values $\boldsymbol{r}_i^\top\boldsymbol{a}_k$ and $\boldsymbol{s}_i^\top\boldsymbol{a}_k+\boldsymbol{t}_i^\top\boldsymbol{c}+h_i$ evolve. A *profile (for φ)* is a vector in $\mathbb{Z}_\omega^{2n}$ where $\mathbb{Z}_\omega:=\mathbb{Z}\cup\{\omega\}$. Suppose $\boldsymbol{p}=(p_1,\dots,p_{2n})$. Then value p_{2i-1} being an integer means that $\boldsymbol{r}_i^\top\boldsymbol{a}_1,\boldsymbol{r}_i^\top\boldsymbol{a}_2,\dots$ is bounded from above by p_{2i-1}. If p_{2i-1} is ω, then the sequence

$\boldsymbol{r}_i^\top \boldsymbol{a}_1, \boldsymbol{r}_i^\top \boldsymbol{a}_2, \ldots$ tends to infinity. Similarly, even-indexed entries p_{2i} describe the evolution of the sequence $\boldsymbol{s}_i^\top \boldsymbol{a}_1 + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i, \boldsymbol{s}_i^\top \boldsymbol{a}_2 + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i, \ldots$.

Let us make this precise. If $\boldsymbol{p}$ is a profile and $\boldsymbol{c}$ is a vector over $\mathbb{Z}$, then a sequence $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$ of pairwise distinct vectors over $\mathbb{Z}$ is *compatible with $\boldsymbol{p}$ for $\boldsymbol{c}$* if for every $k < \ell$, we have $\boldsymbol{u}_j^\top \boldsymbol{a}_k \approx_{e_j}^{j} \boldsymbol{v}_j^\top \boldsymbol{a}_\ell + \boldsymbol{w}_j^\top \boldsymbol{c} + d_j$ and

$$\sup\{\boldsymbol{r}_i^\top \boldsymbol{a}_k \mid k = 1, 2, \ldots\} \leq p_{2i-1}, \qquad p_{2i} \leq \liminf\{\boldsymbol{s}_i^\top \boldsymbol{a}_k + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i \mid k = 1, 2, \ldots\}. \tag{8}$$

A profile $\boldsymbol{p} = (p_1, \ldots, p_{2n})$ is *admissible* if for every $i \in [1, n]$, we have $p_{2i-1} < p_{2i}$ or $p_{2i} = \omega$.

Lemma 5.2. *Let $\boldsymbol{c}$ be a vector over $\mathbb{Z}$. Then $\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c})$ if and only if there exists an admissible profile $\boldsymbol{p} \in \mathbb{Z}_\omega^{2n}$ such that there is a sequence compatible with $\boldsymbol{p}$ for $\boldsymbol{c}$.*

Proof. We begin with the "only if" direction. To ease notation, we write $f_i(\boldsymbol{x}) = \boldsymbol{r}_i^\top \boldsymbol{x}$ and $g_i(\boldsymbol{x}) = \boldsymbol{s}_i^\top \boldsymbol{x} + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i$ for $i \in [1, n]$. Suppose $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$ is a directed clique witnessing $\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c})$. First, we may assume that if for some $i \in [1, n]$, the sequence $\{f_i(\boldsymbol{a}_k) \mid k = 1, 2, \ldots\}$ is bounded from above, then for its maximum M, we have $M < g_i(\boldsymbol{a}_k)$ for every $k \geq 1$. If this is not the case, we can achieve it by removing an initial segment of $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$. Now we define the profile $\boldsymbol{p} = (p_1, p_2, \ldots, p_{2n-1}, p_{2n})$ as

$$p_{2i-1} = \sup\{f_i(\boldsymbol{a}_k) \mid k = 1, 2, \ldots\}, \qquad p_{2i} = \liminf\{g_i(\boldsymbol{a}_k) \mid k = 1, 2, \ldots\}.$$

Observe that p_{2i} cannot be $-\omega$ and thus belongs to $\mathbb{Z}_\omega$: This is because the set $\{g_i(\boldsymbol{a}_k) \mid k \geq 1\}$ is bounded from below (by $\min\{f_i(\boldsymbol{a}_1), g_i(\boldsymbol{a}_1)\}$). Then $\boldsymbol{p}$ is admissible: Otherwise, we would have $p_{2i-1} \geq p_{2i}$ and $p_{2i} \in \mathbb{Z}$, implying that there are $k < \ell$ with $f_i(\boldsymbol{a}_k) \geq p_{2i} = g_i(\boldsymbol{a}_\ell)$, which contradicts the fact that $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$ witnesses $\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c})$. Moreover, by definition of $\boldsymbol{p}$, the sequence $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$ is clearly compatible with $\boldsymbol{p}$ for $\boldsymbol{c}$.

Let us now prove the "if" direction. Let $\boldsymbol{p} \in \mathbb{Z}_\omega^{2n}$ be an admissible profile and $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$ be a sequence compatible with $\boldsymbol{p}$ for $\boldsymbol{c}$. Then, we know that for any $k < \ell$, we have $\boldsymbol{u}_j^\top \boldsymbol{a}_k \approx_{e_j}^{j} \boldsymbol{v}_j^\top \boldsymbol{a}_\ell + \boldsymbol{w}_j \boldsymbol{c} + d_j$. We claim that we can select a subsequence of $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$ such that, for every $i \in [1, n]$, we have $f_i(\boldsymbol{a}_k) < g_i(\boldsymbol{a}_\ell)$ for every $k < \ell$. It suffices to do this for each $i = 1, \ldots, n$ individually, because if for some $i \in [1, n]$, we have $f_i(\boldsymbol{a}_k) < g_i(\boldsymbol{a}_\ell)$ for every $k < \ell$, then this is still the case for any subsequence. Likewise, picking an infinite subsequence does not spoil the property of being compatible with $\boldsymbol{p}$ for $\boldsymbol{c}$.

Consider some $i \in [1, n]$. We distinguish two cases, namely whether $p_{2i} \in \mathbb{Z}$ or $p_{2i} = \omega$. First, suppose $p_{2i} \in \mathbb{Z}$. Then, since $\boldsymbol{p}$ is admissible, we have $p_{2i-1} < p_{2i}$. Now compatibility implies that $p_{2i-1} < p_{2i} \leq g_i(\boldsymbol{a}_\ell)$ for almost all ℓ. Hence, by removing some initial segment of our sequence, we can ensure that $f_i(\boldsymbol{a}_k) < g_i(\boldsymbol{a}_\ell)$ for every $k < \ell$.

Now suppose $p_{2i} = \omega$. We successively choose the elements of a subsequence $\boldsymbol{a}'_1, \boldsymbol{a}'_2, \ldots$ of $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$ such that $f_i(\boldsymbol{a}'_k) < g_i(\boldsymbol{a}'_\ell)$ for any $k < \ell$. Suppose we have already chosen $\boldsymbol{a}'_1, \ldots, \boldsymbol{a}'_h$ for some $h \geq 1$. Then the set $\{f_i(\boldsymbol{a}'_k) \mid k \in [1, h]\}$ is finite and thus bounded by some $M \in \mathbb{N}$. By compatibility of $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$, there exist infinitely many $\ell \in \mathbb{N}$ with $M < g_i(\boldsymbol{a}_\ell)$. This allows us to choose $\boldsymbol{a}'_{h+1}$ to extend our sequence.

This completes the construction of our clique witnessing $\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c})$. □

5.2 Compatibility in Terms of Matrices

Our next step is to express the existence of a sequence compatible with $\boldsymbol{p}$ for $\boldsymbol{c}$ in terms of certain inequalities. To this end, we define two matrices $\boldsymbol{A}_{\boldsymbol{p},\boldsymbol{c}}$ and $\boldsymbol{B}_{\boldsymbol{p}}$ and a vector $\boldsymbol{b}_{\boldsymbol{p},\boldsymbol{c}}$. Here, $\boldsymbol{A}_{\boldsymbol{p},\boldsymbol{c}}\boldsymbol{x} \geq \boldsymbol{b}_{\boldsymbol{p},\boldsymbol{c}}$ will express the compatibility conditions involving p_{2i-1} and p_{2i} that are integers. Thus, we define $\boldsymbol{A}_{\boldsymbol{p},\boldsymbol{c}}$ and $\boldsymbol{b}_{\boldsymbol{p},\boldsymbol{c}}$ by describing the system of inequality $\boldsymbol{A}_{\boldsymbol{p},\boldsymbol{c}}\boldsymbol{x} \geq \boldsymbol{b}_{\boldsymbol{p},\boldsymbol{c}}$. For every $i \in [1, n]$ with $p_{2i-1} \in \mathbb{Z}$, we add the inequality $\boldsymbol{r}_i^\top \boldsymbol{x} \le p_{2i-1}$. Moreover, for every $i \in [1, n]$ with $p_{2i} \in \mathbb{Z}$, we add the inequality $p_{2i} \le \boldsymbol{s}_i^\top \boldsymbol{x} + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i$.

Moreover, $\boldsymbol{B_p}$ will be used to express the unboundedness condition on the right side of Equation (8) if $p_{2i} = \omega$. Thus, for every $i \in [1, n]$ with $p_{2i} = \omega$, we add the row $\boldsymbol{s}_i^\top$ to $\boldsymbol{B_p}$. We say that a function $f\colon X \to \mathbb{Z}^\ell$ is *simultaneously unbounded* on a sequence $x_1, x_2, \ldots \in X$ if for every $k \in \mathbb{N}$, we have $f(x_j) \ge (k, \ldots, k)$ for almost all j. Now observe the following:

Lemma 5.3. *Let $\boldsymbol{c}$ be a vector and $\boldsymbol{p} \in \mathbb{Z}_\omega^{2n}$ be a profile. Then there exists a sequence that is compatible with $\boldsymbol{p}$ for $\boldsymbol{c}$ if and only if there exists a sequence $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$ of pairwise distinct vectors such that (i) $\boldsymbol{u}_j^\top \boldsymbol{a}_k \approx_{e_j}^j \boldsymbol{v}_j^\top \boldsymbol{a}_\ell + \boldsymbol{w}_j^\top \boldsymbol{c} + d_j$ for every $j \in [1, m]$ and $k < \ell$ and (ii) $\boldsymbol{A_{p,c}} \boldsymbol{a}_k \ge \boldsymbol{b_{p,c}}$ for every $k \ge 1$ and (iii) $\boldsymbol{B_p}$ is simultaneously unbounded on $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$.*

5.3 Arithmetic Progressions

The last key step is to show that there exists a sequence compatible with $\boldsymbol{p}$ if and only if there exists such a sequence of the form $\boldsymbol{a}_0 + \boldsymbol{a}, \boldsymbol{a}_0 + 2\boldsymbol{a}, \boldsymbol{a}_0 + 3\boldsymbol{a}, \ldots$. This will allow us to express existence of a sequence by the existence of suitable vectors $\boldsymbol{a}_0$ and $\boldsymbol{a}$.

Lemma 5.4. *Let $\boldsymbol{c}$ be a vector and $\boldsymbol{p} \in \mathbb{Z}_\omega^{2n}$ be a profile. There exists a sequence compatible with $\boldsymbol{p}$ for $\boldsymbol{c}$ if and only if there are vectors $\boldsymbol{a}_0, \boldsymbol{a}$ over $\mathbb{Z}$ with $\boldsymbol{a} \ne \boldsymbol{0}$ such that for all $j \in [1, m]$,*

$$\begin{aligned} \boldsymbol{A_{p,c}} \boldsymbol{a}_0 \ge \boldsymbol{b_{p,c}},\ \boldsymbol{A_{p,c}} \boldsymbol{a} \ge \boldsymbol{0}, &\qquad \boldsymbol{B_p} \boldsymbol{a} \gg \boldsymbol{0}, \\ \boldsymbol{u}_j^\top \boldsymbol{a}_0 \approx_{e_j}^j \boldsymbol{v}_j^\top (\boldsymbol{a}_0 + \boldsymbol{a}) + \boldsymbol{w}_j^\top \boldsymbol{c} + d_j, &\qquad \boldsymbol{u}_j^\top \boldsymbol{a} \equiv_{e_j} \boldsymbol{v}_j^\top \boldsymbol{a} \equiv_{e_j} 0. \end{aligned}$$

Proof. We begin with the "if" direction. Suppose there are vectors $\boldsymbol{a}_0$ and $\boldsymbol{a}$ as described. Then we claim that the sequence $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$ with $\boldsymbol{a}_k = \boldsymbol{a}_0 + k \cdot \boldsymbol{a}$ is compatible with $\boldsymbol{p}$ for $\boldsymbol{c}$. We use Lemma 5.3 to show this. First note that since $\boldsymbol{a} \ne \boldsymbol{0}$, the $\boldsymbol{a}_k$ are pairwise distinct. It is clear that the sequence satisfies conditions (i) and (ii) of Lemma 5.3. Condition (iii) holds as well, because in the vector $\boldsymbol{B_p}(k \cdot \boldsymbol{a})$, every entry is at least k. Thus, $\boldsymbol{B_p}$ is simultaneously unbounded on $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$.

For the "only if" direction, suppose $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$ is a sequence of pairwise distinct vectors that satisfies the conditions in Lemma 5.3. Since there are only finitely many possible remainders modulo e_j of the expressions $\boldsymbol{u}_j^\top \boldsymbol{a}_k$ and $\boldsymbol{v}_j^\top \boldsymbol{a}_k$, we can pick a subsequence such that for each $j \in [1, m]$, the maps $k \mapsto \boldsymbol{u}_j^\top \boldsymbol{a}_k$ and $k \mapsto \boldsymbol{v}_j^\top \boldsymbol{a}_k$ are constant modulo e_j. In the second step, we notice that since $\boldsymbol{A_{p,c}} \boldsymbol{a}_k \ge \boldsymbol{b_{p,c}}$ for each $k \ge 1$, the sequence $\boldsymbol{A_{p,c}} \boldsymbol{a}_1, \boldsymbol{A_{p,c}} \boldsymbol{a}_2, \ldots$ cannot contain an infinite strictly descending chain in any component. Thus, by Ramsey's theorem, we may pick a subsequence so that $\boldsymbol{A_{p,c}} \boldsymbol{a}_1 \le \boldsymbol{A_{p,c}} \boldsymbol{a}_2 \le \cdots$. Note that passing to subsequences does not spoil the conditions of Lemma 5.3. Thus, $\boldsymbol{B_p}$ is still simultaneously unbounded on $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$. This allows us to pick a subsequence so that also $\boldsymbol{B_p} \boldsymbol{a}_1 \ll \boldsymbol{B_p} \boldsymbol{a}_2 \ll \cdots$. Therefore, if we set $\boldsymbol{a}_0 := \boldsymbol{a}_1$ and $\boldsymbol{a} := \boldsymbol{a}_2 - \boldsymbol{a}_1$, then $\boldsymbol{a}_0$ and $\boldsymbol{a}$ are as desired. □

5.4 Construction of the Formula

We are now ready to prove Theorem 5.1 in the general case, i.e., $\varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ is an arbitrary existential Presburger formula. By Theorem 4.1 we can assume that φ is quantifier-free. By moving all negations inwards to the atoms and possibly negating those, we may further assume that φ is a positive Boolean combination of inequality atoms $\alpha_i := \boldsymbol{r}_i^\top \boldsymbol{x} < \boldsymbol{s}_i^\top \boldsymbol{y} + \boldsymbol{t}_i^\top \boldsymbol{z} + h_i$ for $i \in [1, n]$ and modulo constraint atoms $\beta_j := \boldsymbol{u}_j^\top \boldsymbol{x} \approx_{e_j}^j \boldsymbol{v}_j^\top \boldsymbol{y} + \boldsymbol{w}_j^\top \boldsymbol{z} + d_j$ with $\approx_{e_j}^j \in \{\equiv_{e_j}, \not\equiv_{e_j}\}$ for $j \in [1, m]$. (Note that in Presburger arithmetic equality can be expressed by a conjunction of two strict inequalities.)

The key idea is to guess (using existentially quantified variables) a subset of the atoms in φ and check that (i) satisfying those atoms makes φ true and (ii) for the conjunction of those atoms, there exists a directed clique. Note that there are only finitely many conjunctions of atoms (from φ) and

φ is equivalent to the disjunction over these conjunctions. Thus, by Ramsey's theorem, there exists a directed clique for φ if and only if there exists one for some conjunction of atoms. Condition (i) is easy to state. To check (ii), we then require the conditions of Lemma 5.4 to be satisfied for our conjunction of atoms.

For each atom α_i, we introduce a variable $q_i^<$, and for each atom β_j, we introduce a variable $q_j^\approx$. To check that (i) holds, we use the formula φ', that is obtained from φ by replacing each α_i with $q_i^< = 1$ and each β_j with $q_j^\approx = 1$, and adding the restrictions $q_i^< = 0 \vee q_i^< = 1$ and $q_j^\approx = 0 \vee q_j^\approx = 1$. Now, φ is equivalent to

$$\psi := \exists \boldsymbol{q}^<, \boldsymbol{q}^\approx \colon \varphi' \wedge \bigwedge_{i=1}^{n} (q_i^< = 1 \to \alpha_i) \wedge \bigwedge_{j=1}^{m} (q_j^\approx = 1 \to \beta_j).$$

Let us now construct the formula for condition (ii) above. To this end, we build a formula γ_i that states all conditions of Lemma 5.4 that stem from the atom α_i. For $i \in [1, n]$ and fresh variables $p_{2i-1}, p_{2i}, \boldsymbol{x}_0, \boldsymbol{x}$ let

$$\begin{aligned} \gamma_i := {} & (p_{2i-1} < \omega \to (\boldsymbol{r}_i^\top \boldsymbol{x}_0 \le p_{2i-1} \wedge \boldsymbol{r}_i^\top \boldsymbol{x} \le 0)) \wedge \\ & (p_{2i} < \omega \to (p_{2i} \le \boldsymbol{s}_i^\top \boldsymbol{x}_0 + \boldsymbol{t}_i^\top \boldsymbol{z} + h_i \wedge \boldsymbol{s}_i^\top \boldsymbol{x} \ge 0)) \wedge \\ & (p_{2i} = \omega \to \boldsymbol{s}_i^\top \boldsymbol{x} > 0) \end{aligned}$$

and for all $j \in [1, m]$ let

$$\delta_j := \boldsymbol{u}_j^\top \boldsymbol{x}_0 \approx_{e_j}^{j} \boldsymbol{v}_j^\top (\boldsymbol{x}_0 + \boldsymbol{x}) + \boldsymbol{w}_j^\top \boldsymbol{z} + d_j \wedge \boldsymbol{u}_j^\top \boldsymbol{x} \equiv_{e_j} 0 \wedge \boldsymbol{v}_j^\top \boldsymbol{x} \equiv_{e_j} 0.$$

Here, $p_\ell < \omega$ and $p_\ell = \omega$ is shorthand notation for $\omega_\ell = 0$ and $\omega_\ell = 1$, respectively, where ω_ℓ is a fresh variable associated with p_ℓ that is restricted to values from $\{0, 1\}$. Thus, from now on we implicitly quantify $\boldsymbol{\omega}$ when $\boldsymbol{p}$ is quantified. The following requires $\boldsymbol{p}$ to be an admissible profile:

$$\theta := \bigwedge_{i=1}^{n} (p_{2i-1} < \omega \wedge p_{2i-1} < p_{2i} \vee p_{2i} = \omega).$$

Then we claim that $\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \psi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ is equivalent to

$$\chi := \exists \boldsymbol{q}^<, \boldsymbol{q}^\approx, \boldsymbol{p}, \boldsymbol{x}_0, \boldsymbol{x} \colon \varphi' \wedge \theta \wedge \boldsymbol{x} \neq \boldsymbol{0} \wedge \bigwedge_{i=1}^{n} (q_i^< = 1 \to \gamma_i) \wedge \bigwedge_{j=1}^{m} (q_j^\approx = 1 \to \delta_j).$$

We show that for any valuation $\boldsymbol{c} \in \mathbb{Z}^{|\boldsymbol{z}|}$ of $\boldsymbol{z}$ we have $\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \psi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c})$ if and only if $\chi(\boldsymbol{c})$. For an assignment ν of the $q_i^<, q_j^\approx$ to $\{0, 1\}$ let $I_\nu := \{i \in [1, n] \mid \nu(q_i^<) = 1\}$ and $J_\nu := \{j \in [1, m] \mid \nu(q_j^\approx) = 1\}$. By Ramsey's theorem we have $\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \psi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c})$ if and only if there is an assignment ν of the $q_i^<, q_j^\approx$ satisfying φ' such that $\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \bigwedge_{i \in I_\nu} \alpha_i(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c}) \wedge \bigwedge_{j \in J_\nu} \beta_j(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c})$. By Lemmas 5.2 and 5.4 this formula is equivalent to $\exists \boldsymbol{p}, \boldsymbol{x}_0, \boldsymbol{x} \colon \theta \wedge \boldsymbol{x} \neq \boldsymbol{0} \wedge \bigwedge_{i \in I_\nu} \gamma_i(\boldsymbol{p}, \boldsymbol{x}_0, \boldsymbol{x}, \boldsymbol{c}) \wedge \bigwedge_{j \in J_\nu} \delta_j(\boldsymbol{x}_0, \boldsymbol{x}, \boldsymbol{c})$ which in turn holds for some assignment ν of the $q_i^<, q_j^\approx$ satisfying φ' if and only if $\chi(\boldsymbol{c})$.

6 RAMSEY QUANTIFIERS IN LINEAR REAL ARITHMETIC

In this section, we describe our procedure to eliminate the Ramsey quantifier if applied to an existential LRA formula. At the end of the section, we mention a version of this result for the structure $\langle \mathbb{Q}; +, <, 1, 0 \rangle$ (Theorem 6.5).

THEOREM 6.1. *Given an existential formula $\varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ in LRA, we can construct in polynomial time an existential formula in LRA of linear size that is equivalent to $\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$.*

Similar to the integer case, we first assume that φ is a conjunction of the following form:

$$\bigwedge_{i=1}^{n} \boldsymbol{r}_i^\top \boldsymbol{x} < \boldsymbol{s}_i^\top \boldsymbol{y} + \boldsymbol{t}_i^\top z + h_i \;\wedge\; \bigwedge_{j=1}^{m} \boldsymbol{u}_j^\top \boldsymbol{x} = \boldsymbol{v}_j^\top \boldsymbol{y} + \boldsymbol{w}_j^\top z + d_j \tag{9}$$

where $\boldsymbol{r}_i, \boldsymbol{s}_i, \boldsymbol{t}_i, \boldsymbol{u}_j, \boldsymbol{v}_j, \boldsymbol{w}_j \in \mathbb{Q}^d$ for $d \geq 1$ and $h_i, d_j \in \mathbb{Q}$.

6.1 Cliques in Terms of Profiles

We now define the notion of a profile with a similar purpose as in the integer case. In the real case, these carry more information: In the case of Presburger arithmetic, it is enough to guess whether a particular function grows or has a particular upper bound. Here, it is possible that a function grows strictly, but it still bounded, because it converges. For example, if $\varphi(x, y)$ says that $x < y$ and $x \leq 1$, then a clique must be a strictly ascending sequence of numbers ≤ 1.

A *profile* (for φ) is a tuple $\boldsymbol{p} = (\rho, \sigma, t_\rho, t_\sigma)$ of functions where $\rho, \sigma \colon \{1, \ldots, n\} \to \mathbb{R} \cup \{-\omega, \omega\}$ and $t_\rho, t_\sigma \colon \{1, \ldots, n\} \to \{-\omega, -1, 0, 1, \omega\}$. For a sequence $(\boldsymbol{a}_k)_{k\geq 1}$ in $\mathbb{R}^d$ let $\boldsymbol{\rho}_i := (\boldsymbol{r}_i^\top \boldsymbol{a}_k)_{k\geq 1}$ and $\boldsymbol{\sigma}_i := (\boldsymbol{s}_i^\top \boldsymbol{a}_k)_{k\geq 1}$. We say that a sequence $(\boldsymbol{a}_k)_{k\geq 1}$ of pairwise distinct vectors is *compatible* with $\boldsymbol{p}$ if $\rho(i)$ and $\sigma(i)$ are the real values to which the sequences $\boldsymbol{\rho}_i$ and $\boldsymbol{\sigma}_i$ converge or ω (resp. $-\omega$) if the corresponding sequence is strictly increasing (resp. decreasing) and diverges to ∞ (resp. $-\infty$) and the functions t_ρ and t_σ describe the type of convergence where type 0 means that the corresponding sequence is constant, type 1 (resp. -1) means that it is strictly increasing (resp. decreasing) and converges from below (resp. above), and the type is ω (resp. $-\omega$) in the divergent case. A profile $\boldsymbol{p}$ is $\boldsymbol{c}$*-admissible* for a vector $\boldsymbol{c} \in \mathbb{R}^d$ if for all $i \in \{1, \ldots, n\}$ we have

- $\sigma(i) \neq -\omega$ and if $\rho(i) = \omega$, then $\sigma(i) = \omega$,
- $\rho(i) < \sigma(i) + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i$ if either $t_\rho(i) \in \{-1, 0\}$ and $t_\sigma(i) \in \{0, 1\}$ or $t_\rho(i) = -1$ and $t_\sigma(i) = -1$,
- $\rho(i) \leq \sigma(i) + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i$ if either $t_\rho(i) = 0$ and $t_\sigma(i) = -1$ or $t_\rho(i) = 1$.

We say that a sequence $(\boldsymbol{a}_k)_{k\geq 1}$ satisfies the equality constraints (of φ) for $\boldsymbol{c} \in \mathbb{R}^d$ if $\boldsymbol{u}_j^\top \boldsymbol{a}_k = \boldsymbol{v}_j^\top \boldsymbol{a}_\ell + \boldsymbol{w}_j^\top \boldsymbol{c} + d_j$ for all $j \in \{1, \ldots, m\}$ and $k < \ell$.

Lemma 6.2. *Let $\boldsymbol{c} \in \mathbb{R}^d$. Then $\exists^{\mathrm{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c})$ if and only if there exists a $\boldsymbol{c}$-admissible profile $\boldsymbol{p}$ such that there is a sequence compatible with $\boldsymbol{p}$ that satisfies the equality constraints for $\boldsymbol{c}$.*

Proof. We first show the "only if" direction. Let $(\boldsymbol{a}_k)_{k\geq 1}$ be a clique witnessing $\exists^{\mathrm{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c})$. For all $i \in \{1, \ldots, n\}$ consider the sequence $\boldsymbol{\rho}_i$. By the Bolzano-Weierstrass theorem if $\boldsymbol{\rho}_i$ is bounded, we can replace $(\boldsymbol{a}_k)_{k\geq 1}$ by an infinite subsequence such that $\boldsymbol{\rho}_i$ converges against a real value $r_i \in \mathbb{R}$. By restricting further to an infinite subsequence, we have that $\boldsymbol{\rho}_i$ is either constant, strictly increasing, or strictly decreasing. Thus, we set $\rho(i) := r_i$ and $t_\rho(i)$ to 0, 1, or -1 depending on whether $\boldsymbol{\rho}_i$ is constant, increasing, or decreasing. If $\boldsymbol{\rho}_i$ is unbounded, we replace $(\boldsymbol{a}_k)_{k\geq 1}$ by an infinite subsequence such that $\boldsymbol{\rho}_i$ is strictly increasing if it is unbounded above and strictly decreasing if it is unbounded below. Then we set $\rho(i)$ and $t_\rho(i)$ to ω or $-\omega$ depending on whether $\boldsymbol{\rho}_i$ is increasing or decreasing. Similarly, we can define $\sigma(i)$ and $t_\sigma(i)$ by considering the sequence $\boldsymbol{\sigma}_i$. Thus, there is a sequence $(\boldsymbol{a}_k)_{k\geq 1}$ that is compatible with the profile $\boldsymbol{p} := (\rho, \sigma, t_\rho, t_\sigma)$. Since $(\boldsymbol{a}_k)_{k\geq 1}$ still satisfies the equality constraints for $\boldsymbol{c}$, it remains to show that $\boldsymbol{p}$ is $\boldsymbol{c}$-admissible. First observe that $\sigma(i) \neq -\omega$ since $\boldsymbol{\sigma}_i$ is bounded from below by $\min\{\boldsymbol{r}_i^\top \boldsymbol{a}_1, \boldsymbol{s}_i^\top \boldsymbol{a}_1\}$. If $\rho(i) = \omega$, then also $\sigma(i) = \omega$ since otherwise there were $k < \ell$ such that $\boldsymbol{r}_i^\top \boldsymbol{a}_k \geq \boldsymbol{s}_i^\top \boldsymbol{a}_\ell + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i$. With a similar reasoning we can show that if $t_\rho(i) \in \{-1, 0\}$ and $t_\sigma(i) \in \{0, 1\}$, then $\rho(i) < \sigma(i) + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i$, if $t_\rho(i) = t_\sigma(i) = -1$, then $\rho(i) < \sigma(i) + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i$, and if either $t_\rho(i) = 0$ and $t_\sigma(i) = -1$ or $t_\rho(i) = 1$, then $\rho(i) \leq \sigma(i) + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i$.

We now turn to the "if" direction. Let $\boldsymbol{p} = (\rho, \sigma, t_\rho, t_\sigma)$ be a $\boldsymbol{c}$-admissible profile and $(\boldsymbol{a}_k)_{k\geq 1}$ be a sequence compatible with $\boldsymbol{p}$ that satisfies the equality constraints for $\boldsymbol{c}$. We successively restrict for each $i \in \{1,\dots,n\}$ to a subsequence such that $\boldsymbol{r}_i^\top \boldsymbol{a}_k < \boldsymbol{s}_i^\top \boldsymbol{a}_\ell + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i$ for all $k < \ell$. We construct the subsequence inductively. Suppose we already constructed the subsequence $\boldsymbol{a}_{i_1},\dots,\boldsymbol{a}_{i_h}$ such that $\boldsymbol{r}_i^\top \boldsymbol{a}_{i_k} < \boldsymbol{s}_i^\top \boldsymbol{a}_{i_\ell} + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i$ for all $1 \le k < \ell \le h$ and the set $L_h := \{\ell > i_h \mid \forall 1 \le k \le h\colon \boldsymbol{r}_i^\top \boldsymbol{a}_{i_k} < \boldsymbol{s}_i^\top \boldsymbol{a}_\ell + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i\}$ is infinite. Let $i_{h+1} := \min(L_h)$. If $\rho(i) < \sigma(i) + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i$, then clearly there is an infinite subset L of L_h such that $\boldsymbol{r}_i^\top \boldsymbol{a}_{i_{h+1}} < \boldsymbol{s}_i^\top \boldsymbol{a}_\ell + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i$ for all $\ell \in L \setminus \{i_{h+1}\}$. If $\rho(i) = \sigma(i) + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i$, then by definition of $\boldsymbol{c}$-admissibility we have that either $t_\rho(i) = 0$ and $t_\sigma(i) = -1$ or $t_\sigma(i) = 1$, or $\rho(i) = \sigma(i) = \omega$. In all of these cases we can find an infinite subset L of L_h such that $\boldsymbol{r}_i^\top \boldsymbol{a}_{i_{h+1}} < \boldsymbol{s}_i^\top \boldsymbol{a}_\ell + \boldsymbol{t}_i^\top \boldsymbol{c} + h_i$ for all $\ell \in L \setminus \{i_{h+1}\}$. Thus, we can extend the subsequence by $\boldsymbol{a}_{i_{h+1}}$ where the set L_{h+1} is infinite since it contains L. Finally, note that passing to subsequences does not spoil the satisfaction of the equality constraints for $\boldsymbol{c}$. Thus, the constructed subsequence is an infinite clique witnessing $\exists^{\mathsf{ram}} \boldsymbol{x}, \boldsymbol{y}\colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c})$. □

6.2 A General Form of Cliques

In the case of Presburger arithmetic, a key insight was that if there exists a clique compatible with a profile, then there exists one of the form $\boldsymbol{a}_0, \boldsymbol{a}_0 + \boldsymbol{a}, \boldsymbol{a}_0 + 2\cdot\boldsymbol{a}, \dots$. The case of reals is more involved in this regard: There are profiles with which no arithmetic progression is compatible.

For example, consider the profile that specifies that in the first component, the numbers must increase strictly in each step and tend to infinity. In the second component, the numbers must also increase strictly, but are bounded from above by 1. A sequence compatible with this would be $(\frac{1}{2}, 1), (\frac{3}{4}, 2), (\frac{4}{5}, 3), \dots$. However, such a sequence cannot be of the form $\boldsymbol{a}_0, \boldsymbol{a}_0 + \boldsymbol{a}, \boldsymbol{a}_0 + 2\cdot\boldsymbol{a}, \dots$: The entry in the first component of $\boldsymbol{a}$ would have to be positive; but if it is, then the first component also tends to infinity. Instead, we look for cliques of the form $\boldsymbol{a}_1, \boldsymbol{a}_2, \dots$ with

$$\boldsymbol{a}_k = \boldsymbol{a} - \tfrac{1}{k}\boldsymbol{d}_c + k\boldsymbol{d}_\infty \tag{10}$$

for some vectors $\boldsymbol{a}$, $\boldsymbol{d}_c$, and $\boldsymbol{d}_\infty$. Here the vector $\boldsymbol{d}_c$ realizes the convergence behavior: By subtracting smaller and smaller fractions of it, the part $\boldsymbol{a} - \frac{1}{k}\boldsymbol{d}_c$ converges to $\boldsymbol{a}$. Moreover, the vector $\boldsymbol{d}_\infty$ realizes divergence to ∞ or $-\infty$: By adding larger and larger multiples of it, we can make sure certain linear functions on $\boldsymbol{a}_k$ grow unboundedly.

We will later formulate necessary conditions on vectors $\boldsymbol{a}$, $\boldsymbol{d}_c$, and $\boldsymbol{d}_\infty$ such that the sequence (10) is compatible with a profile and satisfies the equality constraints of φ. We will then show the converse in Lemma 6.4: If there is a compatible sequence, then there is one of the form (10).

6.3 Extracting $\boldsymbol{a}$ and $\boldsymbol{d}_\infty$

Before we formulate the necessary conditions, we present the key lemma that will yield the existence of $\boldsymbol{a}$ and $\boldsymbol{d}_\infty$. Suppose that we are looking for a sequence $\boldsymbol{a}_1, \boldsymbol{a}_2, \dots$ in $\mathbb{R}^d$ where for some linear maps $\boldsymbol{A}\colon \mathbb{R}^d \to \mathbb{R}^m$ and $\boldsymbol{B}\colon \mathbb{R}^d \to \mathbb{R}^n$, the sequence $\boldsymbol{A}\boldsymbol{a}_1, \boldsymbol{A}\boldsymbol{a}_2, \dots$ converges to some $\boldsymbol{v} \in \mathbb{R}^d$ and the sequence $\boldsymbol{B}\boldsymbol{a}_1, \boldsymbol{B}\boldsymbol{a}_2, \dots$ is simultaneously unbounded. If we want to show that there exists such a sequence of the form Equation (10), then we need $\boldsymbol{a}$ and $\boldsymbol{d}_\infty$ to satisfy (i) $\boldsymbol{A}\boldsymbol{a} = \boldsymbol{v}$, (ii) $\boldsymbol{A}\boldsymbol{d}_\infty = \boldsymbol{0}$ and (iii) $\boldsymbol{B}\boldsymbol{d}_\infty \gg \boldsymbol{0}$. Indeed, we need $\boldsymbol{A}\boldsymbol{d}_\infty = \boldsymbol{0}$, because if $\boldsymbol{A}\boldsymbol{d}_\infty$ had a non-zero component, the sequence $k \mapsto \boldsymbol{A}(a - \frac{1}{k}\boldsymbol{d}_c + k\boldsymbol{d}_\infty)$ would diverge in that component. Moreover, if $\boldsymbol{A}\boldsymbol{d}_\infty = \boldsymbol{0}$, then $k \mapsto \boldsymbol{A}(\boldsymbol{a} - \frac{1}{k}\boldsymbol{d}_c + k\boldsymbol{d}_\infty) = \boldsymbol{A}(\boldsymbol{a} - \frac{1}{k}\boldsymbol{d}_c)$ converges to $\boldsymbol{A}\boldsymbol{a}$, meaning we need $\boldsymbol{A}\boldsymbol{a} = \boldsymbol{v}$. Finally, the map $\boldsymbol{B}$ is simultaneously unbounded on the sequence $k \mapsto \boldsymbol{a} - \frac{1}{k}\boldsymbol{d}_c + k\boldsymbol{d}_\infty$ if and only if $\boldsymbol{B}\boldsymbol{d}_\infty \gg \boldsymbol{0}$.

The following lemma yields vectors $\boldsymbol{a}$ and $\boldsymbol{d}_\infty$ that satisfy these conditions. For the proof we refer to the full version of this paper [Bergsträßer et al. 2023a].

Lemma 6.3. *Let $\boldsymbol{A}\colon \mathbb{R}^d \to \mathbb{R}^m$ and $\boldsymbol{B}\colon \mathbb{R}^d \to \mathbb{R}^n$ be linear maps. Let $\boldsymbol{a}_1, \boldsymbol{a}_2, \dots \in \mathbb{R}^d$ be a sequence such that $\boldsymbol{A}\boldsymbol{a}_1, \boldsymbol{A}\boldsymbol{a}_2, \dots$ converges against $\boldsymbol{v} \in \mathbb{R}^m$ and $\boldsymbol{B}$ is simultaneously unbounded on $\boldsymbol{a}_1, \boldsymbol{a}_2, \dots$. Then there exist (1) $\boldsymbol{a} \in \mathbb{R}^d$ with $\boldsymbol{A}\boldsymbol{a} = \boldsymbol{v}$ and (2) $\boldsymbol{d}_\infty \in \mathbb{R}^d$ with $\boldsymbol{A}\boldsymbol{d}_\infty = \boldsymbol{0}$ and $\boldsymbol{B}\boldsymbol{d}_\infty \gg \boldsymbol{0}$.*

6.4 Compatibility in Terms of Inequalities

We are now ready to describe the necessary and sufficient conditions for the vectors $\boldsymbol{a}$, $\boldsymbol{d}_c$, and $\boldsymbol{d}_\infty$. We define matrices and vectors to describe systems of linear (in)equalities that are needed to express the compatibility conditions. Let $\boldsymbol{p}$ be a profile and define the following inequalities.

Limit values Let $\boldsymbol{L_p}$ be a matrix and $\boldsymbol{\ell_p}$ be a vector such that $\boldsymbol{L_p x} = \boldsymbol{\ell_p}$ if and only if

$$\boldsymbol{r}_i^\top \boldsymbol{x} = \rho(i) \qquad \text{for each } i \text{ with } t_\rho(i) \in \{-1, 1\}$$
$$\boldsymbol{s}_i^\top \boldsymbol{x} = \sigma(i) \qquad \text{for each } i \text{ with } t_\sigma(i) \in \{-1, 1\}$$

Then, as discussed above, our vectors need to satisfy $\boldsymbol{L_p a} = \boldsymbol{\ell_p}$ and $\boldsymbol{L_p d}_\infty = \boldsymbol{0}$.

Constant values Let $\boldsymbol{C_p}$ be a matrix and $\boldsymbol{c_p}$ be a vector such that $\boldsymbol{C_p x} = \boldsymbol{c_p}$ if and only if

$$\boldsymbol{r}_i^\top \boldsymbol{x} = \rho(i) \qquad \text{for each } i \text{ with } t_\rho(i) = 0$$
$$\boldsymbol{s}_i^\top \boldsymbol{x} = \sigma(i) \qquad \text{for each } i \text{ with } t_\sigma(i) = 0$$

Since components i with $t_\rho(i) = 0$ (resp. $t_\sigma(i) = 0$) are those where $\boldsymbol{r}_i^\top \boldsymbol{a}_1, \boldsymbol{r}^\top \boldsymbol{a}_2, \ldots$ (resp. $\boldsymbol{s}_i^\top \boldsymbol{a}_1, \boldsymbol{s}^\top \boldsymbol{a}_2, \ldots$) is constant, our vectors clearly need to satisfy $\boldsymbol{C_p d}_c = \boldsymbol{0}$ and $\boldsymbol{C_p d}_\infty = \boldsymbol{0}$.

Convergence Let $\boldsymbol{D_p}$ be a matrix such that $\boldsymbol{D_p x} \gg \boldsymbol{0}$ if and only if

$$\boldsymbol{r}_i^\top \boldsymbol{x} > 0 \text{ (resp.} < 0) \qquad \text{for each } i \text{ with } t_\rho(i) = 1 \text{ (resp.} = -1),$$
$$\boldsymbol{s}_i^\top \boldsymbol{x} > 0 \text{ (resp.} < 0) \qquad \text{for each } i \text{ with } t_\sigma(i) = 1 \text{ (resp.} = -1)$$

Since the components i with $t_\rho(i) = 1$ (resp. $t_\rho(i) < 0$) are those where $\boldsymbol{r}_i^\top \boldsymbol{a}_1, \boldsymbol{r}_i^\top \boldsymbol{a}_2, \ldots$ converges to a real number from below (resp. from above), and similarly for $\boldsymbol{s}_i^\top \boldsymbol{a}_1, \boldsymbol{s}_i^\top \boldsymbol{a}_2, \ldots$, we must have $\boldsymbol{D_p d}_c \gg \boldsymbol{0}$.

Unboundedness Let $\boldsymbol{U_p}$ be a matrix such that $\boldsymbol{U_p x} \gg \boldsymbol{0}$ if and only if

$$\boldsymbol{r}_i^\top \boldsymbol{x} > 0 \text{ (resp.} < 0) \qquad \text{for each } i \text{ with } t_\rho(i) = \omega \text{ (resp.} = -\omega)$$
$$\boldsymbol{s}_i^\top \boldsymbol{x} > 0 \text{ (resp.} < 0) \qquad \text{for each } i \text{ with } t_\sigma(i) = \omega \text{ (resp.} = -\omega)$$

Since the components i with $t_\rho(i) = \omega$ are those where $\boldsymbol{r}_i^\top \boldsymbol{a}_1, \boldsymbol{r}_i^\top \boldsymbol{a}_2, \ldots$ diverges to ∞ (and analogous relationships hold for $t_\rho(i) = -\omega$ and for $t_\sigma(i)$), we must have $\boldsymbol{U_p d}_\infty \gg \boldsymbol{0}$.

Let us now formally provide a list of necessary and sufficient conditions on $\boldsymbol{a}$, $\boldsymbol{d}_c$, and $\boldsymbol{d}_\infty$ for the existence of a sequence compatible with $\boldsymbol{p}$ that satisfies the equality constraints for $\boldsymbol{c}$.

Lemma 6.4. *Let $\boldsymbol{c} \in \mathbb{R}^d$ and $\boldsymbol{p}$ be a profile. Then there exists a sequence compatible with $\boldsymbol{p}$ that satisfies the equality constraints for $\boldsymbol{c}$ if and only if there are vectors $\boldsymbol{a}, \boldsymbol{d}_c, \boldsymbol{d}_\infty \in \mathbb{R}^d$ with $\boldsymbol{d}_c \neq \boldsymbol{0}$ with*

(1) $\boldsymbol{L_p a} = \boldsymbol{\ell_p}$, $\boldsymbol{C_p a} = \boldsymbol{c_p}$,
(2) $\boldsymbol{D_p d}_c \gg \boldsymbol{0}$, $\boldsymbol{C_p d}_c = \boldsymbol{0}$,
(3) $\boldsymbol{L_p d}_\infty = \boldsymbol{0}$, $\boldsymbol{C_p d}_\infty = \boldsymbol{0}$, $\boldsymbol{U_p d}_\infty \gg \boldsymbol{0}$, and
(4) $\boldsymbol{u}_j^\top \boldsymbol{d}_c = \boldsymbol{u}_j^\top \boldsymbol{d}_\infty = 0$, $\boldsymbol{v}_j^\top \boldsymbol{d}_c = \boldsymbol{v}_j^\top \boldsymbol{d}_\infty = 0$, $(\boldsymbol{u}_j^\top - \boldsymbol{v}_j^\top)\boldsymbol{a} = \boldsymbol{w}_j^\top \boldsymbol{c} + d_j$ for all $j \in \{1, \ldots, m\}$.

Proof. We start with the "only if" direction. Let $(\boldsymbol{a}_k)_{k \geq 1}$ be a sequence compatible with $\boldsymbol{p}$ that satisfies the equality constraints for $\boldsymbol{c}$. First observe that the equality constraints imply that $\boldsymbol{u}_j^\top \boldsymbol{a}_k = \boldsymbol{u}_j^\top \boldsymbol{a}_\ell$, $\boldsymbol{v}_j^\top \boldsymbol{a}_k = \boldsymbol{v}_j^\top \boldsymbol{a}_\ell$, and $(\boldsymbol{u}_j^\top - \boldsymbol{v}_j^\top)\boldsymbol{a}_k = \boldsymbol{w}_j^\top \boldsymbol{c} + d_j$ for all $2 \leq k < \ell$. Thus, by removing the first vector of the sequence we can assume that $(\boldsymbol{a}_k)_{k \geq 1}$ fulfills this property already for $1 \leq k < \ell$. For $\boldsymbol{d}_c$ we choose $\boldsymbol{a}_2 - \boldsymbol{a}_1$ where $\boldsymbol{d}_c \neq \boldsymbol{0}$ since $\boldsymbol{a}_1 \neq \boldsymbol{a}_2$. This fulfills (2) since the sequences $\boldsymbol{\rho}_i$ and $\boldsymbol{\sigma}_i$ are strictly increasing/decreasing if $t_\rho, t_\sigma \in \{-1, 1\}$ and constant if $t_\rho = t_\sigma = 0$. Moreover, $\boldsymbol{d}_c$ fulfills (4) since $\boldsymbol{u}_j^\top \boldsymbol{d}_c = \boldsymbol{u}_j^\top \boldsymbol{a}_2 - \boldsymbol{u}_j^\top \boldsymbol{a}_1 = 0$ and $\boldsymbol{v}_j^\top \boldsymbol{d}_c = \boldsymbol{v}_j^\top \boldsymbol{a}_2 - \boldsymbol{v}_j^\top \boldsymbol{a}_1 = 0$. Let $\boldsymbol{A}$ be the matrix obtained by concatenating $\boldsymbol{L_p}$ and $\boldsymbol{C_p}$ vertically and adding the rows $\boldsymbol{u}_j^\top$, $\boldsymbol{v}_j^\top$, and $\boldsymbol{u}_j^\top - \boldsymbol{v}_j^\top$ for all $j \in \{1, \ldots, m\}$. In parallel, we define the vector $\boldsymbol{v}$ as the vertical concatenation of $\boldsymbol{\ell_p}$ and $\boldsymbol{c_p}$ extended by the entry

$\boldsymbol{u}_j^\top \boldsymbol{a}_1$ in the row of $\boldsymbol{u}_j^\top$, the entry $\boldsymbol{v}_j^\top \boldsymbol{a}_1$ in the row of $\boldsymbol{v}_j^\top$, and the entry $\boldsymbol{w}_j^\top \boldsymbol{c} + d_j$ in the row of $\boldsymbol{u}_j^\top - \boldsymbol{v}_j^\top$. Then we have that the sequence $(\boldsymbol{A}\boldsymbol{a}_k)_{k\geq 1}$ converges against $\boldsymbol{v}$. We can now apply Lemma 6.3 for $\boldsymbol{A}$, $\boldsymbol{v}$, and $\boldsymbol{B} := \boldsymbol{U_p}$ to obtain vectors $\boldsymbol{a}$ and $\boldsymbol{d}_\infty$ with the desired properties.

For the "if" direction let $\boldsymbol{a}, \boldsymbol{d}_c, \boldsymbol{d}_\infty \in \mathbb{R}^d$ be as in the lemma. We claim that the sequence with $\boldsymbol{a}_k := \boldsymbol{a} - \frac{1}{k}\boldsymbol{d}_c + k\boldsymbol{d}_\infty$ for all $k \geq k_0$ and sufficiently large k_0 is as desired. For convenient notation, define the sequence $\boldsymbol{\rho}_i = \boldsymbol{r}_i^\top \boldsymbol{a}_k$ for each i.

We first show that the sequence $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$ is compatible with $\boldsymbol{p}$. Since $\boldsymbol{d}_c \neq \boldsymbol{0}$, the $\boldsymbol{a}_k$ are pairwise distinct for all $k \geq k_0$ and sufficiently large k_0. If $t_\rho(i) = 1$ (resp. $t_\rho(i) = -1$), then $\boldsymbol{r}_i^\top \boldsymbol{a}_k = \boldsymbol{r}_i^\top \boldsymbol{a} - \frac{1}{k}\boldsymbol{r}_i^\top \boldsymbol{d}_c + k\boldsymbol{r}_i^\top \boldsymbol{d}_\infty$ and since $\boldsymbol{r}_i^\top \boldsymbol{d}_\infty = 0$, $\boldsymbol{r}_i^\top \boldsymbol{a} = \rho(i)$, and $\boldsymbol{r}_i^\top \boldsymbol{d}_c > 0$ (resp. < 0), we have that the sequence $\boldsymbol{\rho}_i$ is strictly increasing (resp. decreasing) and converges against $\rho(i)$ from below (resp. above). If $t_\rho(i) = 0$, then $\boldsymbol{\rho}_i$ is constantly $\rho(i)$ since $\boldsymbol{r}_i^\top \boldsymbol{d}_\infty = 0$, $\boldsymbol{r}_i^\top \boldsymbol{a} = \rho(i)$, and $\boldsymbol{r}_i^\top \boldsymbol{d}_c = 0$. Finally, if $t_\rho(i) = \omega$ (resp. $t_\rho(i) = -\omega$), then $\boldsymbol{r}_i^\top \boldsymbol{d}_\infty > 0$ (resp. < 0) which means that for sufficiently large k_0, the sequence $(\boldsymbol{\rho}_i)_{k\geq k_0} = (\boldsymbol{r}_i^\top \boldsymbol{a}_k)_{k\geq k_0}$ is strictly increasing (resp. decreasing) and diverges to ∞ (resp. $-\infty$). The statement can be shown analogously for $\boldsymbol{\sigma}_i$.

It remains to show that the sequence satisfies the equality constraints for $\boldsymbol{c}$. By (4) we have that $\boldsymbol{u}_j^\top(\boldsymbol{a} - \frac{1}{k}\boldsymbol{d}_c + k\boldsymbol{d}_\infty) = \boldsymbol{v}_j^\top(\boldsymbol{a} - \frac{1}{\ell}\boldsymbol{d}_c + \ell\boldsymbol{d}_\infty) + \boldsymbol{w}_j^\top \boldsymbol{c} + d_j$ for $k < \ell$ if and only if $\boldsymbol{u}_j^\top \boldsymbol{a} = \boldsymbol{v}_j^\top \boldsymbol{a} + \boldsymbol{w}_j^\top \boldsymbol{c} + d_j$ which holds if and only if $(\boldsymbol{u}_j^\top - \boldsymbol{v}_j^\top)\boldsymbol{a} = \boldsymbol{w}_j^\top \boldsymbol{c} + d_j$ which is fulfilled by (4). □

6.5 Constructing the Formula

We now prove Theorem 6.1 in the general case, i.e., $\varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ is an arbitrary existential LRA formula. If φ is a conjunction of inequalities, Lemma 6.4 essentially tells us how to construct an existential formula for $\exists^{\mathsf{ram}}\boldsymbol{x}, \boldsymbol{y}\colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$. Moreover, by Theorem 4.1, we may assume φ to be quantifier-free. Thus, it remains to treat the case that φ is a Boolean combination of constraints as in (9).

We first move all negations inward and, if necessary, negate atoms, so that we are left with a positive Boolean combination of atoms. Let $\alpha_i := \boldsymbol{r}_i^\top \boldsymbol{x} < \boldsymbol{s}_i^\top \boldsymbol{y} + \boldsymbol{t}_i^\top \boldsymbol{z} + h_i$ for $i \in [1, n]$ be the inequality atoms and $\beta_j := \boldsymbol{u}_j^\top \boldsymbol{x} = \boldsymbol{v}_j^\top \boldsymbol{y} + \boldsymbol{w}_j^\top \boldsymbol{z} + d_j$ for $j \in [1, m]$ be the equality atoms in φ.

As in the Presburger case, we now guess a subset of the atoms and then assert that (i) satisfying all these atoms makes φ true and (ii) there exists a clique satisfying the conjunction of these atoms.

Let φ' be the formula obtained from φ by replacing each α_i by $q_i^< = 1$ for a fresh variable $q_i^<$, for all $i \in [1, n]$, and each β_j by $q_j^= = 1$ for a fresh variable $q_j^=$, for all $j \in [1, m]$, and add the restrictions $q_i^< = 0 \vee q_i^< = 1$ and $q_j^= = 0 \vee q_j^= = 1$. Now, φ is equivalent to

$$\psi := \exists \boldsymbol{q}^<, \boldsymbol{q}^=\colon \varphi' \wedge \bigwedge_{i=1}^{n}(q_i^< = 1 \to \alpha_i) \wedge \bigwedge_{j=1}^{m}(q_j^= = 1 \to \beta_j).$$

We represent a profile $\boldsymbol{p}$ by the variables ρ_i, σ_i, $t_{\rho,i}$, and $t_{\sigma,i}$ for all $i \in [1, n]$ where ρ_i, σ_i range over $\mathbb{R}$ and $t_{\rho,i}, t_{\sigma,i}$ range over $\{-2, -1, 0, 1, 2\}$. Here, -2 and 2 represent $-\omega$ and ω, respectively.

We now define formulas for the inequalities and equality constraints from Lemma 6.4. For $i \in [1, n]$, let $\rho_i, \sigma_i, t_{\rho,i}, t_{\sigma,i}, \boldsymbol{x}, \boldsymbol{x}_c, \boldsymbol{x}_\infty$ be fresh variables. Our first formula λ_i contains all the constraints from $\boldsymbol{L_p}\boldsymbol{x} = \boldsymbol{\ell_p}$ and $\boldsymbol{L_p}\boldsymbol{x}_\infty = \boldsymbol{0}$ that stem from the atom α_i:

$$\begin{aligned}\lambda_i := ((t_{\rho,i} = -1 \vee t_{\rho,i} = 1) \to \boldsymbol{r}_i^\top \boldsymbol{x} = \rho_i \wedge \boldsymbol{r}_i^\top \boldsymbol{x}_\infty = 0) \wedge \\ ((t_{\sigma,i} = -1 \vee t_{\sigma,i} = 1) \to \boldsymbol{s}_i^\top \boldsymbol{x} = \sigma_i \wedge \boldsymbol{s}_i^\top \boldsymbol{x}_\infty = 0)\end{aligned}$$

Next, χ_i states the constraints about constant values—meaning: those from $\boldsymbol{C_p}\boldsymbol{x} = \boldsymbol{c_p}$ and $\boldsymbol{C_p}\boldsymbol{x}_c = \boldsymbol{C_p}\boldsymbol{x}_\infty = \boldsymbol{0}$—that stem from α_i:

$$\begin{aligned}\chi_i := (t_{\rho,i} = 0 \to \boldsymbol{r}_i^\top \boldsymbol{x} = \rho_i \wedge \boldsymbol{r}_i^\top \boldsymbol{x}_c = 0 \wedge \boldsymbol{r}_i^\top \boldsymbol{x}_\infty = 0) \wedge \\ (t_{\sigma,i} = 0 \to \boldsymbol{s}_i^\top \boldsymbol{x} = \sigma_i \wedge \boldsymbol{s}_i^\top \boldsymbol{x}_c = 0 \wedge \boldsymbol{s}_i^\top \boldsymbol{x}_\infty = 0)\end{aligned}$$

With δ_i, we express the convergence constraints from $\boldsymbol{D_p x_c} \gg \boldsymbol{0}$ required by α_i:

$$\delta_i := (t_{\rho,i} = -1 \to \boldsymbol{r}_i^\top \boldsymbol{x}_c < 0) \wedge (t_{\rho,i} = 1 \to \boldsymbol{r}_i^\top \boldsymbol{x}_c > 0) \wedge$$
$$(t_{\sigma,i} = -1 \to \boldsymbol{s}_i^\top \boldsymbol{x}_c < 0) \wedge (t_{\sigma,i} = 1 \to \boldsymbol{s}_i^\top \boldsymbol{x}_c > 0)$$

Furthermore, μ_i states the unboundedness condition $\boldsymbol{U_p x_\infty} \gg \boldsymbol{0}$:

$$\mu_i := (t_{\rho,i} = -2 \to \boldsymbol{r}_i^\top \boldsymbol{x}_\infty < 0) \wedge (t_{\rho,i} = 2 \to \boldsymbol{r}_i^\top \boldsymbol{x}_\infty > 0) \wedge$$
$$(t_{\sigma,i} = -2 \to \boldsymbol{s}_i^\top \boldsymbol{x}_\infty < 0) \wedge (t_{\sigma,i} = 2 \to \boldsymbol{s}_i^\top \boldsymbol{x}_\infty > 0)$$

Finally, ε_j expresses the equality constraints (5) in Lemma 6.4: For all $j \in [1, m]$ let

$$\varepsilon_j := \boldsymbol{u}_j^\top \boldsymbol{x}_c = 0 \wedge \boldsymbol{u}_j^\top \boldsymbol{x}_\infty = 0 \wedge \boldsymbol{v}_j^\top \boldsymbol{x}_c = 0 \wedge \boldsymbol{v}_j^\top \boldsymbol{x}_\infty = 0 \wedge (\boldsymbol{u}_j^\top - \boldsymbol{v}_j^\top)\boldsymbol{x} = \boldsymbol{w}_j^\top \boldsymbol{z} + x_j.$$

To check if $\boldsymbol{p}$ is a $\boldsymbol{z}$-admissible profile, we define the formula

$$\theta := \bigwedge_{i=1}^{n} t_{\rho,i} \in \{-2,-1,0,1,2\} \wedge t_{\sigma,i} \in \{-1,0,1,2\} \wedge (t_{\rho,i} = 2 \to t_{\sigma,i} = 2) \wedge$$
$$[(t_{\rho,i} \in \{-1,0\} \wedge t_{\sigma,i} \in \{0,1\} \vee t_{\rho,i} = -1 \wedge t_{\sigma,i} = -1) \to \rho_i < \sigma_i + \boldsymbol{t}_i^\top \boldsymbol{z} + h_i] \wedge$$
$$[(t_{\rho,i} = 0 \wedge t_{\sigma,i} = -1 \vee t_{\rho,i} = 1) \to \rho_i \le \sigma_i + \boldsymbol{t}_i^\top \boldsymbol{z} + h_i]$$

where we use set notation as a shorthand. Then we claim that $\exists^{\mathrm{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \psi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ is equivalent to

$$\gamma := \exists \boldsymbol{q}^<, \boldsymbol{q}^=, \boldsymbol{p}, \boldsymbol{x}, \boldsymbol{x}_c, \boldsymbol{x}_\infty \colon \varphi' \wedge \theta \wedge \boldsymbol{x}_c \neq \boldsymbol{0} \wedge$$
$$\bigwedge_{i=1}^{n} (q_i^< = 1 \to \lambda_i \wedge \chi_i \wedge \delta_i \wedge \mu_i) \wedge \bigwedge_{j=1}^{m} (q_j^= = 1 \to \varepsilon_j).$$

We show that for any valuation $\boldsymbol{c} \in \mathbb{R}^d$ of $\boldsymbol{z}$ we have $\exists^{\mathrm{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \psi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c})$ if and only if $\gamma(\boldsymbol{c})$. For an assignment ν of the $q_i^<, q_j^=$ to $\{0, 1\}$ let $I_\nu := \{i \in [1, n] \mid \nu(q_i^<) = 1\}$ and $J_\nu := \{j \in [1, m] \mid \nu(q_j^=) = 1\}$. By Ramsey's theorem, $\exists^{\mathrm{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \psi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c})$ holds if and only if there there is an assignment ν of the $q_i^<, q_j^=$ satisfying φ' such that $\exists^{\mathrm{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \bigwedge_{i \in I_\nu} \alpha_i(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c}) \wedge \bigwedge_{j \in J_\nu} \beta_j(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c})$. By Lemmas 6.2 and 6.4, this is equivalent to $\exists \boldsymbol{p}, \boldsymbol{x}, \boldsymbol{x}_c, \boldsymbol{x}_\infty \colon \theta(\boldsymbol{p}, \boldsymbol{c}) \wedge \boldsymbol{x}_c \neq \boldsymbol{0} \wedge \bigwedge_{i \in I_\nu} \lambda_i \wedge \chi_i \wedge \delta_i \wedge \mu_i \wedge \bigwedge_{j \in J_\nu} \varepsilon_j(\boldsymbol{x}, \boldsymbol{x}_c, \boldsymbol{x}_\infty, \boldsymbol{c})$. This holds for some assignment ν of the $q_i^<, q_j^=$ satisfying φ' if and only if $\gamma(\boldsymbol{c})$.

Using standard arguments, one can observe that Theorem 6.1 has an analogue over the rationals:

Theorem 6.5. *Given an existential formula $\varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ over $\langle \mathbb{Q}; +, <, 1, 0 \rangle$, we can construct in polynomial time an existential formula over $\langle \mathbb{Q}; +, <, 1, 0 \rangle$ of linear size that is equivalent to $\exists^{\mathrm{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$.*

The proof can be found in the full version of this paper [Bergsträßer et al. 2023a].

7 RAMSEY QUANTIFIERS IN LINEAR INTEGER REAL ARITHMETIC

We now show elimination of the Ramsey quantifier in LIRA. At the end of the section, we mention a version of this result for the structure $\langle \mathbb{Q}; \lfloor \cdot \rfloor, +, <, 1, 0 \rangle$ (Theorem 7.3).

Theorem 7.1. *Given an existential formula $\varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ in LIRA, we can construct in polynomial time an existential formula in LIRA of linear size that is equivalent to $\exists^{\mathrm{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$.*

Proof. It suffices to show the theorem for the decomposition of φ: Given φ, we first compute its decomposition $\varphi'(\boldsymbol{x}^{\mathrm{i/r}}, \boldsymbol{y}^{\mathrm{i/r}}, \boldsymbol{z}^{\mathrm{i/r}})$ using Lemma 3.5. We then show how to compute a formula $\psi'(\boldsymbol{z}^{\mathrm{i/r}})$ in LIRA that is equivalent to $\exists^{\mathrm{ram}} \boldsymbol{x}^{\mathrm{i/r}}, \boldsymbol{y}^{\mathrm{i/r}} \colon \varphi'(\boldsymbol{x}^{\mathrm{i/r}}, \boldsymbol{y}^{\mathrm{i/r}}, \boldsymbol{z}^{\mathrm{i/r}})$. Let $\psi(\boldsymbol{z})$ be the formula obtained from ψ' by replacing every z_i^{int} by $\lfloor z_i \rfloor$ and every z_i^{real} by $z_i - \lfloor z_i \rfloor$. Now ψ is equivalent to $\exists^{\mathrm{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ since $\exists^{\mathrm{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{c})$ if and only if $\boldsymbol{c} = \boldsymbol{c}^{\mathrm{int}} + \boldsymbol{c}^{\mathrm{real}}$ and $\exists^{\mathrm{ram}} \boldsymbol{x}^{\mathrm{i/r}}, \boldsymbol{y}^{\mathrm{i/r}} \colon \varphi'(\boldsymbol{x}^{\mathrm{i/r}}, \boldsymbol{y}^{\mathrm{i/r}}, \boldsymbol{c}^{\mathrm{i/r}})$.

Thus, we now assume that $\varphi'(\boldsymbol{x}^{\mathrm{i/r}}, \boldsymbol{y}^{\mathrm{i/r}}, \boldsymbol{z}^{\mathrm{i/r}})$ is a decomposition of φ. By Theorem 4.1 we can assume that φ' is quantifier-free. We further assume that all negations are moved directly into the atoms. Let $\alpha_1, \ldots, \alpha_n$ be the atoms in LRA and $\beta_1, \ldots, \beta_m$ be the Presburger atoms of φ'. For fresh real variables $p_1, \ldots, p_n$ and integer variables $q_1, \ldots, q_m$ let σ be the formula obtained from φ' by replacing every α_i by $p_i = 1$ and every β_j by $q_j = 1$ and adding the constraints $p_i = 0 \vee p_i = 1$ and $q_j = 0 \vee q_j = 1$. Then φ' is equivalent to

$$\delta := \exists \boldsymbol{p}, \boldsymbol{q} \colon \sigma \wedge \bigwedge_{i=1}^{n} (p_i = 1 \rightarrow \alpha_i) \wedge \bigwedge_{j=1}^{m} (q_j = 1 \rightarrow \beta_j).$$

Since each p_i and q_j only has finitely many (in fact two) possible valuations, Ramsey's theorem implies that $\exists^{\mathrm{ram}} \boldsymbol{x}^{\mathrm{i/r}}, \boldsymbol{y}^{\mathrm{i/r}} \colon \delta(\boldsymbol{x}^{\mathrm{i/r}}, \boldsymbol{y}^{\mathrm{i/r}}, \boldsymbol{z}^{\mathrm{i/r}})$ is equivalent to

$$\exists \boldsymbol{p}, \boldsymbol{q} \colon \sigma \wedge \exists^{\mathrm{ram}} \boldsymbol{x}^{\mathrm{i/r}}, \boldsymbol{y}^{\mathrm{i/r}} \colon \bigwedge_{i=1}^{n} (p_i = 1 \rightarrow \alpha_i) \wedge \bigwedge_{j=1}^{m} (q_j = 1 \rightarrow \beta_j).$$

Let $\alpha := \bigwedge_{i=1}^{n} (p_i = 1 \rightarrow \alpha_i)$ and $\beta := \bigwedge_{j=1}^{m} (q_j = 1 \rightarrow \beta_j)$. To split the vectors of the Ramsey quantified variables into real and integer components, we have to allow that in the infinite clique either the real components or the integer components do not change throughout the clique. To this end, we introduce a fresh variable r that is either 0 or 1 and get the equivalent formula

$$\begin{aligned} &\exists \boldsymbol{p}, \boldsymbol{q}, r \colon \sigma \wedge (r = 0 \vee r = 1) \wedge \\ &\qquad \left[\left(\exists^{\mathrm{ram}} \boldsymbol{x}^{\mathrm{real}}, \boldsymbol{y}^{\mathrm{real}} \colon \alpha(\boldsymbol{p}, \boldsymbol{x}^{\mathrm{real}}, \boldsymbol{y}^{\mathrm{real}}, \boldsymbol{z}^{\mathrm{real}}) \right) \vee r = 0 \wedge \exists \boldsymbol{x}^{\mathrm{real}} \colon \alpha(\boldsymbol{p}, \boldsymbol{x}^{\mathrm{real}}, \boldsymbol{x}^{\mathrm{real}}, \boldsymbol{z}^{\mathrm{real}}) \right] \wedge \\ &\qquad \left[\left(\exists^{\mathrm{ram}} \boldsymbol{x}^{\mathrm{int}}, \boldsymbol{y}^{\mathrm{int}} \colon \beta(\boldsymbol{q}, \boldsymbol{x}^{\mathrm{int}}, \boldsymbol{y}^{\mathrm{int}}, \boldsymbol{z}^{\mathrm{int}}) \right) \vee r = 1 \wedge \exists \boldsymbol{x}^{\mathrm{int}} \colon \beta(\boldsymbol{q}, \boldsymbol{x}^{\mathrm{int}}, \boldsymbol{x}^{\mathrm{int}}, \boldsymbol{z}^{\mathrm{int}}) \right] \end{aligned}$$

where the Ramsey quantifiers can be eliminated by Theorems 5.1 and 6.1. □

Let us mention a simple consequence.

Corollary 7.2. *The infinite clique problem for existential formulas in LIRA is* NP*-complete.*

The proof can be found in the full version of this paper [Bergsträßer et al. 2023a].

Theorem 7.3. *Given an existential formula $\varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ over $\langle \mathbb{Q}; \lfloor \cdot \rfloor, +, <, 1, 0 \rangle$, we can construct in polynomial time an existential formula of linear size that is equivalent to $\exists^{\mathrm{ram}} \boldsymbol{x}, \boldsymbol{y} \colon \varphi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ over $\langle \mathbb{Q}; \lfloor \cdot \rfloor, +, <, 1, 0 \rangle$.*

Proof. We use almost the same construction as for Theorem 7.1. The only difference is that we use Theorem 6.5 in place of Theorem 6.1. □

8 APPLICATIONS

In this section, we present further applications of our results.

8.1 Monadic Decomposability

A formula is called *monadic* if every atom contains at most one variable. As mentioned above, monadic formulas play an important role in constraint databases[Grumbach et al. 2001; Kuper et al. 2000]. Partly motivated by this, Veanes et al. [2017] recently raised the question of how to decide whether a given formula φ is equivalent to a monadic formula. In this case, φ is called *monadically decomposable*. For LIA, monadic decomposability was shown decidable (under slightly different terms) by Ginsburg and Spanier [1966, Corollary, p. 1048] and a more general result by Libkin [2003, Theorem 3] establishes dedicability for LIA, LRA, and other logics [Libkin 2003, Corollaries 7,8]. In terms of complexity, given a quantifier-free LIA formula, monadic decomposability was shown coNP-complete by Hague et al. [2020]. However, it remained open what the complexity is in the case of LRA and LIRA. From Theorems 5.1, 6.1 and 7.1, we can conclude the following:

Corollary 8.1. *Given a quantifier-free formula in LIA, LRA, or LIRA, deciding monadic decomposability is* coNP*-complete.*

As mentioned above, the result about LIA was also shown by Hague et al. [2020]. The coNP-hardness in Corollary 8.1 uses the same idea as [Bergsträßer et al. 2022, Lemma 6.4] (see the full version of this paper [Bergsträßer et al. 2023a] for details). The coNP upper bound follows from Theorems 5.1, 6.1 and 7.1 as follows. Suppose $\varphi(x, \boldsymbol{y})$ is a formula in LIA, LRA, or LIRA with some free variable x and a further vector $\boldsymbol{y}$ of free variables. We define the equivalence $\sim_{\varphi,x}$ on the domain D (i.e. $\mathbb{R}$ or $\mathbb{Z}$) by

$$a \sim_{\varphi,x} b \iff \text{for all } \boldsymbol{c} \in D^{|\boldsymbol{y}|}, \text{ we have } \varphi(a, \boldsymbol{c}) \text{ iff } \varphi(b, \boldsymbol{c}).$$

Note that if φ is quantifier-free, we can easily construct a linear-size existential formula for the negation of $\sim_{\varphi,x}$ by setting $\delta_{\varphi,x}(x, x') := \exists \boldsymbol{y} \colon \neg(\varphi(x, \boldsymbol{y}) \leftrightarrow \varphi(x', \boldsymbol{y}))$. For LIA, LRA, and LIRA, the formula $\varphi(x_1, \ldots, x_n)$ is monadically decomposable if and only if for each $i \in \{1, \ldots, n\}$, the equivalence $\sim_{\varphi,x_i}$ has only finitely many equivalence classes. This is shown in [Libkin 2003, Lemma 4] for LIA and LRA and in [Bergsträßer and Ganardi 2023b, Lemma 10] for LIRA. Thus, the formula $\varphi(x_1, \ldots, x_n)$ is *not* monadically decomposable if and only if $\mu_x := \exists^{\mathsf{ram}}(x, x') \colon \delta_{\varphi,x_i}$ holds for some $i \in \{1, \ldots, n\}$. Thus, we can decide monadic non-decomposability by deciding in NP each of the n formulas μ_x by applying Theorems 5.1, 6.1 and 7.1.

We should mention that although a coNP algorithm was known for LIA, our new procedure to decide monadic decomposability is asymptotically much more efficient than the one by Hague et al. [2020]: They construct for each variable x a formula ν_x that contains an exponential constant B [Hague et al. 2020, p. 128]. They choose $B = 2^{dmn+3}$ [Hague et al. 2020, p. 132], where (i) d is the number of bits needed to encode constants in φ, (ii) n is the number of linear inequalities in any disjunct in φ, and (iii) m is the number of variables in φ. Thus, this constant requires $dmn + 3$ bits, meaning ν_x is of length $O(dmn)$, which is cubic in the size of the input formula. In contrast, each of our formulas μ_x (and thus the result after eliminating $\exists^{\mathsf{ram}}$) is of linear size.

8.2 Linear Liveness for Systems with Counters and Clocks

As already observed by Bergsträßer et al. [2022], the Ramsey quantifier can be used to check liveness properties of formal systems, provided that the reachability relation is expressible in the respective logic. This yields several applications for systems that involve counters and/or clocks.

Specifically, there is a rich variety of models where a configuration is an element of $C = Q \times \mathbb{Z}^k \times \mathbb{D}^\ell$, where Q is a finite set of control states, and $\mathbb{D}$ is either $\mathbb{R}$ or $\mathbb{Q}$, with a step relation $\to \subseteq C \times C$, and for $p, q \in Q$, one can effectively construct an existential first-order formula $\varphi_{p,q}(\boldsymbol{x}, \boldsymbol{y})$ for the reachability relation: This means, $(p, \boldsymbol{x}) \xrightarrow{*} (q, \boldsymbol{y})$ if and only if $\varphi_{p,q}(\boldsymbol{x}, \boldsymbol{y})$. Here the components $\mathbb{Z}^k$ and $\mathbb{D}^\ell$ hold counter or clock values. We will see some concrete examples below.

For systems of this type, we can consider the *linear liveness problem*:

Given A description of a system, a formula $\psi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$, and a state q.
Question Is there an infinite run $(q_1, \boldsymbol{u}_1) \to (q_2, \boldsymbol{u}_2) \to \cdots$ and a vector $\boldsymbol{v}$ such that for some infinite set $I \subseteq \mathbb{N}$, we have $q_i = q$ for every $i \in I$ and $\psi(\boldsymbol{u}_i, \boldsymbol{u}_j, \boldsymbol{v})$ for any $i, j \in I$ with $i < j$.

Here, a simple case is that ψ simply states a linear condition on each configuration (thus, $\psi(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ would just depend on $\boldsymbol{x}$). But one can also require that between $(q_i, \boldsymbol{u}_i)$ and $(q_j, \boldsymbol{u}_j)$, the values in $\boldsymbol{u}_i$ and $\boldsymbol{u}_j$ have increased by at least some positive value in $\boldsymbol{v}$. With this, one can express, e.g. that clock values grow unboundedly (rather than converging).

If we have reachability formulas $\varphi_{p,q}$ as above, linear liveness can easily be decided using the Ramsey quantifier: Note that there is a run as above if and only if for some state $q \in Q$, we have

$$\exists z \colon \exists^{\mathsf{ram}}(x, y) \colon \varphi_{q,q}(x, y) \wedge \psi(x, y, z). \tag{11}$$

Let us see some applications of this.

Timed (pushdown) automata. In a *timed automaton* [Alur and Dill 1994], configurations are elements of $Q \times \mathbb{R}^{\ell}$ and the real numbers are clock values. In each step, some time can elapse or, depending on satisfaction of guards, some counters can be reset; see [Alur and Dill 1994] for details. It was shown by Comon and Jurski [1999, Theorem 5] that the reachability relation in timed automata is effectively definable in $\langle \mathbb{R}; +, <, 1, 0 \rangle$. Using a conceptually simpler construction, Quaas et al. [2017, Theorem 10] construct an exponential-size existential formula in $\langle \mathbb{R}; +, <, 0, 1 \rangle$ for the reachability relation. Using the formula (11) and our results, we can thus decide the linear liveness problem for timed automata in NEXPTIME. Recall that liveness in timed automata is PSPACE-complete [Alur and Dill 1994, Theorem 4.17]. The difference to linear liveness is that in the latter, one can express arbitrary LIRA constraints (even between configurations).

In order to model timed behavior of recursive programs, timed automata have been extended by stacks, where each stack either has [Abdulla et al. 2012] or does not have [Bouajjani et al. 1994] its own clock value. These two versions are semantically equivalent and have been extended to *timed pushdown automata* [Clemente and Lasota 2018], a strict extension that allows additional counter constraints. Clemente and Lasota [2018, Theorem 5] show that the reachability relation, between two configurations with empty stack, is definable by a doubly-exponential existential formula over $\langle \mathbb{Q}; +, <, 0, 1 \rangle$ (for the more restricted model of Abdulla et al. [2012], existence of such a formula had been shown by Dang [2003], but without complexity bounds). Based on this, our results allow us to decide the linear liveness problem for timed pushdown automata in 2NEXPTIME, if we view each run from empty stack to empty stack as a single step of the system.

Continuous vector addition systems with states. Vector addition systems with states (VASS; a.k.a. Petri nets) are arguably the most popular formal model for concurrent systems. They consist of a control state and some counters that assume natural numbers. Since the reachability problem is Ackermann-complete [Czerwinski and Orlikowski 2021; Leroux 2021; Leroux and Schmitz 2019] and the coverability problem is EXPSPACE-complete [Lipton 1976; Rackoff 1978], there has been substantial interest in finding overapproximations where these problems become easier.

A particularly successful overapproximation is the *continuous semantics*, where each added vector is non-deterministically multiplied by some $0 < \alpha < 1$. This has been used to speed up the backward search procedure by pruning configurations that cannot cover the target continuously [Blondin et al. 2016]. Thus, in continuous semantics, the configurations belong to $Q \times \mathbb{Q}^{\ell}$.

As shown by Blondin and Haase [2017, Theorem 4.9], the reachability relation under continuous semantics can be described by a polynomial-size existential formula over $\langle \mathbb{Q}; +, <, 0, 1 \rangle$. Thus, our results imply NP-completeness of the linear liveness problem for continuous VASS (NP-hardness easily follows from NP-hardness of reachability [Blondin and Haase 2017, Theorem 4.14]).

Systems with discrete counters. There are several prominent types of counter systems for which one can compute an existential Presburger formula for the reachability relation. The most well-known example are *reversal-bounded counter machines* (RBCM) [Ibarra 1978]. These admit an existential formula for the reachability relation, even if one of the counters has no reversal bound [Ibarra et al. 2000, Theorem 12], even with a polynomial-time construction [Hague and Lin 2011].

Closely related to RBCM are *Parikh automata* (PA) [Klaedtke and Rueß 2003], for which one can also compute an existential Presburger formula for the reachability relation in polynomial time.

Another example is the class of *succinct one-counter systems*, which have one unrestricted counter with binary-encoded updates. Based on [Haase et al. 2009], Li et al. [2020] have shown that one can construct in polynomial time an existential Presburger formula for the reachability relations. In fact, using the proof techniques in [Hague and Lin 2012; To 2009], this result can be extended to multithreaded programs with k threads — each represented as a succinct one-counter system — where inter-thread communication is limited, e.g., when the number of context switches is also fixed (in the style of [Qadeer and Rehof 2005]).

Thus, our results imply that for these models, the linear liveness problem is NP-complete. (Again, NP-hardness follows using a simple reduction from the reachability problem.) In the case of PA, this strengthens recent results on PA over infinite words [Grobler et al. 2023; Guha et al. 2022].

8.3 Deciding Whether a Relation Is a WQO

The concept of well-structured transition systems (WSTS) [Abdulla et al. 1996; Finkel and Schnoebelen 2001] is a cornerstone of the verification of infinite-state systems. Here, the key idea is to order the configurations of a system by a well-quasi-ordering (WQO). This recently led Finkel and Gupta [2019a] to consider the problem of automatically establishing that a given counter machine is well-structured. In particular, they raised the problem of deciding whether a relation, specified by a formula in Presburger arithmetic, is a well-quasi ordering. Finkel and Gupta [2019b] show that this is decidable using Ramsey quantifiers in automatic structures, which leads to high complexity: For quantifier-free formulas this results in a PSPACE procedure by constructing an NFA for the negation and then evaluating a Ramsey quantifier using [Bergsträßer et al. 2022].

Our results settle the complexity, if the relation is given by a quantifier-free formula $\varphi(\boldsymbol{x},\boldsymbol{y})$: Deciding whether φ defines a WQO is coNP-complete. Suppose $\boldsymbol{x}$ and $\boldsymbol{y}$ are vectors of k variables and thus φ defines a relation on $\mathbb{Z}^k$. Recall that a relation $R \subseteq \mathbb{Z}^k$ is a WQO iff it is reflexive and transitive and for every infinite sequence $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots$, there are $i < j$ with $(\boldsymbol{a}_i, \boldsymbol{a}_j) \in R$. Thus, φ violates the conditions of a WQO if and only if (1) $\exists \boldsymbol{x}\colon \neg\varphi(\boldsymbol{x},\boldsymbol{x})$ (reflexivity violation) or (2) $\exists \boldsymbol{x},\boldsymbol{y},\boldsymbol{z}\colon \varphi(\boldsymbol{x},\boldsymbol{y}) \wedge \varphi(\boldsymbol{y},\boldsymbol{z}) \wedge \neg\varphi(\boldsymbol{x},\boldsymbol{z})$ (transitivity violation) or (3) $\exists^{\mathsf{ram}}\boldsymbol{x},\boldsymbol{y}\colon \neg\varphi(\boldsymbol{x},\boldsymbol{y})$ (violation of the sequence condition). Thus, we obtain an NP procedure using Theorem 5.1 and Proposition 3.1. Here, coNP-hardness can be shown using a simple ad-hoc proof (see the full version of this paper [Bergsträßer et al. 2023a]).

9 EXPERIMENTS

We have implemented a prototype (which can be found at [Bergsträßer et al. 2023b]) of our Ramsey quantifier elimination algorithms for LIA, LRA, and LIRA in Python using the Z3 [de Moura and Bjørner 2008] interface Z3Py. We have tested it against two sets of micro-benchmarks. The first benchmarks contain the following examples, where the dimension d of $\boldsymbol{x}$ and $\boldsymbol{y}$ is a parameter:

(a) $\varphi_{\text{half}} := \exists^{\mathsf{ram}}\boldsymbol{x},\boldsymbol{y}\colon 2\boldsymbol{y} \le \boldsymbol{x} \wedge \boldsymbol{x} \ge \boldsymbol{t}$ for parameter $t \in \mathbb{Z}$
(b) $\varphi_{\text{eq\_ex}} := \exists^{\mathsf{ram}}\boldsymbol{x},\boldsymbol{y}\colon \exists \boldsymbol{z}\colon \boldsymbol{x} \ll \boldsymbol{y} \wedge \boldsymbol{x} = \boldsymbol{z}$
(c) $\varphi_{\text{eq\_free}} := \exists^{\mathsf{ram}}\boldsymbol{x},\boldsymbol{y}\colon \boldsymbol{x} \ll \boldsymbol{y} \wedge \boldsymbol{x} = \boldsymbol{z}$
(d) $\varphi_{\text{dickson}} := \exists^{\mathsf{ram}}\boldsymbol{x},\boldsymbol{y}\colon \boldsymbol{x} \ge \boldsymbol{0} \wedge (\boldsymbol{x} > \boldsymbol{y} \vee \boldsymbol{x} \not\le \boldsymbol{y} \wedge \boldsymbol{y} \not\le \boldsymbol{x})$ where unsatisfiability over $\mathbb{Z}$ proves Dickson's lemma
(e) $\varphi_{\text{program}} := \exists^{\mathsf{ram}}(\boldsymbol{x}_1,\boldsymbol{x}_2),(\boldsymbol{y}_1,\boldsymbol{y}_2)\colon \boldsymbol{x}_1 \gg \boldsymbol{0} \wedge \boldsymbol{x}_2 \gg \boldsymbol{0} \wedge \boldsymbol{y}_1 \ge \boldsymbol{0.5}\boldsymbol{x}_1 + \boldsymbol{0.5} \wedge \boldsymbol{y}_2 \le \boldsymbol{x}_2 - \lfloor \boldsymbol{x}_1 \rfloor$ that describes an under-approximation of the non-terminating program in Section 2, where $\boldsymbol{x}_1, \boldsymbol{y}_1$ are vectors of real variables and $\boldsymbol{x}_2, \boldsymbol{y}_2$ are vectors of integer variables.

Here, $\lfloor \boldsymbol{v} \rfloor$ for a vector $\boldsymbol{v}$ denotes the vector $(\lfloor v_1 \rfloor, \ldots, \lfloor v_d \rfloor)$. Moreover, recall that for numbers n we write $\boldsymbol{n}$ for the vector $(n, \ldots, n)$ of appropriate dimension.

Table 1. Experiments for the elimination of the Ramsey quantifier with a 500 seconds timeout.

formula	dom	sat	input		output						
			#vars	#atoms	#vars	#atoms	$d=1$	$d=10$	$d=20$	$d=50$	$d=100$
φ_{half}	$\mathbb{Z}$	no	$2d$	$2d$	$22d$	$130d$	0.04s	0.33s	0.84s	3.35s	11.00s
	$\mathbb{R}$	$t \leq 0$			$25d$	$284d$	0.06s	0.75s	2.10s	10.31s	38.48s
$\varphi_{\text{eq\_ex}}$	$\mathbb{Z}$	yes	$3d$	$2d$	$31d$	$166d$	0.05s	0.67s	2.01s	10.23s	39.44s
	$\mathbb{R}$	yes			$25d$	$213d$	0.05s	1.03s	3.53s	20.01s	82.46s
$\varphi_{\text{eq\_free}}$	$\mathbb{Z}$	no	$3d$	$2d$	$28d$	$162d$	0.05s	0.44s	1.12s	4.73s	16.11s
	$\mathbb{R}$	no			$20d$	$209d$	0.08s	0.54s	1.64s	8.18s	31.03s
φ_{dickson}	$\mathbb{Z}$	no	$2d$	$5d$	$37d$	$226d$	0.06s	0.58s	1.52s	6.33s	21.60s
	$\mathbb{R}$	yes			$40d$	$482d$	0.08s	1.17s	4.48s	17.18s	66.46s
φ_{program}	$\mathbb{R}, \mathbb{Z}$	yes	$6d$	$14d$	$426d+1$	$3858d+4$	0.84s	68.28s	445.89s	> 500s	> 500s

The experiments were conducted on an Intel(R) Core(TM) i7-10510U CPU with 16GB of RAM running on Windows 10. The results are summarized in Table 1. We observe that the number of output variables and atoms linearly depends on the number of input variables and atoms. In the first three cases, the output formula has ca. 5 times as many variables as the input has variables and atoms. The choice of parameter $t \in \mathbb{Z}$ has no notable effect on the size of the output formula or the running time since it only changes constants. For φ_{program} our prototype implementation assumes the formula to be decomposed into a Boolean combination of LIA and LRA formulas whose size is given in the input column of Table 1. Then the running time is dominated by the Z3 satisfiability check due to the large number of variables and atoms in the output.

For the second benchmarks we used our elimination procedure to check monadic decomposability, as described in Section 8, of the following formulas:

(a) $\varphi_{\text{imp}} := \bigwedge_{i=1}^{d} x_i \geq 0 \rightarrow x_i + y_i \geq k \wedge y_i \geq 0$ for parameter $k \in \mathbb{N}$
(b) $\varphi_{\text{diagonal}} := \mathbf{0} \leq \boldsymbol{x} \leq \boldsymbol{k} \wedge x_1 = \cdots = x_d$ for parameter $k \in \mathbb{N}$
(c) $\varphi_{\text{cubes2d}} := x_1 + x_2 \leq k \wedge \bigwedge_{i=1}^{k} i \leq x_1 \leq i+2 \wedge i \leq x_2 \leq i+2$ where parameter $k \in \mathbb{N}$ is the number of cubes
(d) $\varphi_{\text{cubes10}} := \bigwedge_{i=1}^{10} \boldsymbol{i} \leq \boldsymbol{x} \leq \boldsymbol{i}+\mathbf{2}$
(e) $\varphi_{\text{mixed}} := \boldsymbol{x} = \lfloor \boldsymbol{y} \rfloor \wedge \mathbf{0} \leq \boldsymbol{y} \leq \boldsymbol{k}$ over LIRA with parameter $k \in \mathbb{N}$

The results are shown in Table 2 where either the dimension d or parameter k is varied. The size of the input refers to the formula $\delta_{\varphi,(x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_d)}$ for $i = 1$ that is defined similarly to $\delta_{\varphi,x}$ in Section 8 but uses only one existentially quantified variable. This has the advantage that the algorithm only has to eliminate one existential variable before eliminating the Ramsey quantifier. For the output we measure the size of the first formula given to Z3, i.e., $\delta_{\varphi,(x_2,\ldots,x_d)}$ after elimination of the Ramsey quantifier. We observe that if n is the number of input variables plus atoms, on these instances the number of output variables can be estimated by $5 \cdot n$ over $\mathbb{Z}$ and $10 \cdot n$ over $\mathbb{R}$. Note that not only is the formula $\delta_{\varphi,(x_2,\ldots,x_d)}$ (the input to the elimination procedure) larger than φ, where monadic decomposability is checked on, we also have to consider all of the $\delta_{\varphi,(x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_d)}$ in case φ is monadically decomposable. This explains the slowdown compared to Table 1.

For φ_{imp} and $\varphi_{\text{diagonal}}$ we observe that, despite the larger output formula, over $\mathbb{R}$ the algorithm terminates significantly faster than over $\mathbb{Z}$ since it only needs to construct $\delta_{\varphi,(x_2,\ldots,x_d)}$ to detect that φ is not monadically decomposable. The first four examples are taken from [Markgraf et al. 2021] where the authors compare their tool to mondec_1 from [Veanes et al. 2017] that *computes* a monadic decomposition if one exists. We observe that on these instances our *decision* algorithm is significantly faster than mondec_1, especially for φ_{imp} and $\varphi_{\text{diagonal}}$ when only the parameter k is varied (and $d = 1$ resp. $d = 2$ as in [Markgraf et al. 2021]). The reason for this is that mondec_1 computes the monadic decomposition whose size grows exponentially in the encoding of k, whereas in our approach, where we only decide if a decomposition exists, k only changes a constant in the

Table 2. Experiments for monadic decomposability with a 500 seconds timeout.

formula	dom	mondec	input		output					
			#vars	#atoms	#vars	#atoms				
φ_{imp}			$4d-1$	$12d$			$d=1$	$d=5$	$d=10$	$d=20$
	$\mathbb{Z}$	yes			$84d-8$	$516d-62$	0.12s	6.19s	34.11s	224.53s
	$\mathbb{R}$	no			$136d-7$	$1678d-130$	0.32s	2.27s	6.69s	19.01s
$\varphi_{\text{diagonal}}$			$2d-1$	$4d+4$			$d=2$	$d=10$	$d=20$	$d=30$
	$\mathbb{Z}$	yes			$52d-8$	$322d-62$	0.15s	8.20s	46.48s	151.42s
	$\mathbb{R}$	no			$35d+59$	$416d+716$	0.32s	1.33s	3.82s	7.37s
φ_{cubes2d}			3	$16k+4$			$k=50$	$k=100$	$k=150$	$k=250$
	$\mathbb{Z}$	yes			$80k+36$	$512k+198$	7.74s	18.77s	37.73s	109.17s
	$\mathbb{R}$	no			$176k+63$	$2256k+702$	210.12s	> 500s	> 500s	> 500s
φ_{cubes10}			$2d-1$	$80d$			$d=2$	$d=10$	$d=15$	$d=20$
	$\mathbb{Z}$	yes			$412d-8$	$2626d-62$	1.18s	66.40s	231.67s	482.39s
	$\mathbb{R}$	yes			$893d-7$	$11414d-130$	5.83s	> 500s	> 500s	> 500s
φ_{mixed}							$d=1$	$d=2$	$d=3$	$d=4$
	$\mathbb{R},\mathbb{Z}$	yes	$6d-1$	$28d$	$842d-28$	$7710d-198$	3.67s	42.84s	192.76s	> 500s

formulas where the Ramsey quantifier is eliminated. Therefore, changing k in the two examples (and also in φ_{mixed}) does not have any notable effect on the running time in Table 2. In this case, our algorithm is also faster than the one developed in [Markgraf et al. 2021] that outputs the decomposition in form of cubes. Since both algorithms in [Veanes et al. 2017] and [Markgraf et al. 2021] only terminate if the input formula is monadically decomposable, our algorithm is the only one that terminates on φ_{imp}, $\varphi_{\text{diagonal}}$, and φ_{cubes2d} over $\mathbb{R}$ and can therefore be used as a termination check in the other algorithms. Finally, note that the increase of the running time for φ_{cubes2d}, φ_{cubes10} over $\mathbb{R}$ and φ_{mixed} is due to the large number of atoms in the output, which is problematic not only for the elimination procedure but especially for the satisfiability check with Z3. We observe that for large instances, the running time is dominated by the satisfiability check.

10 CONCLUSION AND FUTURE WORK

We have given efficient algorithms for removing Ramsey quantifiers from the theories of Linear Integer Arithmetic (LIA), Linear Real Arithmetic (LRA), and Linear Integer Real Arithmetic (LIRA). The algorithm runs in polynomial time and is guaranteed to produce formulas of linear size. We have shown that this leads to applications in proving termination/non-termination of programs, as well as checking variable dependencies (a.k.a. monadic decomposability) in a given formula.

We mention several future research avenues. First, combined with existing results on computation of reachability relations [Bardin et al. 2008, 2005; Boigelot and Herbreteau 2006; Boigelot et al. 2003; Legay 2008], we obtain fully-automatic methods for proving termination/non-termination. Recent software verification frameworks, however, rely on *Constraint Horn Clauses* (CHC), which extend SMT with recursive predicate, e.g., see [Bjørner et al. 2015, 2012]. To extend the framework for proving termination, one typically extends CHC with ad-hoc well-foundedness conditions [Beyene et al. 2013]. Our results suggest that we can instead extend CHC with Ramsey quantifiers, and develop synthesis algorithms for the framework. We leave this for future work. Second, we also mention that eliminability of Ramsey quantifiers from other theories (e.g. non-linear real arithmetics and EUF) remains open, which we also leave for future work.

DATA-AVAILABILITY STATEMENT

The experimental results of this paper may be reproduced using the artifact on Zenodo [Bergsträßer et al. 2023b]. The implementation is also available on GitHub: https://github.com/bergstraesser/ramsey-linear-arithmetics.

ACKNOWLEDGMENTS

We thank anonymous reviewers, Arie Gurfinkel, and Philipp Rümmer for their helpful comments. Moreover, we are grateful to Christoph Haase for discussions on existing quantifier elimination techniques.

Funded by the European Union (ERC, LASD, 101089343 (https://doi.org/10.3030/101089343), and FINABIS, 101077902 (https://doi.org/10.3030/101077902)). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

REFERENCES

Parosh Aziz Abdulla, Mohamed Faouzi Atig, and Jari Stenman. 2012. Dense-Timed Pushdown Automata. In *Proceedings of the 27th Annual IEEE Symposium on Logic in Computer Science, LICS 2012, Dubrovnik, Croatia, June 25-28, 2012*. IEEE Computer Society, 35–44. https://doi.org/10.1109/LICS.2012.15

Parosh Aziz Abdulla, Karlis Cerans, Bengt Jonsson, and Yih-Kuen Tsay. 1996. General Decidability Theorems for Infinite-State Systems. In *Proceedings, 11th Annual IEEE Symposium on Logic in Computer Science, New Brunswick, New Jersey, USA, July 27-30, 1996*. IEEE Computer Society, 313–321. https://doi.org/10.1109/LICS.1996.561359

Rajeev Alur and David L Dill. 1994. A theory of timed automata. *Theoretical computer science* 126, 2 (1994), 183–235.

Pablo Barceló, Chih-Duo Hong, Xuan Bach Le, Anthony W. Lin, and Reino Niskanen. 2019. Monadic Decomposability of Regular Relations. In *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece (LIPIcs, Vol. 132)*, Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 103:1–103:14. https://doi.org/10.4230/LIPIcs.ICALP.2019.103

Sébastien Bardin, Alain Finkel, Jérôme Leroux, and Laure Petrucci. 2008. FAST: acceleration from theory to practice. *Int. J. Softw. Tools Technol. Transf.* 10, 5 (2008), 401–424. https://doi.org/10.1007/s10009-008-0064-3

Sébastien Bardin, Alain Finkel, Jérôme Leroux, and Philippe Schnoebelen. 2005. Flat Acceleration in Symbolic Model Checking. In *Automated Technology for Verification and Analysis, Third International Symposium, ATVA 2005, Taipei, Taiwan, October 4-7, 2005, Proceedings (Lecture Notes in Computer Science, Vol. 3707)*, Doron A. Peled and Yih-Kuen Tsay (Eds.). Springer, 474–488. https://doi.org/10.1007/11562948_35

J. Barwise and S. Feferman (Eds.). 1985. *Model-Theoretic Logics*. Perspectives in Logic, Vol. 8. Association for Symbolic Logic.

Michael Benedikt, Leonid Libkin, Thomas Schwentick, and Luc Segoufin. 2003. Definable relations and first-order query languages over strings. *J. ACM* 50, 5 (2003), 694–751. https://doi.org/10.1145/876638.876642

Pascal Bergsträßer and Moses Ganardi. 2023a. Revisiting Membership Problems in Subclasses of Rational Relations. In *2023 38th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*. 1–14. https://doi.org/10.1109/LICS56636.2023.10175722

Pascal Bergsträßer and Moses Ganardi. 2023b. Revisiting Membership Problems in Subclasses of Rational Relations. *CoRR* abs/2304.11034 (2023). https://doi.org/10.48550/arXiv.2304.11034 arXiv:2304.11034

Pascal Bergsträßer, Moses Ganardi, Anthony W. Lin, and Georg Zetzsche. 2022. Ramsey Quantifiers over Automatic Structures: Complexity and Applications to Verification. In *LICS '22: 37th Annual ACM/IEEE Symposium on Logic in Computer Science, Haifa, Israel, August 2 - 5, 2022*. 28:1–28:14. https://doi.org/10.1145/3531130.3533346

Pascal Bergsträßer, Moses Ganardi, Anthony W. Lin, and Georg Zetzsche. 2023a. Ramsey Quantifiers in Linear Arithmetics. *CoRR* abs/2311.04031 (2023). https://doi.org/10.48550/arXiv.2311.04031 arXiv:2311.04031

Pascal Bergsträßer, Moses Ganardi, Anthony W. Lin, and Georg Zetzsche. 2023b. Ramsey Quantifiers in Linear Arithmetics - Artifact. https://doi.org/10.5281/zenodo.8422415

Tewodros A. Beyene, Corneliu Popeea, and Andrey Rybalchenko. 2013. Solving Existentially Quantified Horn Clauses. In *Computer Aided Verification - 25th International Conference, CAV 2013, Saint Petersburg, Russia, July 13-19, 2013. Proceedings (Lecture Notes in Computer Science, Vol. 8044)*, Natasha Sharygina and Helmut Veith (Eds.). Springer, 869–882. https://doi.org/10.1007/978-3-642-39799-8_61

Nikolaj S. Bjørner, Arie Gurfinkel, Kenneth L. McMillan, and Andrey Rybalchenko. 2015. Horn Clause Solvers for Program Verification. In *Fields of Logic and Computation II - Essays Dedicated to Yuri Gurevich on the Occasion of His 75th Birthday (Lecture Notes in Computer Science, Vol. 9300)*, Lev D. Beklemishev, Andreas Blass, Nachum Dershowitz, Bernd Finkbeiner, and Wolfram Schulte (Eds.). Springer, 24–51. https://doi.org/10.1007/978-3-319-23534-9_2

Nikolaj S. Bjørner, Kenneth L. McMillan, and Andrey Rybalchenko. 2012. Program Verification as Satisfiability Modulo Theories. In *10th International Workshop on Satisfiability Modulo Theories, SMT 2012, Manchester, UK, June 30 - July 1, 2012 (EPiC Series in Computing, Vol. 20)*, Pascal Fontaine and Amit Goel (Eds.). EasyChair, 3–11. https://doi.org/10.29007/1l7f

Michael Blondin, Alain Finkel, Christoph Haase, and Serge Haddad. 2016. Approaching the Coverability Problem Continuously. In *Proc. of the 22nd International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2016) (LNCS, Vol. 9636)*. Springer, 480–496. https://doi.org/10.1007/978-3-662-49674-9_28

Michael Blondin and Christoph Haase. 2017. Logics for continuous reachability in Petri nets and vector addition systems with states. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*. IEEE Computer Society, 1–12. https://doi.org/10.1109/LICS.2017.8005068

Achim Blumensath and Erich Grädel. 2000. Automatic Structures. In *15th Annual IEEE Symposium on Logic in Computer Science, Santa Barbara, California, USA, June 26-29, 2000*. IEEE Computer Society, 51–62. https://doi.org/10.1109/LICS.2000.855755

Bernard Boigelot and Frédéric Herbreteau. 2006. The Power of Hybrid Acceleration. In *Computer Aided Verification, 18th International Conference, CAV 2006, Seattle, WA, USA, August 17-20, 2006, Proceedings (Lecture Notes in Computer Science, Vol. 4144)*, Thomas Ball and Robert B. Jones (Eds.). Springer, 438–451. https://doi.org/10.1007/11817963_40

Bernard Boigelot, Axel Legay, and Pierre Wolper. 2003. Iterating Transducers in the Large (Extended Abstract). In *Computer Aided Verification, 15th International Conference, CAV 2003, Boulder, CO, USA, July 8-12, 2003, Proceedings (Lecture Notes in Computer Science, Vol. 2725)*, Warren A. Hunt Jr. and Fabio Somenzi (Eds.). Springer, 223–235. https://doi.org/10.1007/978-3-540-45069-6_24

Itshak Borosh and Leon Bruce Treybig. 1976. Bounds on positive integral solutions of linear Diophantine equations. *Proc. Amer. Math. Soc.* 55, 2 (1976), 299–304.

Ahmed Bouajjani, Marius Bozga, Peter Habermehl, Radu Iosif, Pierre Moro, and Tomás Vojnar. 2011. Programs with lists are counter automata. *Formal Methods Syst. Des.* 38, 2 (2011), 158–192. https://doi.org/10.1007/s10703-011-0111-7

Ahmed Bouajjani, Rachid Echahed, and Riadh Robbana. 1994. On the Automatic Verification of Systems with Continuous Variables and Unbounded Discrete Data Structures. In *Hybrid Systems II, Proceedings of the Third International Workshop on Hybrid Systems, Ithaca, NY, USA, October 1994 (Lecture Notes in Computer Science, Vol. 999)*, Panos J. Antsaklis, Wolf Kohn, Anil Nerode, and Shankar Sastry (Eds.). Springer, 64–85. https://doi.org/10.1007/3-540-60472-3_4

C. C. Chang and H. J. Keisler. 1990. *Model Theory*. Elsevier.

Lorenzo Clemente and Slawomir Lasota. 2018. Binary Reachability of Timed Pushdown Automata via Quantifier Elimination and Cyclic Order Atoms. In *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic (LIPIcs, Vol. 107)*, Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 118:1–118:14. https://doi.org/10.4230/LIPIcs.ICALP.2018.118

Hubert Comon and Yan Jurski. 1999. Timed Automata and the Theory of Real Numbers. In *CONCUR '99: Concurrency Theory, 10th International Conference, Eindhoven, The Netherlands, August 24-27, 1999, Proceedings (Lecture Notes in Computer Science, Vol. 1664)*, Jos C. M. Baeten and Sjouke Mauw (Eds.). Springer, 242–257. https://doi.org/10.1007/3-540-48320-9_18

Byron Cook, Andreas Podelski, and Andrey Rybalchenko. 2011. Proving program termination. *Commun. ACM* 54, 5 (2011), 88–98. https://doi.org/10.1145/1941487.1941509

Wojciech Czerwinski and Lukasz Orlikowski. 2021. Reachability in Vector Addition Systems is Ackermann-complete. In *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7-10, 2022*. IEEE, 1229–1240. https://doi.org/10.1109/FOCS52979.2021.00120

Zhe Dang. 2001. Binary Reachability Analysis of Pushdown Timed Automata with Dense Clocks. In *Computer Aided Verification, 13th International Conference, CAV 2001, Paris, France, July 18-22, 2001, Proceedings (Lecture Notes in Computer Science, Vol. 2102)*, Gérard Berry, Hubert Comon, and Alain Finkel (Eds.). Springer, 506–518. https://doi.org/10.1007/3-540-44585-4_48

Zhe Dang. 2003. Pushdown timed automata: a binary reachability characterization and safety verification. *Theor. Comput. Sci.* 302, 1-3 (2003), 93–121. https://doi.org/10.1016/S0304-3975(02)00743-0

Zhe Dang and Oscar H. Ibarra. 2002. The Existence of ω-Chains for Transitive Mixed Linear Relations and Its Applications. *Int. J. Found. Comput. Sci.* 13, 6 (2002), 911–936. https://doi.org/10.1142/S0129054102001539

Zhe Dang, Oscar H. Ibarra, Tevfik Bultan, Richard A. Kemmerer, and Jianwen Su. 2000. Binary Reachability Analysis of Discrete Pushdown Timed Automata. In *Computer Aided Verification, 12th International Conference, CAV 2000, Chicago, IL, USA, July 15-19, 2000, Proceedings (Lecture Notes in Computer Science, Vol. 1855)*, E. Allen Emerson and A. Prasad Sistla (Eds.). Springer, 69–84. https://doi.org/10.1007/10722167_9

Zhe Dang, Pierluigi San Pietro, and Richard A. Kemmerer. 2001. On Presburger Liveness of Discrete Timed Automata. In *STACS 2001, 18th Annual Symposium on Theoretical Aspects of Computer Science, Dresden, Germany, February 15-17, 2001, Proceedings (Lecture Notes in Computer Science, Vol. 2010)*, Afonso Ferreira and Horst Reichel (Eds.). Springer, 132–143. https://doi.org/10.1007/3-540-44693-1_12

Leonardo Mendonça de Moura and Nikolaj S. Bjørner. 2008. Z3: An Efficient SMT Solver. In *Tools and Algorithms for the Construction and Analysis of Systems, 14th International Conference, TACAS 2008, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2008, Budapest, Hungary, March 29-April 6, 2008. Proceedings*

(Lecture Notes in Computer Science, Vol. 4963), C. R. Ramakrishnan and Jakob Rehof (Eds.). Springer, 337–340. https://doi.org/10.1007/978-3-540-78800-3_24

Alain Finkel and Ekanshdeep Gupta. 2019a. The Well Structured Problem for Presburger Counter Machines. In *39th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2019, December 11-13, 2019, Bombay, India (LIPIcs, Vol. 150)*, Arkadev Chattopadhyay and Paul Gastin (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 41:1–41:15. https://doi.org/10.4230/LIPIcs.FSTTCS.2019.41

Alain Finkel and Ekanshdeep Gupta. 2019b. The Well Structured Problem for Presburger Counter Machines. *CoRR* abs/1910.02736 (2019). arXiv:1910.02736

Alain Finkel and Philippe Schnoebelen. 2001. Well-structured transition systems everywhere! *Theor. Comput. Sci.* 256, 1-2 (2001), 63–92. https://doi.org/10.1016/S0304-3975(00)00102-X

Jean Baptiste Joseph Fourier. 1826. Solution d'une question particuliere du calcul des inégalités. *Nouveau Bulletin des Sciences par la Société philomatique de Paris* 99 (1826).

Seymour Ginsburg and Edwin H Spanier. 1966. Bounded regular sets. *Proc. Amer. Math. Soc.* 17, 5 (1966), 1043–1049. https://doi.org/10.1090/S0002-9939-1966-0201310-3

Mario Grobler, Leif Sabellek, and Sebastian Siebertz. 2023. Parikh Automata on Infinite Words. *CoRR* abs/2301.08969 (2023). https://doi.org/10.48550/arXiv.2301.08969 arXiv:2301.08969

Stéphane Grumbach, Philippe Rigaux, and Luc Segoufin. 2001. Spatio-Temporal Data Handling with Constraints. *GeoInformatica* 5, 1 (2001), 95–115. https://doi.org/10.1023/A:1011464022461

Shibashis Guha, Ismaël Jecker, Karoliina Lehtinen, and Martin Zimmermann. 2022. Parikh Automata over Infinite Words. In *42nd IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2022, December 18-20, 2022, IIT Madras, Chennai, India (LIPIcs, Vol. 250)*, Anuj Dawar and Venkatesan Guruswami (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 40:1–40:20. https://doi.org/10.4230/LIPIcs.FSTTCS.2022.40

Christoph Haase, Stephan Kreutzer, Joël Ouaknine, and James Worrell. 2009. Reachability in Succinct and Parametric One-Counter Automata. In *CONCUR 2009 - Concurrency Theory, 20th International Conference, CONCUR 2009, Bologna, Italy, September 1-4, 2009. Proceedings (Lecture Notes in Computer Science, Vol. 5710)*, Mario Bravetti and Gianluigi Zavattaro (Eds.). Springer, 369–383. https://doi.org/10.1007/978-3-642-04081-8_25

Matthew Hague and Anthony Widjaja Lin. 2011. Model Checking Recursive Programs with Numeric Data Types. In *Computer Aided Verification - 23rd International Conference, CAV 2011, Snowbird, UT, USA, July 14-20, 2011. Proceedings (Lecture Notes in Computer Science, Vol. 6806)*, Ganesh Gopalakrishnan and Shaz Qadeer (Eds.). Springer, 743–759. https://doi.org/10.1007/978-3-642-22110-1_60

Matthew Hague and Anthony Widjaja Lin. 2012. Synchronisation- and Reversal-Bounded Analysis of Multithreaded Programs with Counters. In *Computer Aided Verification - 24th International Conference, CAV 2012, Berkeley, CA, USA, July 7-13, 2012 Proceedings*. 260–276. https://doi.org/10.1007/978-3-642-31424-7_22

Matthew Hague, Anthony W. Lin, Philipp Rümmer, and Zhilin Wu. 2020. Monadic Decomposition in Integer Linear Arithmetic. In *Automated Reasoning - 10th International Joint Conference, IJCAR 2020, Paris, France, July 1-4, 2020, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 12166)*, Nicolas Peltier and Viorica Sofronie-Stokkermans (Eds.). Springer, 122–140. https://doi.org/10.1007/978-3-030-51074-9_8

Oscar H. Ibarra. 1978. Reversal-Bounded Multicounter Machines and Their Decision Problems. *J. ACM* 25, 1 (1978), 116–133. https://doi.org/10.1145/322047.322058

Oscar H. Ibarra, Jianwen Su, Zhe Dang, Tevfik Bultan, and Richard A. Kemmerer. 2000. Conter Machines: Decidable Properties and Applications to Verification Problems. In *Mathematical Foundations of Computer Science 2000, 25th International Symposium, MFCS 2000, Bratislava, Slovakia, August 28 - September 1, 2000, Proceedings (Lecture Notes in Computer Science, Vol. 1893)*, Mogens Nielsen and Branislav Rovan (Eds.). Springer, 426–435. https://doi.org/10.1007/3-540-44612-5_38

Ranjit Jhala and Rupak Majumdar. 2009. Software model checking. *ACM Comput. Surv.* 41, 4 (2009), 21:1–21:54. https://doi.org/10.1145/1592434.1592438

Felix Klaedtke and Harald Rueß. 2003. Monadic Second-Order Logics with Cardinalities. In *Automata, Languages and Programming, 30th International Colloquium, ICALP 2003, Eindhoven, The Netherlands, June 30 - July 4, 2003. Proceedings (Lecture Notes in Computer Science, Vol. 2719)*, Jos C. M. Baeten, Jan Karel Lenstra, Joachim Parrow, and Gerhard J. Woeginger (Eds.). Springer, 681–696. https://doi.org/10.1007/3-540-45061-0_54

Gabriel Kuper, Leonid Libkin, and Jan Paredaens. 2000. *Constraint Databases*. Springer.

Dietrich Kuske. 2010. Is Ramsey's Theorem omega-automatic?. In *27th International Symposium on Theoretical Aspects of Computer Science, STACS 2010, March 4-6, 2010, Nancy, France (LIPIcs, Vol. 5)*, Jean-Yves Marion and Thomas Schwentick (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 537–548. https://doi.org/10.4230/LIPIcs.STACS.2010.2483

Axel Legay. 2008. T(O)RMC: A Tool for (omega)-Regular Model Checking. In *Computer Aided Verification, 20th International Conference, CAV 2008, Princeton, NJ, USA, July 7-14, 2008, Proceedings (Lecture Notes in Computer Science, Vol. 5123)*, Aarti Gupta and Sharad Malik (Eds.). Springer, 548–551. https://doi.org/10.1007/978-3-540-70545-1_52

K. Rustan M. Leino. 2023. Program Proofs. (2023).

Jérôme Leroux. 2021. The Reachability Problem for Petri Nets is Not Primitive Recursive. In *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7-10, 2022*. IEEE, 1241–1252. https://doi.org/10.1109/FOCS52979.2021.00121

Jérôme Leroux and Sylvain Schmitz. 2019. Reachability in Vector Addition Systems is Primitive-Recursive in Fixed Dimension. In *34th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2019, Vancouver, BC, Canada, June 24-27, 2019*. IEEE, 1–13. https://doi.org/10.1109/LICS.2019.8785796

Xie Li, Taolue Chen, Zhilin Wu, and Mingji Xia. 2020. Computing Linear Arithmetic Representation of Reachability Relation of One-Counter Automata. In *Dependable Software Engineering. Theories, Tools, and Applications - 6th International Symposium, SETTA 2020, Guangzhou, China, November 24-27, 2020, Proceedings (Lecture Notes in Computer Science, Vol. 12153)*, Jun Pang and Lijun Zhang (Eds.). Springer, 89–107. https://doi.org/10.1007/978-3-030-62822-2_6

Leonid Libkin. 2003. Variable independence for first-order definable constraints. *ACM Trans. Comput. Log.* 4, 4 (2003), 431–451. https://doi.org/10.1145/937555.937557

Richard Lipton. 1976. The reachability problem is exponential-space hard. *Yale University, Department of Computer Science, Report* 62 (1976).

Zohar Manna and Amir Pnueli. 1970. Formalization of Properties of Functional Programs. *J. ACM* 17, 3 (1970), 555–569. https://doi.org/10.1145/321592.321606

Oliver Markgraf, Daniel Stan, and Anthony W. Lin. 2021. Learning Union of Integer Hypercubes with Queries - (with Applications to Monadic Decomposition). In *Computer Aided Verification - 33rd International Conference, CAV 2021, Virtual Event, July 20-23, 2021, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 12760)*, Alexandra Silva and K. Rustan M. Leino (Eds.). Springer, 243–265. https://doi.org/10.1007/978-3-030-81688-9_12

Greg Nelson and Derek C. Oppen. 1980. Fast Decision Procedures Based on Congruence Closure. *J. ACM* 27, 2 (1980), 356–364. https://doi.org/10.1145/322186.322198

Andreas Podelski and Andrey Rybalchenko. 2004. Transition Invariants. In *19th IEEE Symposium on Logic in Computer Science (LICS 2004), 14-17 July 2004, Turku, Finland, Proceedings*. IEEE Computer Society, 32–41. https://doi.org/10.1109/LICS.2004.1319598

Mojżesz Presburger. 1929. Über die Vollständigkeit eines gewissen Systems der Arithmetik ganzer Zahlen, in welchem die Addition als einzige Operation hervortritt. *Comptes Rendus du I congres de Mathematiciens de Pays Slaves* (1929).

Shaz Qadeer and Jakob Rehof. 2005. Context-Bounded Model Checking of Concurrent Software. In *Tools and Algorithms for the Construction and Analysis of Systems, 11th International Conference, TACAS 2005, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2005, Edinburgh, UK, April 4-8, 2005, Proceedings (Lecture Notes in Computer Science, Vol. 3440)*, Nicolas Halbwachs and Lenore D. Zuck (Eds.). Springer, 93–107. https://doi.org/10.1007/978-3-540-31980-1_7

Karin Quaas, Mahsa Shirmohammadi, and James Worrell. 2017. Revisiting reachability in timed automata. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*. IEEE Computer Society, 1–12. https://doi.org/10.1109/LICS.2017.8005098

Charles Rackoff. 1978. The covering and boundedness problems for vector addition systems. *Theoretical Computer Science* 6, 2 (1978), 223–231.

F. P. Ramsey. 1930. On a Problem of Formal Logic. *Proceedings of the London Mathematical Society* s2-30, 1 (01 1930), 264–286. https://doi.org/10.1112/plms/s2-30.1.264

James H Schmerl and Stephen G Simpson. 1982. On the role of Ramsey quantifiers in first order arithmetic1. *The Journal of Symbolic Logic* 47, 2 (1982), 423–435.

Robert E. Shostak. 1984. Deciding Combinations of Theories. *J. ACM* 31, 1 (1984), 1–12. https://doi.org/10.1145/2422.322411

Eduardo D. Sontag. 1985. Real Addition and the Polynomial Hierarchy. *Inform. Process. Lett.* 20, 3 (April 1985), 115–120. https://doi.org/10.1016/0020-0190(85)90076-6

Anthony Widjaja To. 2009. Model Checking FO(R) over One-Counter Processes and beyond. In *Computer Science Logic, 23rd international Workshop, CSL 2009, 18th Annual Conference of the EACSL, Coimbra, Portugal, September 7-11, 2009. Proceedings*. 485–499. https://doi.org/10.1007/978-3-642-04027-6_35

Anthony Widjaja To and Leonid Libkin. 2008. Recurrent Reachability Analysis in Regular Model Checking. In *Logic for Programming, Artificial Intelligence, and Reasoning, 15th International Conference, LPAR 2008, Doha, Qatar, November 22-27, 2008. Proceedings*. 198–213. https://doi.org/10.1007/978-3-540-89439-1_15

Margus Veanes, Nikolaj S. Bjørner, Lev Nachmanson, and Sergey Bereg. 2017. Monadic Decomposition. *J. ACM* 64, 2 (2017), 14:1–14:28. https://doi.org/10.1145/3040488

Volker Weispfenning. 1997. Complexity and Uniformity of Elimination in Presburger Arithmetic. In *Proceedings of the 1997 International Symposium on Symbolic and Algebraic Computation, ISSAC 1997, Maui, Hawaii, USA, July 21-23, 1997*, Bruce W. Char, Paul S. Wang, and Wolfgang Küchlin (Eds.). ACM, 48–53. https://doi.org/10.1145/258726.258746

Volker Weispfenning. 1999. Mixed Real-Integer Linear Quantifier Elimination. In *Proceedings of the 1999 International Symposium on Symbolic and Algebraic Computation, ISSAC '99, Vancouver, B.C., Canada, July 29-31, 1999*, Keith O. Geddes,

Bruno Salvy, and Samuel S. Dooley (Eds.). ACM, 129–136. https://doi.org/10.1145/309831.309888
H Paul Williams. 1986. Fourier's method of linear programming and its dual. *The American mathematical monthly* 93, 9 (1986), 681–695. https://doi.org/10.2307/2322281

Received 2023-07-11; accepted 2023-11-07