A Practical Tool for Analyzing the Distribution of Numerical Values*

Roberto Bagnara[1], Katy Dobson[2], Patricia M. Hill[2], Matthew Mundell[2], and Enea Zaffanella[1]

[1] Department of Mathematics, University of Parma, Italy
{bagnara,zaffanella}@cs.unipr.it
[2] School of Computing, University of Leeds, UK
{katyd,hill,mattm}@comp.leeds.ac.uk

Abstract. The abstract domain of *rational grids* (or lattices) is a domain that is able to represent sets of equally spaced points and hyperplanes over an n-dimensional vector space. Such a domain is useful for the static analysis of the patterns of distribution of the values that program variables can take. This paper explores how this domain may be used in program analysis, describing grid operations such as affine image, affine preimage and widenings needed by such an application. The paper also shows how any grid may be approximated by a less precise non-relational grid and describes how such an approximation can be computed. Illustrative examples show how the domain may be used in the analysis of programs containing simple assignment statements, while loops and recursive procedures.

1 Introduction

The static analysis of numerical information about the values program variables can take is a challenging problem that has led to considerable research in both the theory and its practical realization. We distinguish between two kinds of numerical information: outer *limits* (or bounds within which the values must lie) and the pattern of *distribution* of these values. Both kinds of information have important applications. For example, in the field of automatic program verification, limit information is crucial to ensure that array accesses are within bounds, while distribution information is what is required to ensure that the external memory accesses obey the alignment restriction imposed by the host architecture. In the field of program optimization, limit information can be used to compile out various kinds of run-time tests, whereas distribution information enables several transformations for efficient parallel execution as well as optimizations that enhance cache behavior.

* This work has been partly supported by EPSRC project EP/C520726/1 "Numerical Domains for Software Analysis," by MIUR project "AIDA—Abstract Interpretation: Design and Applications," and by a Royal Society (UK) International Joint Project (ESEP) award.

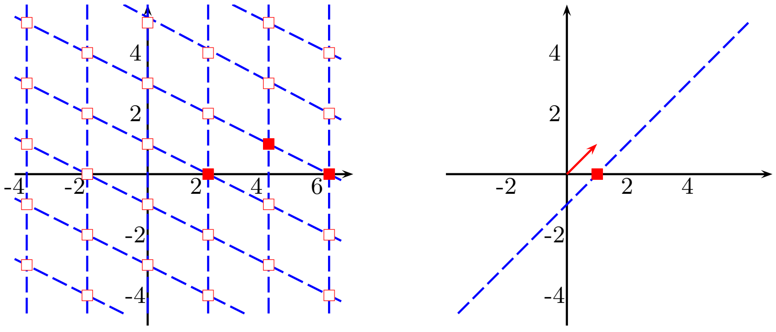
Both limit and distribution information often come in a *relational* form; for instance, the outer limits or the pattern of possible values of one variable may depend on the values of one or more other variables. Domains that can capture relational information are generally much more complex than domains that do not have this capability; in exchange they usually offer significantly more precision, often important for the overall performance of the client application. Relational limit information can be captured, among other possibilities, by means of *polyhedral domains*, that is, domains that represent regions of some n-dimensional vector space bounded by a finite set of hyperplanes [11]. While several polyhedral domains have been proposed and that of convex polyhedra —the most popular relational polyhedral domain— has been thoroughly researched and widely used, relational domains for representing the (linear) distribution of numerical values have been less well researched. For instance, previous investigations have hardly touched on the issue of widening operators; and have failed to support non-relational approximation. Moreover, as far as we know and at the time of writing, there is no available implementation providing all the operations needed by a relational abstract domain for distribution information. This, in spite of the fact that previous research has shown that a knowledge about (discrete) distribution of numerical information, especially when combined with that of the limit information, can significantly improve the quality of the analysis results [1].

This is the second of two papers designed to close this gap and provide a complete account of the relational domain of *rational grids*; a domain for capturing numerical distribution information. The two main representations; how these can be reduced and how to convert from one to another; and the basic operations on them is already described in the first of these papers. This second paper on the grid domain provides further grid operations; for widening, for computing the affine image and preimage, and for computing the best approximation of a grid by a non-relational grid. A preliminary combined version of both these papers is available as a technical report [2].

Grids in a Nutshell. Figure 1 illustrates two ways of describing a grid; either by means of a finite set of congruence relations that all grid points must satisfy (given by the dashed lines) or by means of a finite set of generating vectors used for constructing the grid points and lines (given by the filled squares and arrow).

The squares in Figure 1(a) illustrate the grid $\mathcal{L}$ indicating possible values of integer variables x and y resulting from the execution of the program fragment in Figure 2 for any value of `m`.

The congruence relations $x = 0 \pmod 2$ and $x + 2y = 2 \pmod 4$ are represented by the vertical dashed lines and sloping lines, respectively. The set of congruence relations $\mathcal{C} = \bigl\{x = 0 \pmod 2, x + 2y = 2 \pmod 4\bigr\}$, called a *congruence system*, is said to *describe* $\mathcal{L}$. The filled squares mark the points $\boldsymbol{p}_1 = \left(\begin{smallmatrix}2\\0\end{smallmatrix}\right)$, $\boldsymbol{p}_2 = \left(\begin{smallmatrix}6\\0\end{smallmatrix}\right)$ and $\boldsymbol{p}_3 = \left(\begin{smallmatrix}4\\1\end{smallmatrix}\right)$ while all the squares (both filled and unfilled) mark points $\boldsymbol{v} = \pi_1\boldsymbol{p}_1 + \pi_2\boldsymbol{p}_2 + \pi_3\boldsymbol{p}_3$, where $\pi_1, \pi_2, \pi_3 \in \mathbb{Z}$ and $\pi_1 + \pi_2 + \pi_3 = 1$. The set of *points* $P = \{\boldsymbol{p}_1, \boldsymbol{p}_2, \boldsymbol{p}_3\}$ is said to *generate* $\mathcal{L}$. Some of these generating points can be replaced by *parameters* that give the direction and spac-



(a) The grid $\mathcal{L}$ (b) The grid $\mathcal{L}'$

Fig. 1. Two grids in $\mathbb{R}^2$

ing for the neighboring points. Specifically, by subtracting the point $\boldsymbol{p}_1$ from each of the other two generating points $\boldsymbol{p}_2$ and $\boldsymbol{p}_3$, we obtain the parameters $\boldsymbol{q}_2 = \left(\begin{smallmatrix}4\\0\end{smallmatrix}\right)$ and $\boldsymbol{q}_3 = \left(\begin{smallmatrix}2\\1\end{smallmatrix}\right)$ for $\mathcal{L}$. It follows that each point $\boldsymbol{v} \in \mathcal{L}$ can be written as $\boldsymbol{v} = \boldsymbol{p}_1 + \pi_2 \boldsymbol{q}_2 + \pi_3 \boldsymbol{q}_3$ for some $\pi_2, \pi_3 \in \mathbb{Z}$.

The dashed line in Figure 1(b) illustrates the grid $\mathcal{L}'$ defining the line $x = y + 1$ and marks the vectors of values of the real variables x and y after an assignment `x := y + 1`, assuming that nothing is known about the value of y. As equalities are congruences modulo 0, the set $\mathcal{C}' = \{x - y = 1\}$ is also called a congruence system and describes $\mathcal{L}'$. Observe that the grid $\mathcal{L}'$ consists of all points that can be obtained as $\lambda\boldsymbol{\ell} + \boldsymbol{p}'$, for any $\lambda \in \mathbb{R}$, where $\boldsymbol{\ell} = \left(\begin{smallmatrix}1\\1\end{smallmatrix}\right)$ and $\boldsymbol{p}' = \left(\begin{smallmatrix}1\\0\end{smallmatrix}\right)$; the vector $\boldsymbol{\ell}$ defines the direction of the *line* and the vector $\boldsymbol{p}'$ its position (illustrated in Figure 1(b) by the arrow and the filled square, respectively).

```
x := 2; y := 0;   (P1)
for i := 1 to m   (P2)
  if ... then
    x := x + 4    (P3)
  else
    x := x + 2;
    y := y + 1    (P4)
  endif           (P5)
endfor
```

Fig. 2. Fragment based on an example in [11]

From what we have just seen, any grid can be represented both by a congruence system and by a *generator system*. The latter may consist of three components: a set of lines, a set of parameters and a set of points. For instance, the triples $\mathcal{G}_1 = \bigl(\varnothing, \varnothing, P\bigr)$ and $\mathcal{G}_2 = \bigl(\varnothing, \{\boldsymbol{q}_2, \boldsymbol{q}_3\}, \{\boldsymbol{p}_1\}\bigr)$ are both generator systems for $\mathcal{L}$ while the triple $\mathcal{G}' = \bigl(\{\boldsymbol{\ell}\}, \varnothing, \{\boldsymbol{p}'\}\bigr)$ is a generator system for $\mathcal{L}'$.

Contributions and Related Work. The paper shows that the domain of rational grids is a practical tool for program analyzers and describes the specific operations required for program analysis and similar applications. The particular contributions are as follows.

We show, by means of an example, how *rational* grids can be used to model the effect of (possibly recursive) procedures and hence be used for interprocedural program analysis where the input parameters are not known. We observe that, previously, examples of interprocedural analysis for distribution information [20] assume that the input values are known in advance of the analysis.

Affine image and preimage operators can be used to capture the effect of assignment statements in a program when the expression is linear although, as noted by Müller-Olm and Seidl in [19], analyses that use affine spaces for approximating the semantics of procedures are not sufficiently precise to detect all valid affine relations for programs with procedures. Here we specify, for the domain of grids, the affine image and preimage operators for a *single update* where only one dimension is modified. We illustrate how these can capture the behavior of assignments in (possibly recursive) procedures, comparing our technique with the interprocedural analysis in [20]. Moreover, we describe generalized forms of these operators which, though still having the necessary closure property for grids, map individual points to (possibly infinite) sets of points and show how they can capture the result of loops containing simple assignment statements.

In order to dynamically manage the complexity/precision trade-off of program analysis, the analyzer should be able to dynamically switch (in response to timeouts or driven by heuristics) to a simpler analysis domain, with a more compact representation and more efficient operations. The first proposal for a numerical domain for representing distribution information was for a domain (we call here *rectilinear*), that can be described by arithmetic congruence relations [13] over the integers. This is less precise than *rational* grids (that is, grids described by the relational form), but most of its operations have complexity $\mathrm{O}(n)$. Thus, by providing algorithms for conversion from a grid to its *rectilinear* approximation and vice-versa, we make dynamic switching between relational and non-relational domains a practical option. We also show how rectilinear grids can be represented using the standard domain of intervals and thereby able to exploit all the advantages of this domain. Apart from using rectilinear grids for approximating the relational grid, we could consider intermediate *weakly relational* domains; that is, domains that lie between these two extremes. In [18], Miné describes a weakly relational domain defined by congruences of the form $x - y = a \pmod{b}$ where a and b are rationals.

It was observed by Granger in [14], that rational grids do not satisfy the ascending chain condition; so, to guarantee termination of the analysis, a widening operation is required. In [14, Proposition 10], a widening is given for rectilinear rational grids that returns a line parallel to an axis whenever the modulus for that dimension changes. It is then proposed that a generalized form of this could be used as a widening for relational grids; however, exactly how this is to be done is unclear. In this paper, we define two possible generalizations which come with simple syntactic checks that have efficient implementations.

The grid domain as described here and in [2] is implemented in the Parma Polyhedra Library (PPL) [5, 6]; the source code is publicly available in the PPL's CVS repository (see the PPL web site at `http://www.cs.unipr.it/ppl/` for more information). Apart from the PPL, the only other freely available library providing numerical domains for applications in static analysis and computer-aided verification including a domain similar to that for grids, is PolyLib [17].

This domain is based on the integer lattice domain proposed by Ancourt [1] (which, together with polyhedra, forms the compound domain of $\mathbb{Z}$-polyhedra). Only the generator representation is supported and some operations, instead of computing a single grid, return *sets* of grids while additional operators are provided to manipulate and simplify these sets. Image and preimage operators are supported by PolyLib although, as the integral lattice must be full dimensional, only invertible image operators are allowed.

Plan of the Paper. Section 2 gives the required notations and preliminary concepts; Section 3 re-introduces a grid together with its congruence and generator representations; Sections 4, 5 and 6 introduce, in turn: the affine image and preimage operations; an operation for approximating an arbitrary grid by a non-relational grid; and two grid widening operations. Section 7 concludes the main body of the paper.[3]

2 Preliminaries

The *cardinality* of a set S is denoted by $\# S$. The set of integers is denoted by $\mathbb{Z}$, rationals by $\mathbb{Q}$ and reals by $\mathbb{R}$. The complexities given in the paper assume a unit cost for every arithmetic operation; we take the computation of the greatest common divisor of a pair of numbers $a, b \in \mathbb{R}$ to be a single operation. Given sets X, Y and any relation $R \subseteq X \times Y$, the *image* for R on a subset A of X is $\{\, y \in Y \mid \exists x \in A \mathrel{.} (x, y) \in R \,\}$, and the *preimage* for R on a subset B of Y is $\{\, x \in X \mid \exists y \in B \mathrel{.} (x, y) \in R \,\}$.

For each $i \in \{1, \dots, n\}$, v_i denotes the i-th component of (column) vector $\boldsymbol{v} \in \mathbb{R}^n$; and the *scalar product* of vectors $\boldsymbol{v}$ and $\boldsymbol{w} \in \mathbb{R}^n$, denoted by $\langle \boldsymbol{v}, \boldsymbol{w} \rangle$, is the real number $\boldsymbol{v}^{\mathrm{T}} \boldsymbol{w} = \sum_{i=1}^{n} v_i w_i$. The notation $\mathrm{piv}_{<}(\boldsymbol{v})$ (resp., $\mathrm{piv}_{>}(\boldsymbol{v})$) denotes the maximum (resp., minimum) index i such that $v_i \neq 0$; if $\boldsymbol{v} = \boldsymbol{0}$, we define $\mathrm{piv}_{<}(\boldsymbol{v}) := 0$ and $\mathrm{piv}_{>}(\boldsymbol{v}) := n + 1$. We write $\boldsymbol{v} \Uparrow \boldsymbol{v}'$, if $\mathrm{piv}_{<}(\boldsymbol{v}) = \mathrm{piv}_{<}(\boldsymbol{v}') = k$ and, if $k \geq 1$, $v_k = v'_k$ and $\boldsymbol{v} \Downarrow \boldsymbol{v}'$, if $\mathrm{piv}_{>}(\boldsymbol{v}) = \mathrm{piv}_{>}(\boldsymbol{v}') = k$ and, if $k \leq n$, $v_k = v'_k$. The vector $\boldsymbol{e}_i \in \mathbb{R}^n$ has 1 in the i-th position and 0 in every other position. The set $S = \{\boldsymbol{v}_1, \dots, \boldsymbol{v}_k\} \subseteq \mathbb{R}^n$ is *affinely independent* if, for all $\boldsymbol{\lambda} \in \mathbb{R}^k$, $\boldsymbol{\lambda} = \boldsymbol{0}$ is the only solution of the set of equations $\bigl\{ \sum_{i=1}^{k} \lambda_i \boldsymbol{v}_i = \boldsymbol{0},\ \sum_{i=1}^{k} \lambda_i = 0 \bigr\}$. For all $\boldsymbol{\lambda} \in \mathbb{R}^k$, the vector $\boldsymbol{v} = \sum_{j=1}^{k} \lambda_j \boldsymbol{v}_j$ is said to be a *linear* combination of S. This combination is *affine*, if $\sum_{j=1}^{k} \lambda_j = 1$; and *integral*, if $\boldsymbol{\lambda} \in \mathbb{Z}^k$. The set of all linear (resp., affine, integral, integral and affine) combinations of S is denoted by linear.hull (resp., affine.hull(S), int.hull(S), int.affine.hull(S)).

Let $H = (\boldsymbol{h}_1, \dots, \boldsymbol{h}_m) \in \mathbb{R}^{n \times m}$ be a matrix. Then H^{T} denotes the *transposition* of H, for each $i \in \{1, \dots, n\}$, H_i denotes the i-th row and, for each $j \in \{1, \dots, m\}$, H_{ij} denotes the ij-th component. We sometimes consider H as the set of vectors $\{\boldsymbol{h}_1, \dots, \boldsymbol{h}_m\} \subseteq \mathbb{R}^n$. As any vector in $\mathbb{R}^{n \times 1}$ is a matrix, the matrix operations are also applicable to vectors.

[3] For referees: the proofs of all the stated results are available in the Appendix.

For any $a, b, f \in \mathbb{R}$, $a \equiv_f b$ denotes the *congruence* $\exists \mu \in \mathbb{Z} \, . \, a - b = \mu f$. Let $\mathbb{S} \in \{\mathbb{Q}, \mathbb{R}\}$. For each vector $\boldsymbol{a} \in \mathbb{S}^n$ and scalars $b, f \in \mathbb{S}$, the notation $\langle \boldsymbol{a}, \boldsymbol{x} \rangle \equiv_f b$ stands for the *linear congruence relation in* $\mathbb{S}^n$ defined by the set of vectors $\{ \boldsymbol{v} \in \mathbb{R}^n \mid \exists \mu \in \mathbb{Z} \, . \, \langle \boldsymbol{a}, \boldsymbol{v} \rangle = b + \mu f \}$; when $f \neq 0$, the relation is said to be *proper*; $\langle \boldsymbol{a}, \boldsymbol{x} \rangle \equiv_0 b$ (i.e., when $f = 0$) denotes the equality $\langle \boldsymbol{a}, \boldsymbol{x} \rangle = b$. Thus, provided $\boldsymbol{a} \neq \boldsymbol{0}$, the relation $\langle \boldsymbol{a}, \boldsymbol{x} \rangle \equiv_f b$ defines the set of affine hyperplanes $\{ (\langle \boldsymbol{a}, \boldsymbol{x} \rangle = b + \mu f) \mid \mu \in \mathbb{Z} \}$; when $\boldsymbol{a} = \boldsymbol{0}$, we assume that $b \neq 0$; if $b \equiv_f 0$, $\langle \boldsymbol{0}, \boldsymbol{x} \rangle \equiv_f b$ defines the universe $\mathbb{R}^n$ and the empty set, otherwise. Any vector that satisfies $\langle \boldsymbol{a}, \boldsymbol{x} \rangle = b + \mu f$ for some $\mu \in \mathbb{Z}$ is said to *satisfy* the relation $\langle \boldsymbol{a}, \boldsymbol{x} \rangle \equiv_f b$. Congruence relations in $\mathbb{S}^n$, such as $\langle \boldsymbol{a}, \boldsymbol{x} \rangle \equiv_1 b$ and $\langle 2\boldsymbol{a}, \boldsymbol{x} \rangle \equiv_2 2b$, defining the same hyperplanes are considered equivalent. The pivot notation is extended to congruences: if $\beta = (\langle \boldsymbol{a}, \boldsymbol{x} \rangle \equiv_f a_0)$ then $\mathrm{piv}_<(\beta) := \mathrm{piv}_<(\boldsymbol{a})$; if $\gamma = (\langle \boldsymbol{c}, \boldsymbol{x} \rangle \equiv_g c_0)$ and $g\boldsymbol{a} \Uparrow f\boldsymbol{c}$, then we write $\beta \Uparrow \gamma$; so that β and γ are either both equalities or both proper congruences.

3 Rational Grids

In this section we recall the basic definitions for rational grids as described in [2]. A *congruence system in* $\mathbb{Q}^n$ is a finite set of congruence relations $\mathcal{C}$ in $\mathbb{Q}^n$. As we do not distinguish between syntactically different congruences defining the same set of vectors, we can assume that all proper congruences in $\mathcal{C}$ have modulus 1.

Definition 1. *Let $\mathcal{C}$ be a congruence system in $\mathbb{R}^n$. If $\mathcal{L}$ is the set of vectors in $\mathbb{R}^n$ that satisfy all the congruences in $\mathcal{C}$, we say that $\mathcal{L}$ is a* rational grid described by *a congruence system $\mathcal{C}$ in $\mathbb{Q}^n$. We also say that $\mathcal{C}$* is a congruence system for $\mathcal{L}$ *and write $\mathcal{L} = \mathrm{gcon}(\mathcal{C})$. If $\mathrm{gcon}(\mathcal{C}) = \varnothing$, then we say that $\mathcal{C}$ is* inconsistent. *The* grid domain $\mathbb{G}_n$ *is the set of all rational grids in $\mathbb{R}^n$ ordered by the set inclusion relation, so that the empty set $\varnothing$ and the vector space $\mathbb{R}^n$ are the bottom and top elements of $\mathbb{G}_n$ respectively.*

The vector space $\mathbb{R}^n$ is also called the *universe* grid. In set theoretical terms, $\mathbb{G}_n$ is a *lattice* under set inclusion. Many algorithms require the congruence systems not only to have a minimal number of elements but also such that the coefficients of (a permutation of) the congruences can form a triangular matrix.

Definition 2. *Suppose $\mathcal{C}$ is a congruence system in $\mathbb{Q}^n$. Then we say that $\mathcal{C}$ is in* minimal form *if either $\mathcal{C} = \{\langle \boldsymbol{0}, \boldsymbol{x} \rangle \equiv_0 1\}$ or $\mathcal{C}$ is consistent and, for each congruence $\beta = (\langle \boldsymbol{a}, \boldsymbol{x} \rangle \equiv_f b) \in \mathcal{C}$, the following hold:*

1. *if $\mathrm{piv}_<(\beta) = k$, then $k > 0$ and $a_k > 0$;*
2. *for all $\beta' \in \mathcal{C} \setminus \{\beta\}$, $\mathrm{piv}_<(\boldsymbol{\beta'}) \neq \mathrm{piv}_<(\boldsymbol{\beta})$.*

Let $\mathcal{L}$ be a grid in $\mathbb{G}_n$. Then a vector $\boldsymbol{p} \in \mathcal{L}$ is called a *point* of $\mathcal{L}$; a vector $\boldsymbol{q} \in \mathbb{R}^n \setminus \{\boldsymbol{0}\}$ is called a *parameter* of $\mathcal{L}$ if $\mathcal{L} \neq \varnothing$ and $\boldsymbol{p} + \mu\boldsymbol{q} \in \mathcal{L}$, for all points $\boldsymbol{p} \in \mathcal{L}$ and all $\mu \in \mathbb{Z}$; a vector $\boldsymbol{\ell} \in \mathbb{R}^n \setminus \{\boldsymbol{0}\}$ is called a *line* of $\mathcal{L}$ if $\mathcal{L} \neq \varnothing$ and $\boldsymbol{p} + \lambda\boldsymbol{\ell} \in \mathcal{L}$, for all points $\boldsymbol{p} \in \mathcal{L}$ and all $\lambda \in \mathbb{R}$.

If L, Q and P are each finite subsets of $\mathbb{Q}^n$ and

$$\mathcal{L} := \text{linear.hull}(L) + \text{int.hull}(Q) + \text{int.affine.hull}(P)$$

where the symbol '+' denotes the Minkowski's sum,[4] then $\mathcal{L} \in \mathbb{G}_n$ is a rational grid. The 3-tuple (L, Q, P) is said to be a *generator system* in $\mathbb{Q}^n$ for $\mathcal{L}$ and we write $\mathcal{L} = \text{ggen}(L, Q, P)$. Note that the grid $\mathcal{L} = \text{ggen}(L, Q, P) = \varnothing$ if and only if the set of points $P = \varnothing$. If $P \neq \varnothing$, then $\mathcal{L} = \text{ggen}(L, \varnothing, Q_{\boldsymbol{p}} \cup P)$ where, for some $\boldsymbol{p} \in P$, $Q_{\boldsymbol{p}} = \{\, \boldsymbol{p} + \boldsymbol{q} \in \mathbb{R}^n \mid \boldsymbol{q} \in Q \,\}$. As for congruence systems, for many procedures in the implementation, it is useful if the generator systems have a minimal number of elements.

Definition 3. *Suppose $\mathcal{G} = (L, Q, P)$ is a generator system in $\mathbb{Q}^n$. Then we say that $\mathcal{G}$ is in* minimal form *if either $L = Q = P = \varnothing$ or $\# P = 1$ and, for each generator $\boldsymbol{v} \in L \cup Q$, the following hold:*

1. *if $\text{piv}_{>}(\boldsymbol{v}) = k$, then $v_k > 0$;*
2. *for all $\boldsymbol{v}' \in (L \cup Q) \setminus \{\boldsymbol{v}\}$, $\text{piv}_{>}(\boldsymbol{v}') \neq \text{piv}_{>}(\boldsymbol{v})$.*

We have shown that any grid $\mathcal{L}$ can be described by using a congruence system $\mathcal{C}$ and also generated by a generator system $\mathcal{G}$. For the same reasons as for the polyhedral domain, it is useful to represent the grid $\mathcal{L}$ by the *double description* $(\mathcal{C}, \mathcal{G})$ since having easy access to both the representations is assumed in the implementation of many operators including the affine transformations and widenings described here. For any pair of grids $\mathcal{L}_1, \mathcal{L}_2 \in \mathbb{G}_n$, the *intersection* of $\mathcal{L}_1$ and $\mathcal{L}_2$, defined as the set intersection $\mathcal{L}_1 \cap \mathcal{L}_2$, is the largest grid included in both $\mathcal{L}_1$ and $\mathcal{L}_2$; similarly, the *grid join* of $\mathcal{L}_1$ and $\mathcal{L}_2$, denoted by $\mathcal{L}_1 \oplus \mathcal{L}_2$, is the smallest grid that includes both $\mathcal{L}_1$ and $\mathcal{L}_2$. In theoretical terms, the intersection and grid join defined above are the binary *meet* and *join* on the lattice $\mathbb{G}_n$.

Example 1. Consider the procedure in Figure 3, the running example of [20]. The effect of calling `q` with the pair of variables $(\mathtt{x}, \mathtt{y})$ set to the pair of values (a, b) will be to bind the vector $(\mathtt{x}, \mathtt{y})^{\mathrm{T}}$ to the vectors $(a, b)^{\mathrm{T}}$, $(15a, 18a + b)^{\mathrm{T}}$, $(225a, 282a + b)^{\mathrm{T}}$ and so on. Note that using integral grids, without knowing the values a and b, we cannot perform any of the grid operations since all the values are parametric on the pair (a, b). We therefore need a grid where the values capture the effect of the procedure but do not refer explicitly to the values (a, b). Since the effect of the procedure in the case $a = 0$ is trivial, we assume $a \neq 0$. Consider a grid with variables (u, v, w) where $u = \frac{x}{a}$, $v = \frac{y}{a} - \frac{b}{a}$ and $w = \frac{b}{a}$. Then $(1, 0, \frac{b}{a})^{\mathrm{T}}$ will be the initial vector for (u, v, w) in any call to `q`. This will result in a vector of values represented as a point in a grid $\mathcal{L}_i = \text{ggen}\bigl((L, \varnothing, P_i)\bigr)$, where i is the number of iterations through the body of the procedure; the

```
q(var x, var y)
  if ... then
    x := 3*x
    y := x + y
    q(x, y)
    x := 5*x
    y := x + y
  endif
```

Fig. 3. A recursive procedure

[4] This is defined, for each $S, T \subseteq \mathbb{R}^n$, by $S + T := \{\, \boldsymbol{s} + \boldsymbol{t} \in \mathbb{R}^n \mid \boldsymbol{s} \in S, \boldsymbol{t} \in T \,\}$.

singleton set of lines $L = \{(0,0,1)^{\mathrm{T}}\}$ represents the fact that there is no information about the initial value $\frac{b}{a}$ for w and, for the first three iterations, the sets of points are given by $P_0 := \{(1,0,0)^{\mathrm{T}}\}$, $P_1 := \{(15,18,0)^{\mathrm{T}}\}$ and $P_2 := \{(225,282,0)^{\mathrm{T}}\}$. Letting $\mathcal{M} := \mathcal{L}_0 \oplus \mathcal{L}_1 \oplus \mathcal{L}_2$, we have $\mathcal{M} = \mathrm{ggen}\big((L, \varnothing, P)\big)$ where $P = \{(1,0,0)^{\mathrm{T}}, (15,18,0)^{\mathrm{T}}, (225,282,0)^{\mathrm{T}}\}$. Converting this to the congruence representation we obtain $\mathcal{M} = \mathrm{gcon}\big(\{u \equiv_{14} 1, v \equiv_6 0\}\big)$.

4 Affine Images and Preimages.

Affine transformations for the vector space $\mathbb{R}^n$ will map hyperplanes to hyperplanes and preserve intersection properties between hyperplanes; such transformations can be represented by matrices in $\mathbb{R}^{n \times n}$. It follows that the set $\mathbb{G}_n$ is closed under the set of all affine transformations for $\mathbb{R}^n$. Simple and useful linear affine transformations for numerical domains, including the grids, are provided by the 'single update' affine image and affine preimage operators.

Given a grid $\mathcal{L} \in \mathbb{G}_n$, a variable x_k and linear expression $e = \langle \boldsymbol{a}, \boldsymbol{x} \rangle + b$ with coefficients in $\mathbb{Q}$, the *affine image operator* $\phi(\mathcal{L}, x_k, e)$ maps the grid $\mathcal{L}$ to

$$\Big\{ \big(p_1, \ldots, p_{k-1}, \langle \boldsymbol{a}, \boldsymbol{p} \rangle + b, p_{k+1}, \ldots, p_n\big)^{\mathrm{T}} \in \mathbb{R}^n \;\Big|\; \boldsymbol{p} \in \mathcal{L} \Big\}.$$

Conversely, the *affine preimage operator* $\phi^{-1}(\mathcal{L}, x_k, e)$ maps the grid $\mathcal{L}$ to

$$\Big\{ \boldsymbol{p} \in \mathbb{R}^n \;\Big|\; \big(p_1, \ldots, p_{k-1}, \langle \boldsymbol{a}, \boldsymbol{p} \rangle + b, p_{k+1}, \ldots, p_n\big)^{\mathrm{T}} \in \mathcal{L} \Big\}.$$

Observe that the affine image $\phi(\mathcal{L}, x_k, e)$ and preimage $\phi^{-1}(\mathcal{L}, x_k, e)$ are invertible if and only if the coefficient a_k in the vector $\boldsymbol{a}$ is non-zero.

Example 2. Consider again Example 1 and the recursive procedure in Figure 3. Then `x := 3*x`, the first assignment in `q`, corresponds to the transformation $\phi(\mathcal{L}, u, 3u)$. When $\mathcal{L} = \mathcal{L}_0$, this returns $\mathcal{L}_0' := \mathrm{ggen}\big((L, \varnothing, P_0')\big)$ where $P_0' = \{(3,0,0)^{\mathrm{T}}\}$. The next assignment in `q` is `y := x + y`. The corresponding affine transformation is $\phi(\mathcal{L}, v, v{+}u)$ and, letting $\mathcal{L} = \mathcal{L}_0'$, we obtain the grid $\mathcal{L}_0'' := \mathrm{ggen}\big((L, \varnothing, P_0'')\big)$ where $P_0'' = \{(3,3,0)^{\mathrm{T}}\}$. Figure 4 contains an abstract version $\mathtt{q}^\sharp$ of `q` where the the argument to the procedure is replaced by a grid and the assignment statements are replaced by the corresponding affine transformations. Thus we can now compute the grids $\mathcal{L}_i = \mathrm{ggen}\big((L, \varnothing, P_i)\big)$ for any i. In particular we can compute P_0, P_1 and P_2, as given in Example 1 and also $P_3 := \{(3375, 4224, 0)^{\mathrm{T}}\}$; hence $\mathcal{M} = \mathcal{L}_0 \oplus \mathcal{L}_1 \oplus \mathcal{L}_2 \oplus \mathcal{L}_3$ represents a fixpoint for the procedure; thus it includes all possible values for the vector $(\mathtt{x}, \mathtt{y})^{\mathrm{T}}$ that might be obtained as a result

```
q♯(L)
  if ... then
    L := φ(L, u, 3u)
    L := φ(L, v, u + v)
    q♯(L)
    L := φ(L, u, 5u)
    L := φ(L, v, u + v)
  endif
```

Fig. 4. An abstraction of `q`

of calling q. If, as in [20], the procedure is called with $\mathtt{x} = 2$ and $\mathtt{y} = 0$, then all the possible values for the vector $(\mathtt{x}, \mathtt{y})^{\mathrm{T}}$ are represented by the grid

$$\mathrm{gcon}\bigl(\{\mathtt{x} \equiv_{28} 2, \mathtt{y} \equiv_{12} 0\}\bigr) = \phi\bigl(\phi(\mathcal{M}, \mathtt{x}, 2\mathtt{x}), \mathtt{y}, 2\mathtt{y}\bigr).$$

We also consider a generalized affine image operator. This determines a set of congruence relations that hold between the given grid and its image. Clearly, since the relations are congruences, the image is also a grid. Note though, that in this case a hyperplane will be replaced by a, possibly infinite, set of hyperplanes.

The *generalized affine image* (resp., *generalized affine preimage*) is an extension of the affine image (resp., affine preimage) operator defined above. Given a grid $\mathcal{L} \in \mathbb{G}_n$, linear expressions $e' = \langle \boldsymbol{c}, \boldsymbol{x} \rangle + d$ and $e = \langle \boldsymbol{a}, \boldsymbol{x} \rangle + b$ with coefficients in $\mathbb{Q}$ and $f \in \mathbb{Q}$, the generalized affine image operator $\psi = \psi(\mathcal{L}, e', e, f)$ is defined as

$$\forall \boldsymbol{v}, \boldsymbol{w} \in \mathbb{R}^n : (\boldsymbol{v}, \boldsymbol{w}) \in \psi \iff \bigl(\langle \boldsymbol{c}, \boldsymbol{w} \rangle + d \equiv_f \langle \boldsymbol{a}, \boldsymbol{v} \rangle + b\bigr) \wedge \bigwedge_{\substack{0 \le i < n \\ c_i = 0}} w_i = v_i.$$

Note that, when $e' = x_k$ and $f = 0$, then the transformation is equivalent to the standard affine transformation on $\mathcal{L}$ with respect to the variable x_k and the affine expression e; that is

$$\psi(\mathcal{L}, x_k, e, 0) = \phi(\mathcal{L}, x_k, e).$$

However, when $e' = x_k$ and $f = 1$, then the transformation maps the point $\boldsymbol{x}$ to the set of points $\bigl\{\, (x_1, \ldots, x_{k-1}, x'_k, x_{k+1}, \ldots, x_n) \bigm| x'_k \equiv_1 \langle \boldsymbol{a}, \boldsymbol{x} \rangle + b \,\bigr\}$.

The following example illustrates how the generalized affine image, as defined here for grids, can be used to model the effect of a procedure containing a while loop.

Example 3. Consider the program procedure in Figure 5. The effect of procedure p invoked with values (a, b) will be to bind u to $3a + 2b + 1 + 3i$ for some $i \in \mathbb{Z}$ and leave $\mathtt{v} = b$ unchanged. Thus, if all possible initial values for the parameters u and v are given by the points of the grid $\mathcal{L}$, then, after executing p, the values of u and v will be given by the points of the grid $\psi(\mathcal{L}, x, 3x + 2y + 1, 3)$. Therefore, if $\mathcal{L} = \mathrm{ggen}\bigl((\varnothing, \varnothing, P)\bigr)$ where $P = \left(\begin{smallmatrix} 0 & 0 & 3 \\ 0 & 3 & 0 \end{smallmatrix}\right)$,

```
p(var u, var v)
  u := 3u + v + 1
  while ...
    u := u + 3
  endwhile
```

Fig. 5. A simple procedure

$$\psi(\mathcal{L}, x, 3x + 2y + 1, 3) = \mathrm{ggen}\Bigl(\bigl(\varnothing, \varnothing, \left(\begin{smallmatrix} 1 & 4 & 1 \\ 0 & 0 & 3 \end{smallmatrix}\right)\bigr)\Bigr).$$

5 Rectilinear Grids and Covering Boxes.

We show how we can reuse the standard interval domain [7] to represent a *rectilinear grid*, which is a grid that can be defined by a set of non-relational congruences (i.e., congruences where at most one coefficient is non-zero). We also show how such grids can provide safe approximations for any rational grid.

A non-empty n-dimensional *rational box* $\mathcal{B}$ is a sequence $(I_1, \ldots, I_n)$ of rational intervals. A *rational interval* is a pair $(\mu_i, \nu_i) \in \bigl(\mathbb{Q} \cup \{\infty\}\bigr)^2$ of *bounds*, called *lower* and *upper*, respectively. The lower bound is always less than or equal to the upper bound, where $q < \infty$ for each $q \in \mathbb{Q}$. If both the bounds are in $\mathbb{Q}$, the interval is said to be *bounded.* The essential idea we use here exploits the fact that the shape and size of an n-dimensional box $\mathcal{B}$ defines a rectangular "tile" that may be used to cover an n-dimensional vector space. It therefore follows that the box $\mathcal{B}$ determines a *covering* (i.e., tiling) of the n-dimensional vector space, where the given box $\mathcal{B}$ provides the position for one of the tiles. By defining a grid to be the vertices of the tiles in such a tiling, we obtain a rational rectilinear grid $\mathcal{L}$ and call $\mathcal{B}$ a *covering box for* $\mathcal{L}$.

Let $\mathcal{B} = (I_1, \ldots, I_n)$ be a non-empty box. For each $i = 1, \ldots, n$, let $I_i = (\mu_i, \nu_i)$; then, if $\mu_i \neq \nu_i$, let

$$\beta_i := \begin{cases} \bigl(\langle \boldsymbol{e}_i, \boldsymbol{x} \rangle \equiv_{\nu_i - \mu_i} \mu_i\bigr), & \text{if } I_i \text{ is bounded;} \\ \bigl(\langle \boldsymbol{e}_i, \boldsymbol{x} \rangle = \mu_i\bigr), & \text{if } I_i \text{ is not bounded.} \end{cases}$$

Then we say that the box $\mathcal{B}$ *represents* the grid $\mathcal{L} := \mathrm{gcon}(\mathcal{C})$, where $\mathcal{C} := \{\, \beta_i \bigm| 1 \leq i \leq n, \mu_i \neq \nu_i \,\}$. Note that the congruence system $\mathcal{C}$ is in minimal form. Observe also that, when $\mu_i = \nu_i$ for some $1 \leq i \leq n$, there is no corresponding congruence in $\mathcal{C}$ for (μ_i, ν_i); this is because, in this case, the tiling will cover every value in this dimension.

Consider now any non-empty rational grid $\mathcal{L}$. A *covering box* for $\mathcal{L}$ is a rational box representing the smallest rectilinear grid that contains $\mathcal{L}$. We now provide a procedure for computing the covering box of a grid.

Proposition 1. *Let $\mathcal{L} = \mathrm{ggen}(\mathcal{G})$ where $\mathcal{G} = \bigl(L, Q, \{\boldsymbol{p}\}\bigr)$. Let $q = \# Q$ and $Q = \{\boldsymbol{q}_1, \ldots, \boldsymbol{q}_q\}$. Let $\mathcal{B} = (I_1, \ldots, I_n)$ such that, for each $i \in \{1, \ldots, n\}$:*

1. *if, for some $\boldsymbol{\ell} \in L$, $\ell_i \neq 0$, then let $I_i := (0, 0)$;*
2. *if, for all $\boldsymbol{\ell} \in L$, $\ell_i = 0$, and $q_{1i} = \cdots = q_{qi} = 0$, let $I_i := (p_i, \infty)$;*
3. *otherwise, let $I_i := \bigl(p_i, p_i + |g|\bigr)$ where $g = \gcd\bigl(\{q_{1i}, \ldots, q_{qi}\}\bigr)$.*

Then $\mathcal{B}$ is a covering box for $\mathcal{L}$.

The complexity of computing the covering box for a grid in $\mathbb{G}_n$ using the procedure given in Proposition 1 is $\mathrm{O}\bigl(n^2\bigr)$.

Example 4. Consider grid $\mathcal{L}_1 = \mathrm{gcon}\bigl(\{x \equiv_3 0, y \equiv_2 1\}\bigr)$; then $\mathcal{L}_1$ is rectilinear and box $\mathcal{B}_1 = \bigl\{(0,3), (1,3)\bigr\}$ is a covering box representing $\mathcal{L}_1$. Consider the grid $\mathcal{L}_2 = \mathrm{gcon}\bigl(\{x \equiv_1 0, x + y \equiv_3 2\}\bigr)$; then $\mathcal{L}_2$ is not rectilinear and box $\mathcal{B}_2 = \bigl\{(1,2), (1,2)\bigr\}$ is a covering box for $\mathcal{L}_2$. Thus $\mathcal{B}_2$ represents the grid $\mathcal{L}'_2 = \mathrm{gcon}\bigl(\{x \equiv_1 1, y \equiv_1 1\}\bigr)$.

It should be stressed that a grid $\mathcal{L} \in \mathbb{G}_n$ does not have, in general, a unique covering box. For instance, if $\mathcal{B} = (I_1, \ldots, I_i, \ldots, I_n)$ is a covering box for $\mathcal{L}$, $I_i = (\mu_i, \nu_i)$ is bounded and $\mu_i \neq \nu_i$, then the box $\mathcal{B}' = (I_1, \ldots, I'_i, \ldots, I_n)$ is also a covering box for $\mathcal{L}$ if, for some $m \in \mathbb{Z}$, $I'_i = (\mu_i + m(\nu_i - \mu_i), \nu_i + m(\nu_i - \mu_i))$.

6 Grid Widening

A simple and general characterization of a widening for enforcing and accelerating convergence of an upward iteration sequence is given in [7–10]. We assume here a minor variation of this classical definition (see footnote 6 in [10, p. 275]).

Definition 4. (Widening.) *Let $\langle D, \vdash, \mathbf{0}, \oplus \rangle$ be a join-semilattice. The partial operator $\nabla \colon D \times D \rightarrowtail D$ is a* widening *if*

1. *for each $d_1, d_2 \in D$, $d_1 \vdash d_2$ implies that $d_1 \nabla d_2$ is defined and $d_2 \vdash d_1 \nabla d_2$;*
2. *for each increasing chain $d_0 \vdash d_1 \vdash \cdots$, the increasing chain defined by $d'_0 := d_0$ and $d'_{i+1} := d'_i \nabla (d'_i \oplus d_{i+1})$, for $i \in \mathbb{N}$, is not strictly increasing.*

In addition to the formal requirements in Definition 4, it is also important to have a widening that has an efficient implementation, preferably, one that depends on a simple syntactic mapping of the representations. At the same time, so that the widening is well-defined, the result of this operation should be independent of the actual representation used. For this reason, the widening we propose requires the concept of *strong minimal form* for the congruence systems.

Definition 5. *A congruence system $\mathcal{C}$ in minimal form is in* strong minimal form *if, for each pair of distinct proper congruences, $\langle \boldsymbol{a}, \boldsymbol{x} \rangle \equiv_1 b$ and $\langle \boldsymbol{c}, \boldsymbol{x} \rangle \equiv_1 d$ in $\mathcal{C}$, if $\operatorname{piv}_<(\boldsymbol{c}) = k > 0$, then $-c_k < 2a_k \leq c_k$.*

A congruence system in minimal form can always be reduced to a congruence system in strong minimal form without affecting the values of the pivot elements.

Definition 6. *The congruence systems $\mathcal{C}_1$ and $\mathcal{C}_2$ in minimal form are said to be* pivot equivalent *if, for each $i, j \in \{1, 2\}$ where $i \neq j$, for each $\beta \in \mathcal{C}_i$, there exists $\gamma \in \mathcal{C}_j$ such that $\beta \Uparrow \gamma$.*

Proposition 2. *There exists an algorithm with complexity $\mathrm{O}\bigl(n^3\bigr)$ for converting a congruence system $\mathcal{C}$ in $\mathbb{Q}^n$ in minimal form to a congruence system $\mathcal{C}'$ in strong minimal form such that $\mathcal{C}$ is pivot equivalent to $\mathcal{C}'$ and $\operatorname{gcon}(\mathcal{C}) = \operatorname{gcon}(\mathcal{C}')$.*

We also have that, relative to the affine hull of a grid, the strong minimal form of any congruence system that describes that grid is canonical.

Proposition 3. *Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be congruence systems in strong minimal form in $\mathbb{Q}^n$ such that $\mathcal{L} = \operatorname{gcon}(\mathcal{C}_1) = \operatorname{gcon}(\mathcal{C}_2)$. Then, for all $\beta \in \mathcal{C}_1$, there exists $\gamma \in \mathcal{C}_2$ such that $\operatorname{gcon}\bigl(\{\beta\}\bigr) \cap \operatorname{affine.hull}(\mathcal{L}) = \operatorname{gcon}\bigl(\{\gamma\}\bigr) \cap \operatorname{affine.hull}(\mathcal{L})$.*

We now define a widening for grids that uses the congruence systems.

Definition 7. *Let $\mathcal{L}_1 = \operatorname{gcon}(\mathcal{C}_1)$ and $\mathcal{L}_2 = \operatorname{gcon}(\mathcal{C}_2)$ be two grids in $\mathbb{G}_n$ such that $\mathcal{L}_1 \subseteq \mathcal{L}_2$, $\mathcal{C}_1$ is in minimal form and $\mathcal{C}_2$ is in strong minimal form. Then the* grid widening *$\mathcal{L}_1 \nabla_{\mathrm{c}} \mathcal{L}_2$ is defined by*

$$\mathcal{L}_1 \nabla_{\mathrm{c}} \mathcal{L}_2 := \begin{cases} \mathcal{L}_2, & \text{if } \mathcal{L}_1 = \varnothing \text{ or } \dim(\mathcal{L}_1) < \dim(\mathcal{L}_2), \\ \operatorname{gcon}(\mathcal{C}_S), & \text{otherwise,} \end{cases}$$

where $\mathcal{C}_S := \{\, \gamma \in \mathcal{C}_2 \mid \exists \beta \in \mathcal{C}_1 \mathrel{.} \beta \Uparrow \gamma \,\}$.

Proposition 4. *The operator '∇_c' is a widening on $\mathbb{G}_n$.*

This widening can be implemented to have complexity $\mathrm{O}\bigl(n^2\bigr)$. By Proposition 3, for a fixed variable ordering, the operator '∇_c' is well-defined. The next example illustrates that '∇_c' depends on the variable ordering.

Example 5. Consider the grids $\mathcal{L}_1 := \mathrm{gcon}(\mathcal{C}_1)$ and $\mathcal{L}_2 := \mathrm{gcon}(\mathcal{C}_2)$ in $\mathbb{G}_2$, where

$$\mathcal{C}_1 := \{5x + y \equiv_1 0,\ 22x \equiv_1 0\}, \qquad \mathcal{C}_2 := \{5x + y \equiv_1 0,\ 44x \equiv_1 0\}.$$

Assume that variables are ordered so that x precedes y, as in the vector $(x, y)^{\mathrm{T}}$; then, the congruence systems $\mathcal{C}_1$ and $\mathcal{C}_2$ are in strong minimal form and, according to Definition 7, we obtain $\mathcal{L}_1 \nabla_c \mathcal{L}_2 = \mathrm{gcon}\bigl(\{5x + y \equiv_1 0\}\bigr)$. On the other hand, the same grids as before are also described by the congruence systems

$$\mathcal{C}_1' := \{9y + x \equiv_1 0,\ 22y \equiv_1 0\}, \qquad \mathcal{C}_2' := \{9y + x \equiv_1 0,\ 44y \equiv_1 0\},$$

respectively, which are in strong minimal form when taking the variable order where y precedes x. In this case, by Definition 7, $\mathcal{L}_1 \nabla_c \mathcal{L}_2 = \mathrm{gcon}\bigl(\{9y + x \equiv_1 0\}\bigr)$.

In Definition 7, it is required that $\mathcal{C}_2$ is in strong minimal form. The following example shows that this is necessary for the operator '∇_c' to be well-defined.

Example 6. Let $\mathcal{L}_1 := \mathrm{gcon}(\mathcal{C}_1)$, $\mathcal{L}_2 := \mathrm{gcon}(\mathcal{C}_2)$ and $\mathcal{L}_2' := \mathrm{gcon}(\mathcal{C}_2')$ where $\mathcal{C}_1 = \{x \equiv_2 0,\ y \equiv_2 0\}$, $\mathcal{C}_2 = \{x \equiv_1 0,\ x + y \equiv_2 0\}$, $\mathcal{C}_2' = \{x \equiv_1 0,\ 3x + y \equiv_2 0\}$; then $\mathcal{L}_2 = \mathcal{L}_2'$. Note that only $\mathcal{C}_1$ and $\mathcal{C}_2$ are in strong minimal form. Therefore, assuming $\mathcal{C}_S$ (resp., $\mathcal{C}_S{}'$) is defined as in Definition 7 using $\mathcal{C}_1$ and $\mathcal{C}_2$ (resp., $\mathcal{C}_1$ and $\mathcal{C}_2'$), we have $\mathcal{C}_S = \{x + y \equiv_2 0\}$ and $\mathcal{C}_S{}' = \{3x + y \equiv_2 0\}$. Thus $\mathcal{L}_1 \nabla_c \mathcal{L}_2 = \mathrm{gcon}(\mathcal{C}_S) \neq \mathrm{gcon}(\mathcal{C}_S{}')$.

Widening using Generator Systems. We now describe a widening for the domain of grids using the generator systems which will be seen as similar to that already described for congruence systems. Thus we need the concepts of *strong minimal form* and *pivot equivalence* for generator systems.

Definition 8. *A generator system $\mathcal{G} = \bigl((L, Q, P)\bigr)$ in $\mathbb{Q}^n$ is in* strong minimal form *if $\mathcal{G}$ is in minimal form and, for each pair of distinct parameters $\boldsymbol{u}, \boldsymbol{v} \in Q$, if* $\mathrm{piv}_>(\boldsymbol{v}) = k \leq n$, *then* $-v_k < 2u_k \leq v_k$.

Definition 9. *Generator systems $\mathcal{G}_1 = \bigl(L_1, Q_1, \{\boldsymbol{p}_1\}\bigr)$ and $\mathcal{G}_2 = \bigl(L_2, Q_2, \{\boldsymbol{p}_2\}\bigr)$ in minimal form are said to be* pivot equivalent *if, for each $i, j \in \{1, 2\}$ where $i \neq j$: for each $\boldsymbol{q}_i \in Q_i$, there exists $\boldsymbol{q}_j \in Q_j$ such that $\boldsymbol{q}_i \Downarrow \boldsymbol{q}_j$; and, for each $\boldsymbol{\ell}_i \in L_i$, there exists $\boldsymbol{\ell}_j \in L_j$ such that* $\mathrm{piv}_>(\boldsymbol{\ell}_i) = \mathrm{piv}_>(\boldsymbol{\ell}_j)$.

Proposition 5. *There exists an algorithm with complexity $\mathrm{O}\bigl(n^3\bigr)$ for converting a generator system $\mathcal{G}$ in minimal form to a generator system $\mathcal{G}'$ in strong minimal form such that $\mathcal{G}$ is pivot equivalent to $\mathcal{G}'$ and* $\mathrm{ggen}(\mathcal{G}) = \mathrm{ggen}(\mathcal{G}')$.

We now define a widening for grids using the generator systems.

Definition 10. *Let $\mathcal{L}_1 = \mathrm{ggen}(\mathcal{G}_1)$ and $\mathcal{L}_2 = \mathrm{ggen}(\mathcal{G}_2)$ be two grids in $\mathbb{G}_n$ such that $\mathcal{L}_1 \subseteq \mathcal{L}_2$, $\mathcal{G}_1 = (L_1, Q_1, P_1)$ is in minimal form and $\mathcal{G}_2 = (L_2, Q_2, P_2)$ is in strong minimal form. Then the* grid widening $\mathcal{L}_1 \nabla_{\mathcal{G}} \mathcal{L}_2$ *is defined by*

$$\mathcal{L}_1 \nabla_{\mathcal{G}} \mathcal{L}_2 := \begin{cases} \mathcal{L}_2, & \text{if } \mathcal{L}_1 = \varnothing \text{ or } \dim(\mathcal{L}_1) < \dim(\mathcal{L}_2); \\ \mathrm{ggen}(\mathcal{G}_S), & \text{otherwise,} \end{cases}$$

where $\mathcal{G}_S := \bigl(L_2 \cup (Q_2 \setminus Q_S), Q_S, P_2\bigr)$ and $Q_S := \{\, \boldsymbol{v} \in Q_2 \mid \exists \boldsymbol{u} \in Q_1 \,.\, \boldsymbol{u} \Downarrow \boldsymbol{v} \,\}$.

Proposition 6. *The operator '$\nabla_{\mathcal{G}}$' is a widening on $\mathbb{G}_n$.*

This widening can be implemented to have complexity $\mathrm{O}\bigl(n^2\bigr)$. If we include the reduction to (strong) minimal form of the arguments, the complexity of both widenings is $\mathrm{O}\bigl(n^3\bigr)$. The following shows that the widenings are not equivalent.

Example 7. Consider grids $\mathcal{L}_1 = \mathrm{gcon}(\mathcal{C}_1) = \mathrm{ggen}(\mathcal{G}_1)$ and $\mathcal{L}_2 = \mathrm{gcon}(\mathcal{C}_2) = \mathrm{ggen}(\mathcal{G}_2)$ where

$$\mathcal{C}_1 = \{A \equiv_2 0,\ B \equiv_2 0\}, \qquad \mathcal{G}_1 = \Bigl(\varnothing,\ \bigl\{(2,0)^{\mathrm{T}}, (0,2)^{\mathrm{T}}\bigr\},\ \bigl\{(0,0)^{\mathrm{T}}\bigr\}\Bigr),$$

$$\mathcal{C}_2 = \{A \equiv_1 0,\ A + B \equiv_2 0\}, \qquad \mathcal{G}_2 = \Bigl(\varnothing,\ \bigl\{(1,1)^{\mathrm{T}}, (0,2)^{\mathrm{T}}\bigr\},\ \bigl\{(0,0)^{\mathrm{T}}\bigr\}\Bigr).$$

Then $\mathcal{L}_1 \nabla_{\mathcal{C}} \mathcal{L}_2 = \mathrm{gcon}\bigl(\{A + B \equiv_2 0\}\bigr) \neq \mathcal{L}_1 \nabla_{\mathcal{G}} \mathcal{L}_2 = \mathrm{gcon}\bigl(\{A - B \equiv_2 0\}\bigr)$.

Enhancements. The widenings '$\nabla_{\mathcal{C}}$' and '$\nabla_{\mathcal{G}}$' can be used for developing more refined widenings (or extrapolations) by following techniques such as the frameworks proposed in [3, 4] for convex polyhedra. The precision of a grid widening can also be improved by exploiting the covering boxes, defined in Section 5. That is, we can use boxes to provide a *covered* extrapolation operator that improves the approximation of the widening operator by ensuring that the result cannot be worse than the covering box for the larger of the two grids being widened.

One way to show that an extrapolation operator is, in fact, a widening is to provide the operator with a *finite convergence certificate* [4]. In particular, for the grid domain and widenings '$\nabla_{\mathcal{C}}$' and '$\nabla_{\mathcal{G}}$', such a certificate is defined to be a triple $(\mathcal{O}, \succ, \mu)$ where $(\mathcal{O}, \succ)$ is a well-founded ordered set and $\mu \colon \mathbb{G}_n \to \mathcal{O}$ is such that, for all $\mathcal{L}_1 \subset \mathcal{L}_2 \in \mathbb{G}_n$, $\mu(\mathcal{L}_1) \succ \mu(\mathcal{L}_3)$ where $\mathcal{L}_3 = \mathcal{L}_1 \nabla_{\mathcal{C}} \mathcal{L}_2 = \mathcal{L}_1 \nabla_{\mathcal{G}} \mathcal{L}_2$. Thus, a finite convergence certificate for both the grid widenings can be defined by taking $\mathcal{O}$ equal to $\{0, \dots, n\} \times \{0, \dots, n\}$, $\succ$ the lexicographic ordering on $\mathcal{O}$ and, for all $\mathcal{L} \in \mathbb{G}_n$, letting $\mu(\mathcal{L}) := (\#\,\mathcal{E}, \#\,\mathcal{C})$ where $\mathcal{L} = \mathrm{gcon}(\mathcal{C})$, $\mathcal{C}$ is in minimal form, and $\mathcal{E} \subseteq \mathcal{C}$ is the set of equalities in $\mathcal{C}$. By Definitions 7 and 10 and [2, Propositions 2 and 4], it follows that $\mathcal{L}_1 \neq \mathcal{L}_2$ implies $\mu(\mathcal{L}_1) \succ \mu(\mathcal{L}_3)$; hence we have the same finite convergence certificate for both the grid widenings. Observe that this implies that any iteration using a mixed sequence of congruence and generator grid widenings will converge after a finite number of steps.

It is shown in [4] that a widening for a powerset domain can be obtained from any widening on its base-level domain that has a finite convergence certificate. Thus, with the above certificate for the grid widenings, we can instantiate the generic widenings for powersets to one for powersets of grids, using any combination of the grid widenings '$\nabla_{\mathcal{C}}$' and '$\nabla_{\mathcal{G}}$'.

Discussion. It might be asked why we do not define the congruence widening for grids as the natural counterpart to the standard widening for convex polyhedra as specified in the PhD thesis of N. Halbwachs [15], also described in [16]. To see why we have not taken this approach, consider the following grid extrapolation operator 'h'. Let $\mathcal{L}_1 = \mathrm{gcon}(\mathcal{C}_1)$ and $\mathcal{L}_2 = \mathrm{gcon}(\mathcal{C}_2) \in \mathbb{G}_n \setminus \varnothing$ where $\mathcal{L}_1 \subseteq \mathcal{L}_2$ and $\mathcal{C}_1$ and $\mathcal{C}_2$ are congruence systems in $\mathbb{R}^n$ in strong minimal form. Then $h(\mathcal{L}_1, \mathcal{L}_2) := \mathrm{gcon}(\mathcal{C}'_1 \cup \mathcal{C}'_2)$ where

$$\mathcal{C}'_1 := \left\{ \beta \in \mathcal{C}_1 \;\middle|\; \mathcal{L}_2 \subseteq \mathrm{gcon}\bigl(\{\beta\}\bigr) \right\}, \tag{1}$$

$$\mathcal{C}'_2 := \left\{ \gamma \in \mathcal{C}_2 \;\middle|\; \exists \beta \in \mathcal{C}_1 \,.\, \mathcal{L}_1 = \mathrm{gcon}\bigl(\mathcal{C}_1[\gamma/\beta]\bigr) \right\}. \tag{2}$$

Note that, as shown in Example 6, to ensure the operator 'h' is well-defined, *both* the congruence systems $\mathcal{C}_1$ and $\mathcal{C}_2$ need to be in strong minimal form. Note also that, as there is no finite set of proper congruences semantically equivalent to a single equality, the above definition has ignored any distinction between equalities and proper congruences. The next example illustrates that ignoring the $\mathcal{C}'_2$ component (as can be done in implementations of the standard widening for convex polyhedra when the affine hull is the universe) can lose precision.

Example 8. Consider again the grids and congruence systems in Example 6. Then, assuming definition (1) for $\mathcal{C}'_1$ and (2) for $\mathcal{C}'_2$ and $\mathcal{C}'_3$, we have $\mathcal{C}'_1 = \varnothing$, $\mathcal{C}'_2 = \{x + y \equiv_2 0\}$ and $\mathcal{C}'_3 = \{3x + y \equiv_2 0\}$ so that $h(\mathcal{L}_1, \mathcal{L}_2) = \mathrm{gcon}\bigl(\mathcal{C}'_1 \cup \mathcal{C}'_2\bigr) = \mathrm{gcon}\bigl(\{x + y \equiv_2 0\}\bigr)$ although $h(\mathcal{L}_1, \mathcal{L}_3) = \mathrm{gcon}\bigl(\mathcal{C}'_1 \cup \mathcal{C}'_3\bigr) = \mathrm{gcon}\bigl(\{3x + y \equiv_2 0\}\bigr)$.

7 Conclusion

In this paper, we have taken the domain of rational grids (see [2]) and added further operators to make it into a truly practical domain suitable for static program analysis. For instance, it is well-known that affine transformations are needed for representing the effects of assignment statements; so here, we have specified several affine transformations operators for grids including single update affine image and preimage operators and generalized forms of these that maintain the necessary closure property for grids. We have shown, by means of examples, how the affine image functions may be used to safely abstract (possibly recursive) procedures and functions containing assignment statements and hence be used for interprocedural analysis.

We have shown that the polyhedral domain of intervals may be used to represent a rectangular grid and provided algorithms for finding the best approximation of an arbitrary grid as a rectangular grid. We observe that this idea of using a polyhedral domain to represent the "holes" in a grid could easily be generalized from that of representing a rectangular grid using covering boxes to the representation of any grid using n-dimensional parallelograms.

As rational grids do not satisfy the ascending chain condition, a widening is required. We have proposed two widenings for rational grids; one using congruence systems and the other, generator systems, each coming with a straightforward implementation. To ensure these widenings are well-defined, the systems representing the grid have to be in strong minimal forms; that is, in minimal form where the absolute value of the non-diagonal elements of the matrix of coefficients of the proper congruences or parameters is minimal. We compare the grid widening that uses congruence systems with the grid counterpart of the standard widening for convex polyhedra [16].

All the operators, other than the generator widening, described here are implemented in the PPL [5, 6] together with other operations, such as one for concatenating grids and ones that add, remove, rename and map the space dimensions or expand and fold them along the lines of [12]; however space considerations prevent their discussion here. The interested reader is referred to the web site of the PPL, where all the code and documentation is publicly available.

References

1. C. Ancourt. *Génération Automatique de Codes de Transfert pour Multiprocesseurs à Mémoires Locales*. PhD thesis, Université de Paris VI, March 1991.
2. R. Bagnara, K. Dobson, P. M. Hill, M. Mundell, and E. Zaffanella. A linear domain for analyzing the distribution of numerical values. Report 2005.06, School of Computing, University of Leeds, UK, 2005. Available at `http://www.comp.leeds.ac.uk/research/pubs/reports.shtml`.
3. R. Bagnara, P. M. Hill, E. Ricci, and E. Zaffanella. Precise widening operators for convex polyhedra. In R. Cousot, editor, *Static Analysis: Proceedings of the 10th International Symposium*, volume 2694 of *Lecture Notes in Computer Science*, pages 337–354, San Diego, California, USA, 2003. Springer-Verlag, Berlin.
4. R. Bagnara, P. M. Hill, and E. Zaffanella. Widening operators for powerset domains. In B. Steffen and G. Levi, editors, *Verification, Model Checking and Abstract Interpretation: Proceedings of the 5th International Conference (VMCAI 2004)*, volume 2937 of *Lecture Notes in Computer Science*, pages 135–148, Venice, Italy, 2003. Springer-Verlag, Berlin.
5. R. Bagnara, P. M. Hill, and E. Zaffanella. *The Parma Polyhedra Library User's Manual*. Department of Mathematics, University of Parma, Parma, Italy, release 0.9 edition, To appear in March 2006. Preliminary version available at `http://www.cs.unipr.it/ppl/`.
6. R. Bagnara, E. Ricci, E. Zaffanella, and P. M. Hill. Possibly not closed convex polyhedra and the Parma Polyhedra Library. In M. V. Hermenegildo and G. Puebla, editors, *Static Analysis: Proceedings of the 9th International Symposium*, volume 2477 of *Lecture Notes in Computer Science*, pages 213–229, Madrid, Spain, 2002. Springer-Verlag, Berlin.
7. P. Cousot and R. Cousot. Static determination of dynamic properties of programs. In B. Robinet, editor, *Proceedings of the Second International Symposium on Programming*, pages 106–130, Paris, France, 1976. Dunod, Paris, France.
8. P. Cousot and R. Cousot. Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints. In *Proceedings of the Fourth Annual ACM Symposium on Principles of Programming Languages*, pages 238–252, New York, 1977. ACM Press.
9. P. Cousot and R. Cousot. Abstract interpretation frameworks. *Journal of Logic and Computation*, 2(4):511–547, 1992.

10. P. Cousot and R. Cousot. Comparing the Galois connection and widening/narrowing approaches to abstract interpretation. In M. Bruynooghe and M. Wirsing, editors, *Proceedings of the 4th International Symposium on Programming Language Implementation and Logic Programming*, volume 631 of *Lecture Notes in Computer Science*, pages 269–295, Leuven, Belgium, 1992. Springer-Verlag, Berlin.
11. P. Cousot and N. Halbwachs. Automatic discovery of linear restraints among variables of a program. In *Conference Record of the Fifth Annual ACM Symposium on Principles of Programming Languages*, pages 84–96, Tucson, Arizona, 1978. ACM Press.
12. D. Gopan, F. DiMaio, N. Dor, T. Reps, and M. Sagiv. Numeric domains with summarized dimensions. In K. Jensen and A. Podelski, editors, *Tools and Algorithms for the Construction and Analysis of Systems, 10th International Conference, TACAS 2004*, volume 2988 of *Lecture Notes in Computer Science*, pages 512–529. Springer-Verlag, Berlin, 2004.
13. P. Granger. Static analysis of arithmetical congruences. *International Journal of Computer Mathematics*, 30:165–190, 1989.
14. P. Granger. Static analyses of congruence properties on rational numbers (extended abstract). In P. Van Hentenryck, editor, *Static Analysis: Proceedings of the 4th International Symposium*, volume 1302 of *Lecture Notes in Computer Science*, pages 278–292, Paris, France, 1997. Springer-Verlag, Berlin.
15. N. Halbwachs. *Détermination Automatique de Relations Linéaires Vérifiées par les Variables d'un Programme*. Thèse de 3$^{\text{ème}}$ cycle d'informatique, Université scientifique et médicale de Grenoble, Grenoble, France, March 1979.
16. N. Halbwachs, Y.-E. Proy, and P. Roumanoff. Verification of real-time systems using linear relation analysis. *Formal Methods in System Design*, 11(2):157–185, 1997.
17. V. Loechner. *PolyLib*: A library for manipulating parameterized polyhedra. Available at `http://icps.u-strasbg.fr/~loechner/polylib/`, March 1999. Declares itself to be a continuation of [21].
18. A. Miné. A few graph-based relational numerical abstract domains. In M. V. Hermenegildo and G. Puebla, editors, *Static Analysis: Proceedings of the 9th International Symposium*, volume 2477 of *Lecture Notes in Computer Science*, pages 117–132, Madrid, Spain, 2002. Springer-Verlag, Berlin.
19. M. Müller-Olm and H. Seidl. Precise interprocedural analysis through linear algebra. In N. D. Jones and X. Leroy, editors, *Proceedings of the 31st ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL 2004)*, pages 330–341, Venice, Italy, 2004. ACM.
20. M. Müller-Olm and H. Seidl. A generic framework for interprocedural analysis of numerical properties. In C. Hankin and I. Siveroni, editors, *Static Analysis: Proceedings of the 12th International Symposium*, volume 3672 of *Lecture Notes in Computer Science*, pages 235–250, London, UK, 2005. Springer-Verlag, Berlin.
21. D. K. Wilde. A library for doing polyhedral operations. Master's thesis, Oregon State University, Corvallis, Oregon, December 1993. Also published as IRISA *Publication interne* 785, Rennes, France, 1993.

Appendix

Here we prove the results stated in the main part of the paper. In the following two proofs, we require an extended version of the Euclidean algorithm. Suppose $v, v' \in \mathbb{Z}$. Let $r \in \mathbb{Z}$ be the greatest common divisor of v, v' and $s, s' \in \mathbb{Z}$ be relatively prime such that $sv + s'v' = r$. Then we write

$$\mathrm{gcd}(v, v') := r, \qquad \mathrm{gcdext}(v, v') := \bigl(r, (s, s')\bigr).$$

Proof (of Proposition 1). Let $\mathcal{C}$ be a congruence system for $\mathcal{L}$ and let $\mathcal{L}_R$ be the smallest rectilinear grid that contains $\mathcal{L}$. Suppose also that $\mathcal{B}$ defines the rectilinear grid $\mathcal{L}_B$. Then we need to show that $\mathcal{L}_B = \mathcal{L}_R$. Let $\mathcal{C}_B := \{\, \beta_i \bigm| 1 \leq i \leq n, \mu_i \neq \nu_i \,\}$ and $\mathcal{C}_R$ be a congruence system in minimal form for $\mathcal{L}_R$. Observe that $\mathcal{L}_R = \mathrm{ggen}(\mathcal{G}_R)$ where $\mathcal{G}_R$ is a minimal form for the generator system

$$\bigl(\{\ell_i \boldsymbol{e}_i | \boldsymbol{\ell} \in L, \ell_i \neq 0\}, \{q_i \boldsymbol{e}_i | \boldsymbol{q} \in Q, q_i \neq 0\}, \{\boldsymbol{p}\}\bigr).$$

To prove the result, we consider each dimension i where $1 \leq i \leq n$ separately and show that there exists $\gamma_i = \bigl(\langle \boldsymbol{e}_i, \boldsymbol{x} \rangle \equiv_f b\bigr) \in \mathcal{C}_R$ for some $f, b \in \mathbb{Q}$ if and only if β_i exists and $\mathrm{gcon}(\gamma_i) = \mathrm{gcon}(\beta_i)$. We consider each of the three cases (1), (2) and (3) separately.

Suppose, for some $\boldsymbol{\ell} \in L$, $\ell_i \neq 0$. Then case (1) applies and $I_i = (0, 0)$. Moreover, there is a line $\boldsymbol{e}_i \ell_i$ in $\mathcal{G}_R$. Thus, there are no congruences $\langle \boldsymbol{a}, \boldsymbol{x} \rangle \equiv_f b$ in $\mathcal{C}_B$ or $\mathcal{C}_R$ where $\mathrm{piv}_<(\boldsymbol{a}) = i$.

Suppose, for all $\boldsymbol{\ell} \in L$, $\ell_i = 0$, and $q_{1i} = \cdots = q_{qi} = 0$. Then case (2) applies and $I_i = (p_i, \infty)$. Therefore there exists $\beta_i = \bigl(\langle \boldsymbol{e}_i, \boldsymbol{x} \rangle = p_i\bigr) \in \mathcal{C}_B$. Moreover, there are no lines or parameters in $\mathcal{G}_R$ with a non-zero i-th coordinate. Thus there exists $\gamma_i \in \mathcal{C}_R$ where $f = 0$ and $b = p_i$ so that $\mathrm{gcon}(\gamma_i) = \mathrm{gcon}(\beta_i)$.

Suppose, for all $\boldsymbol{\ell} \in L$, $\ell_i = 0$, and, for some $j = 1, \ldots, q$, $q_{ji} \neq 0$. Then case (3) applies and $I_i = (p_i, p_i + |g|)$ where $g = \mathrm{gcd}\bigl(\{q_{1i}, \ldots, q_{qi}\}\bigr) \neq 0$. Therefore there exists $\beta_i = \bigl(\langle \boldsymbol{e}_i, \boldsymbol{x} \rangle \equiv_g p_i\bigr) \in \mathcal{C}_B$. Moreover, there are no lines in $\mathcal{G}_R$ with a non-zero i-th coordinate. Thus there exists $\gamma_i \in \mathcal{C}_R$ where $f = g$ and $b = p_i + mg$ for some $m \in \mathbb{Z}$. Thus $\mathrm{gcon}(\gamma_i) = \mathrm{gcon}(\beta_i)$. □

Proof (of Proposition 2). Suppose that $\mathcal{C}$ is not in strong minimal form. Then, by Definition 5, there exists a proper congruence $\beta = \bigl(\langle \boldsymbol{a}, \boldsymbol{x} \rangle \equiv_1 b\bigr) \in \mathcal{C}$, such that the following holds:

1. there exists $i > 0$ and a proper congruence $\gamma = \bigl(\langle \boldsymbol{c}, \boldsymbol{x} \rangle \equiv_1 d\bigr) \in \mathcal{C} \setminus \{\beta\}$ where $\mathrm{piv}_<(\gamma) = i$ and either $2a_i \leq -c_i$ or $2a_i > c_i$.

Suppose that $0 \leq k \leq n$ is the maximum value for the index i such that condition (1) holds.

We show, by induction on k, that there exists a sequence of at most n transformations, each of which having complexity $\mathrm{O}\bigl(n\bigr)$, from β to the congruence $\beta' = \bigl(\langle \boldsymbol{a}', \boldsymbol{x} \rangle \equiv_1 b'\bigr)$, such that, if $\mathcal{C}' := \bigl(\mathcal{C} \setminus \{\beta\}\bigr) \cup \{\beta'\}$, then $\mathrm{gcon}(\mathcal{C}') = \mathrm{gcon}(\mathcal{C})$ and condition (1) (when β is replaced by β') does not hold.

If $k = 0$, then condition (1) does not hold for β. Therefore let $\beta' = \beta$.

Suppose now that $k > 0$ so that condition (1) holds for $i = k$. As $\mathcal{C}$ is in minimal form, $k < \operatorname{piv}_<(\boldsymbol{a})$. Let

$$\boldsymbol{a}'' = \begin{cases} \boldsymbol{a} - \left\lceil \frac{a_k}{c_k} \right\rceil \boldsymbol{c} \quad \text{and} \quad b'' = b - \left\lceil \frac{a_k}{c_k} \right\rceil d, & \text{if } a_k \bmod c_k > \frac{a_k}{2}; \\ \boldsymbol{a} - \left\lfloor \frac{a_k}{c_k} \right\rfloor \boldsymbol{c} \quad \text{and} \quad b'' = b - \left\lfloor \frac{a_k}{c_k} \right\rfloor d, & \text{if } a_k \bmod c_k \leq \frac{a_k}{2}. \end{cases}$$

Then $-c_k < 2a''_k \leq c_k$. Also, for $k+1 \leq j \leq n$, we have $c_j = 0$ so that $a_j = a''_j$ and $\operatorname{piv}_<(\boldsymbol{a}'') = \operatorname{piv}_<(\boldsymbol{a})$. Letting $\beta'' := \bigl(\langle \boldsymbol{a}'', \boldsymbol{x} \rangle \equiv_1 b''\bigr)$ and $\mathcal{C}'' := \bigl(\mathcal{C} \setminus \{\beta\}\bigr) \cup \{\beta''\}$, we have $\operatorname{gcon}(\mathcal{C}'') = \operatorname{gcon}(\mathcal{C})$. Note that this transformation has a complexity $\mathrm{O}\bigl(n\bigr)$. As k'', the maximum index such that condition (1) holds for β'', is strictly less than k we can apply the inductive hypothesis to $\mathcal{C}''$ and β''. Thus there is a sequence of at most $n-1$ transformations from β'' to β' such that, $\operatorname{gcon}\Bigl(\bigl(\mathcal{C}'' \setminus \{\beta''\}\bigr) \cup \{\beta'\}\Bigr) = \operatorname{gcon}(\mathcal{C}'')$ and condition (1) (when β is replaced by β') does not hold. Thus there is a sequence of at most n transformations from β to β' such that $\operatorname{gcon}\Bigl(\bigl(\mathcal{C} \setminus \{\beta\}\bigr) \cup \{\beta'\}\Bigr) = \operatorname{gcon}(\mathcal{C})$. As each of the individual steps has complexity $\mathrm{O}\bigl(n\bigr)$, the sequence of transformations has complexity $\mathrm{O}\bigl(n^2\bigr)$.

We repeat this sequence of transformations for each proper congruence in $\mathcal{C}$ to obtain a congruence system $\mathcal{C}'$ such that, for each proper congruence $\beta' \in \mathcal{C}'$, condition (1) does not hold. Thus, by Definition 5, $\mathcal{C}'$ is in strong normal form. Thus, as there are at most n proper congruences in $\mathcal{C}$ since, by hypothesis, $\mathcal{C}$ is in minimal form, the complexity of computing the strong minimal form is $\mathrm{O}\bigl(n^3\bigr)$. $\square$

In the following lemma, which is needed for the proofs of Lemma 2, Propositions 6 and 8, we show how to convert a generator system to a congruence system and vice-versa. In the following proof, letting $H \in \mathbb{R}^{n \times n}$, we say that H is *upper triangular form* if, for all $i = 1, \ldots, n$, $H_{ii} \neq 0$ and, for all j where $1 \leq j < i \leq n$, $H_{ij} = 0$. Similarly, a matrix H is in *lower triangular form* if, for all $i = 1, \ldots, n$, $H_{ii} \neq 0$ and, for all j where $1 \leq i < j \leq n$, $H_{ij} = 0$. Thus the transpose of an upper triangular matrix is lower triangular and vice-versa.

Lemma 1. *There exists a computable, invertible function that converts a generator system $\mathcal{G} = \bigl(L, Q, \{\boldsymbol{p}\}\bigr)$ in $\mathbb{Q}^n$ in minimal form to a congruence system $\mathcal{C} = \mathcal{E} \cup \mathcal{F}$ in $\mathbb{Q}^n$ in minimal form where $\mathcal{E}$ are equalities and $\mathcal{F}$ are proper congruences and such that*

1. $\# Q = \# \mathcal{F} = n - \# L - \# \mathcal{E}$;
2. *there exists $\boldsymbol{q} \in Q$ if and only if there exists $\beta = \bigl(\langle \boldsymbol{a}, \boldsymbol{x} \rangle \equiv_1 0\bigr) \in \mathcal{F}$, such that, for some $k \in \{1, \ldots, n\}$, $\operatorname{piv}_>(\boldsymbol{q}) = \operatorname{piv}_<(\boldsymbol{a}) = k$ and $q_k a_k = 1$;*
3. *for all $\boldsymbol{\ell} \in L$ and $\beta = \bigl(\langle \boldsymbol{a}, \boldsymbol{x} \rangle = 0\bigr) \in \mathcal{E}$, $\operatorname{piv}_>(\boldsymbol{\ell}) \neq \operatorname{piv}_<(\boldsymbol{a})$;*
4. $\operatorname{gcon}(\mathcal{C}) = \operatorname{ggen}(\mathcal{G})$.

Proof. Let $\hat{\mathcal{G}} := \bigl(\hat{L}, \hat{Q} \cup \hat{P}, \{\boldsymbol{0}\}\bigr)$ be the homogeneous form for $\mathcal{G}$ where

$$\hat{L} := \bigl\{ (0, \boldsymbol{\ell}^{\mathrm{T}})^{\mathrm{T}} \bigm| \boldsymbol{\ell} \in L \bigr\},\ \hat{Q} := \bigl\{ (0, \boldsymbol{q}^{\mathrm{T}})^{\mathrm{T}} \bigm| \boldsymbol{q} \in Q \bigr\},\ \hat{P} := \bigl\{ (1, \boldsymbol{p}^{\mathrm{T}})^{\mathrm{T}} \bigm| \boldsymbol{p} \in P \bigr\}.$$

Let also $\hat{\mathcal{C}} = \hat{\mathcal{F}} \cup \hat{\mathcal{E}}$ be the homogeneous form for $\mathcal{C}$ where

$$\hat{\mathcal{E}} := \Bigl\{ \langle (-b, \boldsymbol{a}^{\mathrm{T}})^{\mathrm{T}}, \hat{\boldsymbol{x}} \rangle = 0 \Bigm| \bigl(\langle \boldsymbol{a}, \boldsymbol{x} \rangle = b \bigr) \in \mathcal{E} \Bigr\},$$
$$\hat{\mathcal{F}} := \Bigl\{ \langle f^{-1}(-b, \boldsymbol{a}^{\mathrm{T}})^{\mathrm{T}}, \hat{\boldsymbol{x}} \rangle \equiv_1 0 \Bigm| \bigl(\langle \boldsymbol{a}, \boldsymbol{x} \rangle \equiv_f b \bigr) \in \mathcal{F} \Bigr\} \cup \bigl\{ \langle \boldsymbol{e}_0, \hat{\boldsymbol{x}} \rangle \equiv_1 0 \bigr\}.$$

Let $\hat{\mathcal{E}} = \hat{E}^{\mathrm{T}} \hat{\boldsymbol{x}} = \boldsymbol{0}$ and $\hat{\mathcal{F}} = \hat{F}^{\mathrm{T}} \hat{\boldsymbol{x}} \equiv_1 \boldsymbol{0}$ where $\hat{\boldsymbol{x}} = (x_0, \dots, x_n)^{\mathrm{T}}$. Then, as $\mathcal{G}$ and $\mathcal{C}$ are in minimal form, by the definition of homogeneous form, items (1), (2), (3), and (4) hold if and only if the following hold:

5. $\# \hat{Q} = \# \hat{\mathcal{F}} = n + 1 - \# \hat{L} - \# \hat{\mathcal{E}}$;
6. there exists $\boldsymbol{q} \in \hat{Q}$ if and only if there exists $\beta = \bigl(\langle \boldsymbol{a}, \hat{\boldsymbol{x}} \rangle \equiv_1 0 \bigr) \in \hat{\mathcal{F}}$, such that, for some $k \in \{0, \dots, n\}$, $\mathrm{piv}_{>}(\boldsymbol{q}) = \mathrm{piv}_{<}(\boldsymbol{a}) = k$ and $q_k a_k = 1$;
7. for all $\boldsymbol{\ell} \in \hat{L}$ and $\beta = \bigl(\langle \boldsymbol{a}, \hat{\boldsymbol{x}} \rangle = 0 \bigr) \in \hat{\mathcal{E}}$, $\mathrm{piv}_{>}(\boldsymbol{\ell}) \neq \mathrm{piv}_{<}(\boldsymbol{a})$;
8. $\mathrm{gcon}(\hat{\mathcal{C}}) = \mathrm{ggen}(\hat{\mathcal{G}})$

It therefore remains to prove that (5), (6), (7), and (8) hold.

Let M, N be matrices in $\mathbb{Z}^{n+1}$, whose vectors are of the form $\boldsymbol{e}_i$, $1 \leq i \leq n$, and such that $(\hat{L}, \hat{Q}, M)$ and $(N, \hat{F}, \hat{E})$ are square and nonsingular. Then we let

$$(N, \hat{F}, \hat{E}) := \bigl((\hat{L}, \hat{Q}, M)^{-1} \bigr)^{\mathrm{T}}$$

where $\# N = \# \hat{L}$, $\# \hat{Q} = \# \hat{F}$ and $\# \hat{E} = \# N$; ensuring (5) holds. Note that matrix inversion and transposition are both computable and invertible.

Let $\ell = \# \hat{L}$, $q = \# \hat{Q} - 1$ and $m = \# M$. Then

$$\begin{aligned}
\hat{\boldsymbol{x}} \in \hat{\mathcal{G}} &\iff \hat{\boldsymbol{x}} = \hat{Q}\boldsymbol{\pi} + \hat{L}\boldsymbol{\lambda} + M\boldsymbol{0}, && \text{for } \boldsymbol{\lambda} \in \mathbb{R}^{\ell}, \boldsymbol{\pi} \in \mathbb{Z}^{q+1}, \boldsymbol{0} \in \mathbb{Z}^{m} \\
&\iff \hat{\boldsymbol{x}} = (\hat{Q}, \hat{L}, M)(\boldsymbol{\pi}^{\mathrm{T}}, \boldsymbol{\lambda}^{\mathrm{T}}, \boldsymbol{0}^{\mathrm{T}})^{\mathrm{T}}, && \text{for } \boldsymbol{\lambda} \in \mathbb{R}^{\ell}, \boldsymbol{\pi} \in \mathbb{Z}^{q+1}, \boldsymbol{0} \in \mathbb{Z}^{m} \\
&\iff (\hat{Q}, \hat{L}, M)^{-1} \hat{\boldsymbol{x}} = (\boldsymbol{\pi}^{\mathrm{T}}, \boldsymbol{\lambda}^{\mathrm{T}}, \boldsymbol{0}^{\mathrm{T}})^{\mathrm{T}}, && \text{for } \boldsymbol{\lambda} \in \mathbb{R}^{\ell}, \boldsymbol{\pi} \in \mathbb{Z}^{q+1}, \boldsymbol{0} \in \mathbb{Z}^{m}.
\end{aligned}$$

As $\hat{\mathcal{G}}$ is in minimal form (so that the set of vectors in $(\hat{L}, \hat{Q})$ is a subset of the set of vectors in a triangular matrix) and M adds the missing columns with only the diagonal elements being non-zero, $(\hat{Q}, \hat{L}, M)$ is a permutation of a lower triangular matrix. Thus the transposed inverse $(N, \hat{F}, \hat{E})$ is the same permutation of an upper triangular matrix so that $\hat{\mathcal{C}}$ is also a congruence system in minimal form. The generating system $\hat{\mathcal{G}}$ is the homogeneous form of a system that contains a point if and only if there is an element $\boldsymbol{q} \in \hat{Q}$ such that $q_0 = 1$ and $\mathrm{piv}_{>}(\boldsymbol{q}) = 1$; that is, if and only if there is also an element $\boldsymbol{a} \in \hat{F}$ such that $a_0 = 1$ and $\mathrm{piv}_{<}(\boldsymbol{a}) = 1$. Thus $\hat{\mathcal{C}}$ is a congruence system in minimal homogeneous form if and only if $\hat{\mathcal{G}}$ is a generator system in minimal homogeneous form.

More generally, for vectors $\boldsymbol{v}_i$ and $\boldsymbol{w}_i$ in $(N, \hat{F}, \hat{E})$ and $(\hat{L}, \hat{Q}, M)$, respectively, $\mathrm{piv}_{<}(\boldsymbol{v}_i) = \mathrm{piv}_{>}(\boldsymbol{w}_i) = k$ and $v_{ik} w_{ik} = 1$. Since $\# N = \# \hat{L}$, $\# \hat{Q} = \# \hat{F}$ and $\# \hat{E} = \# M$ we have proved items (6) and (7).

Since $(N, \hat{F}, \hat{E}) = \bigl((\hat{Q}, \hat{L}, M)^{-1} \bigr)^{\mathrm{T}}$ we have that $\hat{\boldsymbol{x}} \in \mathrm{ggen}(\hat{\mathcal{G}})$ if and only if $N^{\mathrm{T}} \hat{\boldsymbol{x}} = \boldsymbol{\lambda}$, $\hat{F}^{\mathrm{T}} \hat{\boldsymbol{x}} = \boldsymbol{\pi}$, $\hat{E}^{\mathrm{T}} \hat{\boldsymbol{x}} = \boldsymbol{0}$ for some $\boldsymbol{\lambda} \in \mathbb{R}^{\ell}, \boldsymbol{\pi} \in \mathbb{Z}^{q+1}$. Since $\boldsymbol{\lambda} \in \mathbb{R}^{\ell}$, the condition $N^{\mathrm{T}}\hat{\boldsymbol{x}} = \boldsymbol{\lambda}$ is vacuous and, as $\boldsymbol{\pi} \in \mathbb{Z}^{q+1}$, the condition $\hat{F}^{\mathrm{T}}\hat{\boldsymbol{x}} = \boldsymbol{\pi}$ holds if and only if $\hat{F}^{\mathrm{T}}\hat{\boldsymbol{x}} \equiv_1 \mathbf{0}$. Thus we have that $\hat{\boldsymbol{x}} \in \operatorname{ggen}(\hat{\mathcal{G}})$ if and only if $\hat{F}^{\mathrm{T}}\hat{\boldsymbol{x}} \equiv_1 \mathbf{0}$, $\hat{E}^{\mathrm{T}}\hat{\boldsymbol{x}} = \mathbf{0}$. Therefore $\operatorname{gcon}(\hat{\mathcal{C}}) = \operatorname{ggen}(\hat{\mathcal{G}})$ and (8) holds. □

The proofs of Propositions 3 and 4 depend on the following lemma. This shows that if one grid is a subset of another then the pivot elements of the proper congruences of the larger grid must be divisible by the corresponding pivot elements of the smaller grid.

Lemma 2. *Let $\mathcal{L}_1 = \operatorname{gcon}(\mathcal{C}_1)$, $\mathcal{L}_2 = \operatorname{gcon}(\mathcal{C}_2)$ be non-empty grids in $\mathbb{G}_n$ such that $\mathcal{L}_1 \subseteq \mathcal{L}_2$ and the congruence systems $\mathcal{C}_1$ and $\mathcal{C}_2$ are in minimal form. Then, for each $\gamma = \bigl(\langle \boldsymbol{c}, \boldsymbol{x}\rangle \equiv_g d\bigr) \in \mathcal{C}_2$, there exists $\beta = \bigl(\langle \boldsymbol{a}, \boldsymbol{x}\rangle \equiv_f b\bigr) \in \mathcal{C}_1$ such that* $\operatorname{piv}_<(\boldsymbol{a}) = \operatorname{piv}_<(\boldsymbol{c}) = k$ *and either $f = g = 0$ or $g \neq 0$ and $ga_k \mid fc_k$.*

Proof. Suppose $\gamma = \bigl(\langle \boldsymbol{c}, \boldsymbol{x}\rangle \equiv_g d\bigr) \in \mathcal{C}_2$ and $k = \operatorname{piv}_<(\boldsymbol{c})$. Then, as $\mathcal{L}_1 \subseteq \mathcal{L}_2$, $\mathcal{L}_1 \subseteq \operatorname{gcon}\bigl(\{\gamma\}\bigr)$. Let $\mathcal{G}_1 = \bigl(L_1, Q_1, \{\boldsymbol{p}\}\bigr)$ be a generator system for $\mathcal{L}_1$ in minimal form constructed as in Lemma 1 from $\mathcal{C}_1$.

We first prove that there exists $\beta = \bigl(\langle \boldsymbol{a}, \boldsymbol{x}\rangle \equiv_f b\bigr) \in \mathcal{C}_1$ such that $\operatorname{piv}_<(\boldsymbol{a}) = k$. To see this, suppose instead that, for all $\beta = \bigl(\langle \boldsymbol{a}, \boldsymbol{x}\rangle \equiv_f b\bigr) \in \mathcal{C}_1$, $\operatorname{piv}_<(\boldsymbol{a}) \neq k$. Then, by Lemma 1, there must exist a line $\boldsymbol{\ell} \in L_1$ such that $\operatorname{piv}_>(\boldsymbol{\ell}) = k$; hence $\langle \boldsymbol{c}, \boldsymbol{\ell}\rangle = c_k \ell_k \neq 0$. Since $\mathcal{L}_1 \subseteq \operatorname{gcon}\bigl(\{\gamma\}\bigr)$, this implies that

$$\bigl\langle \boldsymbol{c}, (\boldsymbol{p} + r\boldsymbol{\ell})\bigr\rangle = \langle \boldsymbol{c}, \boldsymbol{p}\rangle + rc_k\ell_k \equiv_g d,$$

for all $r \in \mathbb{R}$; which is a contradiction.

We next show that if $g = 0$ then $f = 0$. To see this, suppose instead that $g = 0$ but $f \neq 0$. Then, by Lemma 1, there exists $\boldsymbol{q} \in Q_1$ such that $\operatorname{piv}_>(\boldsymbol{q}) = k$; hence $\langle \boldsymbol{c}, \boldsymbol{q}\rangle = c_k q_k \neq 0$. Since $\mathcal{L}_1 \subseteq \operatorname{gcon}\bigl(\{\gamma\}\bigr)$, this implies that

$$\bigl\langle \boldsymbol{c}, (\boldsymbol{p} + r\boldsymbol{q})\bigr\rangle = \langle \boldsymbol{c}, \boldsymbol{p}\rangle + mc_k q_k \equiv_g d,$$

for all $m \in \mathbb{Z}$; which is a contradiction.

We now assume that $g \neq 0$ and show that $ga_k \mid fc_k$. This is trivial if $f = 0$; therefore, suppose $f \neq 0$. By Lemma 1, there exists a parameter $\boldsymbol{q}$ in Q_1 such that $\operatorname{piv}_>(\boldsymbol{q}) = k$ (so that $q_k c_k \neq 0$) and $q_k = fa_k^{-1}$. Thus, as $\mathcal{L}_1 \subseteq \operatorname{gcon}\bigl(\{\gamma\}\bigr)$, $\langle \boldsymbol{q}, \boldsymbol{c}\rangle = q_k c_k = mg$, for some $m \in \mathbb{Z} \setminus \{0\}$. Therefore we must have $ga_k \mid fc_k$. □

The next lemma, which is needed in the proofs of Propositions 3 and 4 shows that if, two grids, one a subset of the other are described by two congruence systems in strong minimal form that are pivot equivalent, then, relative to the affine hull of the grids, pivot equivalent congruences in these systems are the same.

Lemma 3. *Let $\mathcal{L}_1 = \operatorname{gcon}(\mathcal{C}_1)$, $\mathcal{L}_2 = \operatorname{gcon}(\mathcal{C}_2)$ be non-empty grids in $\mathbb{G}_n$ where $\mathcal{L}_1 \subseteq \mathcal{L}_2$ and the congruence systems $\mathcal{C}_1$ and $\mathcal{C}_2$ are in strong minimal form. Suppose that $\mathcal{C}_1$ is pivot equivalent to $\mathcal{C}_2$. Then, for each $\beta \in \mathcal{C}_1$ and $\gamma \in \mathcal{C}_2$ such that $\beta \Uparrow \gamma$,*

$$\operatorname{gcon}\bigl(\{\beta\}\bigr) \cap \operatorname{affine.hull}(\mathcal{L}_1) = \operatorname{gcon}\bigl(\{\gamma\}\bigr) \cap \operatorname{affine.hull}(\mathcal{L}_1). \tag{3}$$

Proof. Let $\beta = \big(\langle \boldsymbol{a}, \boldsymbol{x}\rangle \equiv_f b\big) \in \mathcal{C}_1$. By the definition of pivot equivalence for congruence systems in Section 3, as $\mathcal{C}_1 \Uparrow \mathcal{C}_2$ there exists $\gamma = \big(\langle \boldsymbol{c}, \boldsymbol{x}\rangle \equiv_g d\big) \in \mathcal{C}_2$ such that $\beta \Uparrow \gamma$. We show that equation (3) holds. By the definition of pivot equivalence for congruences in Section 2, $\mathrm{piv}_<(\boldsymbol{a}) = \mathrm{piv}_<(\boldsymbol{c}) = k$ and $ga_k = fc_k$. Thus as $a_k, c_k \neq 0$, either $f = g = 0$ and β, γ are both equalities, or we have $f, g \neq 0$ so that β, γ are both proper congruences and we can assume that $f = g = 1$.

Let $\mathcal{E}_1$ be the set of equalities in $\mathcal{C}_1$. By Gaussian elimination, the set $\mathcal{E}_1$ can be transformed to the set of equalities $\mathcal{E}_1'$ such that $\mathrm{gcon}(\mathcal{E}_1') = \mathrm{gcon}(\mathcal{E}_1) = \mathrm{affine.hull}(\mathcal{L}_1)$ and has the following property: let $\mathcal{E}_1' = \{\beta_1, \dots, \beta_m\}$ such that, for each $i \in \{1, \dots, m\}$, $\mathrm{piv}_<(\beta_i) = k_i$ and $\beta_i = \big(\langle \boldsymbol{a}_i, \boldsymbol{x}\rangle = b_i\big)$; then, for each $i, j \in \{1, \dots, m\}$ where $i \neq j$, we have $a_{ik_j} = 0$. Let

$$\boldsymbol{a}'' = \boldsymbol{a} - \boldsymbol{c} - \sum_{i=1}^{m} \frac{(a_{k_i} - c_{k_i})}{a_{k_i}} \boldsymbol{a}_i, \qquad b'' = b - d - \sum_{i=1}^{m} \frac{(a_{k_i} - c_{k_i})}{a_{k_i}} b_i. \tag{4}$$

Let $\beta'' := \big(\langle \boldsymbol{a}'', \boldsymbol{x}\rangle \equiv_1 b''\big)$ and $\ell := \mathrm{piv}_<(\boldsymbol{a}'')$. Then $\mathcal{L}_1 \subseteq \mathrm{gcon}\big(\{\beta''\}\big) \subseteq \mathcal{L}_2$. Moreover, for any equality $\beta_i \in \mathcal{E}_1'$, $\mathrm{piv}_<(\boldsymbol{a}_i) \neq \ell$. Thus, if β, γ are equalities, $\boldsymbol{a''} = \boldsymbol{0}$ and, as $\mathcal{C}_1$ is consistent, $b'' = 0$. Therefore $\mathrm{gcon}(\mathcal{E}_1') \subseteq \mathrm{gcon}\big(\{\beta\}\big)$ and $\mathrm{gcon}(\mathcal{E}_1') \subseteq \mathrm{gcon}\big(\{\gamma\}\big)$. Hence equation (3) holds.

Consider now the case when β, γ are proper congruences. We first show that $\boldsymbol{a''} = \boldsymbol{0}$ and $b'' \in \mathbb{Z}$. Without loss of generality we can assume that $f = g = 1$. Note that, as $a_k = c_k$ we have $\ell < k$. We show $\ell = 0$; suppose, to the contrary that $\ell > 0$. Since $\mathcal{L}_1 \subseteq \mathcal{L}_2$, and $\gamma \in \mathcal{C}_2$, we have $\mathcal{L}_1 \subseteq \mathrm{gcon}\big(\{\gamma\}\big)$; so we can apply Lemma 2 to the grids $\mathcal{L}_1$ and $\mathrm{gcon}\big(\{\gamma\}\big)$. Thus there exists a proper congruence $\beta' = \big(\langle \boldsymbol{a}', \boldsymbol{x}\rangle \equiv_1 b'\big) \in \mathcal{C}_1$ where $\mathrm{piv}_<(\boldsymbol{a}') = \ell$ and $a'_\ell \mid a''_\ell$. Note that the number of proper congruences p_1 in $\mathcal{C}_1$ is equal to the number of proper congruences p_2 in $\mathcal{C}_2$; since by Lemma 2, $p_2 \leq p_1$ and, by hypothesis, $p_1 \leq p_2$. Therefore, by Lemma 2, there must exist a proper congruence $\gamma' = \big(\langle \boldsymbol{c}', \boldsymbol{x}\rangle \equiv_1 d'\big) \in \mathcal{C}_2$ where $\mathrm{piv}_<(\boldsymbol{c}') = \ell$ and $c'_\ell = a'_\ell$. Now as $\mathcal{C}_1$ and $\mathcal{C}_2$ are in strong minimal form, by Definition 5,

$$-\frac{a'_\ell}{2} < a_\ell \leq \frac{a'_\ell}{2} \quad \text{and} \quad -\frac{c'_\ell}{2} < c_\ell \leq \frac{c'_\ell}{2}.$$

Therefore $-a'_\ell < a''_\ell < a'_\ell$. It follows that, as $a'_\ell | a''_\ell$, $a''_\ell = 0$, contradicting the assumption that $\mathrm{piv}_<(\boldsymbol{a}'') = \ell > 0$. Therefore $\boldsymbol{a}'' = \boldsymbol{0}$ and β'' is the relation $b'' \equiv_1 0$ for some $b'' \in \mathbb{Z}$.

It follows that, by (4),

$$\boldsymbol{a} - \boldsymbol{c} = \sum_{i=1}^{m} \frac{(a_{k_i} - c_{k_i})}{a_{k_i}} \boldsymbol{a}_i, \qquad b - d \equiv_1 \sum_{i=1}^{m} \frac{(a_{k_i} - c_{k_i})}{a_{k_i}} b_i.$$

Thus

$$\mathrm{gcon}\big(\{\gamma, \beta_1, \dots, \beta_m\}\big) \subseteq \mathrm{gcon}\big(\{\beta\}\big) \quad \mathrm{gcon}\big(\{\beta, \beta_1, \dots, \beta_m\}\big) \subseteq \mathrm{gcon}\big(\{\gamma\}\big)$$

so that

$$\mathrm{gcon}\big(\{\gamma, \beta_1, \dots, \beta_m\}\big) = \mathrm{gcon}\big(\{\beta, \beta_1, \dots, \beta_m\}\big).$$

Hence equation (3) holds. $\square$

Proof (of Proposition 3). Suppose $\mathcal{C}_1, \mathcal{C}_2 \neq \varnothing$. By Lemma 2, for each $\beta = \big(\langle \boldsymbol{a}, \boldsymbol{x} \rangle \equiv_f b\big) \in \mathcal{C}_1$ there exists $\gamma = \big(\langle \boldsymbol{c}, \boldsymbol{x} \rangle \equiv_g d\big) \in \mathcal{C}_2$ such that $\mathrm{piv}_<(\boldsymbol{a}) = \mathrm{piv}_<(\boldsymbol{c}) = k$ and either $f = g = 0$ or $f \neq 0$ and $fc_k \mid ga_k$.

Also by Lemma 2, for $\gamma = \big(\langle \boldsymbol{c}, \boldsymbol{x} \rangle \equiv_g d\big) \in \mathcal{C}_2$ there exists $\beta' = \big(\langle \boldsymbol{a}', \boldsymbol{x} \rangle \equiv_{f'} b'\big) \in \mathcal{C}_1$, such that $\mathrm{piv}_<(\boldsymbol{a}') = \mathrm{piv}_<(\boldsymbol{c}) = k$ and either $f' = g = 0$ or $g \neq 0$ and $ga'_k \mid f'c_k$. However as $\mathcal{C}_1$ is in strong minimal form $\beta = \beta'$. Therefore, by Definition 2, $a_k, c_k > 0$, hence $fc_k = ga_k$.

Therefore, for each $\beta \in \mathcal{C}_1$ there exists $\gamma \in \mathcal{C}_2$ such that $\beta \Uparrow \gamma$ so that we can apply Lemma 3 to prove the thesis. $\square$

The following Proposition is needed for the proof of Proposition 4, and shows that if two grids, one a subset of the other are described by two congruence systems in minimal form that are pivot equivalent, then, these grids are the same.

Proposition 7. *Let $\mathcal{L}_1 = \mathrm{gcon}(\mathcal{C}_1)$ and $\mathcal{L}_2 = \mathrm{gcon}(\mathcal{C}_2)$ be non-empty grids in $\mathbb{G}_n$ such that $\mathcal{L}_1 \subseteq \mathcal{L}_2$. If $\mathcal{C}_1$ and $\mathcal{C}_2$ are pivot equivalent congruence systems in minimal form, then $\mathcal{L}_1 = \mathcal{L}_2$.*

Proof. By Proposition 2, we can convert $\mathcal{C}_1$ and $\mathcal{C}_2$ to strong minimal form, $\mathcal{C}'_1$ and $\mathcal{C}'_2$ respectively, so that, for $i = 1, 2$, $\mathcal{L}_i = \mathrm{gcon}(\mathcal{C}'_i)$ and $\mathcal{C}'_i$ is pivot equivalent to $\mathcal{C}_i$. By Lemma 3, for each $\beta \in \mathcal{C}'_1$ and $\gamma \in \mathcal{C}'_2$,

$$\mathrm{gcon}\big(\{\beta\}\big) \cap \mathrm{affine.hull}(\mathcal{L}_1) = \mathrm{gcon}\big(\{\gamma\}\big) \cap \mathrm{affine.hull}(\mathcal{L}_1).$$

Thus $\mathcal{L}_1 = \mathcal{L}_2$ as required. $\square$

Proof (of Proposition 4). In order to show that '∇_c' is a widening operator, we prove that conditions (1) and (2) in Definition 4 hold. Let $\mathcal{L}_1 = \mathrm{gcon}(\mathcal{C}_1)$, $\mathcal{L}_2 = \mathrm{gcon}(\mathcal{C}_2) \in \mathbb{G}_n$, where $\mathcal{L}_1 \subseteq \mathcal{L}_2$, $\mathcal{C}_1$ is in minimal form and $\mathcal{C}_2$ is in strong minimal form.

By Definition 7, if $\mathcal{L}_1 = \varnothing$ or $\dim(\mathcal{L}_1) < \dim(\mathcal{L}_2)$, then $\mathcal{L}_1 \nabla_c \mathcal{L}_2 = \mathcal{L}_2$. Therefore, in this case, condition (1) holds. Clearly, the empty grid can occur only as the first element of a strictly increasing chain of grids; moreover, if $\mathcal{L}$ and $\mathcal{L}'$ are any two successive and distinct grids in the increasing chain of condition (2) in Definition 4, then $0 \leq \dim(\mathcal{L}) \leq \dim(\mathcal{L}') \leq n$. Hence, the case when $\mathcal{L}_1 = \varnothing$ or $\dim(\mathcal{L}_1) < \dim(\mathcal{L}_2)$ hold can occur no more than a finite number of times in such a chain.

Suppose now that $\mathcal{L}_1 \neq \varnothing$ and $\dim(\mathcal{L}_1) = \dim(\mathcal{L}_2)$, so that the second case of the widening computation applies (note that, due to the inclusion hypothesis, $\dim(\mathcal{L}_1) > \dim(\mathcal{L}_2)$ cannot hold), and let $\mathcal{C}_S$ be as given in Definition (7) in Definition 7. Then, since $\mathcal{C}_S \subseteq \mathcal{C}_2$, condition (1) holds. By Proposition 7, if

$\mathcal{C}_S = \mathcal{C}_2$, we have $\mathcal{L}_1 = \mathcal{L}_2$; thus, if $\mathcal{L}_1 \neq \mathcal{L}_2$, we have $\#\mathcal{C}_S < \#\mathcal{C}_2$. By Lemma 2, as $\mathcal{C}_1$ and $\mathcal{C}_2$ are in minimal form, it follows that $\#\mathcal{C}_2 \leq \#\mathcal{C}_1$ so that, if $\mathcal{L}_1 \neq \mathcal{L}_2$, $\#\mathcal{C}_S < \#\mathcal{C}_1$. Therefore condition (2) of Definition 4 holds. □

Proof (of Proposition 5). Suppose that $\mathcal{G} = \big((L, Q, P)\big)$ is not in strong minimal form. Then, by Definition 8, then there exists a parameter $\boldsymbol{u} \in Q$, such that the following holds:

1. there exists $1 \leq i \leq n$ and a parameter $\boldsymbol{v} \in Q \setminus \{\boldsymbol{u}\}$ where $\mathrm{piv}_>(\boldsymbol{v}) = i$ and either $2u_i \leq -v_i$ or $2u_i > v_i$.

If condition (1) does not hold for $\boldsymbol{u}$; let $\boldsymbol{u}' = \boldsymbol{u}$.

Suppose now that condition (1) holds and that $k \in \{1, \dots, n\}$ is such that $n + 1 - k$ is the minimum value for the index i for which this condition holds.

We show, by induction on k, that there exists a sequence of at most n transformations, each of which having complexity $\mathrm{O}\big(n\big)$, from $\boldsymbol{u}$ to the parameter $\boldsymbol{u}'$, such that, if $\mathcal{G}' := \big(\mathcal{G} \setminus \{\boldsymbol{u}\}\big) \cup \{\boldsymbol{u}'\}$, then $\mathrm{ggen}(\mathcal{G}') = \mathrm{ggen}(\mathcal{G})$ and condition (1) (when $\boldsymbol{u}$ is replaced by $\boldsymbol{u}'$) does not hold.

As $\mathcal{G}$ is in minimal form, $i > \mathrm{piv}_>(\boldsymbol{u})$. Let

$$\boldsymbol{u}'' = \begin{cases} \boldsymbol{u} - \left\lceil \frac{u_i}{v_i} \right\rceil \boldsymbol{v}, & \text{if } u_i > \frac{v_i}{2}; \\ \boldsymbol{u} - \left\lfloor \frac{u_i}{v_i} \right\rfloor \boldsymbol{v}, & \text{if } u_i \leq \frac{v_i}{2}. \end{cases}$$

Then $-v_i < 2u''_i \leq v_i$. Also, for $1 \leq j \leq i - 1$, we have $v_j = 0$ so that $u_j = u''_j$ and $\mathrm{piv}_>(\boldsymbol{u}'') = \mathrm{piv}_>(\boldsymbol{u})$. Letting $\mathcal{G}'' := \big(\mathcal{G} \setminus \{\boldsymbol{u}\}\big) \cup \{\boldsymbol{u}''\}$, we have $\mathrm{ggen}(\mathcal{G}'') = \mathrm{ggen}(\mathcal{G})$. Note that this transformation has a complexity $\mathrm{O}\big(n\big)$. As $n + 1 - k''$, the minimum index such that condition (1) holds for $\boldsymbol{u}''$, is strictly greater than $n + 1 - k$, we have that k'' is strictly less than k, therefore we can apply the inductive hypothesis to $\mathcal{G}''$ and $\boldsymbol{u}''$. Thus there is a sequence of at most $n - 1$ transformations from $\boldsymbol{u}''$ to $\boldsymbol{u}'$ such that, $\mathrm{ggen}\Big(\big(\mathcal{G}'' \setminus \{\boldsymbol{u}''\}\big) \cup \{\boldsymbol{u}'\}\Big) = \mathrm{ggen}(\mathcal{G}'')$ and condition (1) (when $\boldsymbol{u}$ is replaced by $\boldsymbol{u}'$) does not hold. Thus there is a sequence of at most n transformations from $\boldsymbol{u}$ to $\boldsymbol{u}'$ such that $\mathrm{ggen}\Big(\big(\mathcal{G} \setminus \{\boldsymbol{u}\}\big) \cup \{\boldsymbol{u}'\}\Big) = \mathrm{ggen}(\mathcal{G})$. As each of the individual steps has complexity $\mathrm{O}\big(n\big)$, the sequence of transformations has complexity $\mathrm{O}\big(n^2\big)$.

We repeat this sequence of transformations for each parameter in $\mathcal{G}$ to obtain a generator system $\mathcal{G}'$ such that, for each parameter $\boldsymbol{u}' \in \mathcal{G}'$, condition (1) does not hold. Thus, by Definition 8, $\mathcal{G}'$ is in strong normal form. Thus, as there are at most n parameters in $\mathcal{G}$ since, by hypothesis, $\mathcal{G}$ is in minimal form, the complexity of computing the strong minimal form is $\mathrm{O}\big(n^3\big)$. □

The following Proposition is needed for the proof of Proposition 6, and shows that if two grids, one a subset of the other are described by two generator systems in minimal form that are pivot equivalent, then, these grids are the same.

Proposition 8. *Let* $\mathcal{L}_1 = \mathrm{ggen}(\mathcal{G}_1)$ *and* $\mathcal{L}_2 = \mathrm{ggen}(\mathcal{G}_2)$ *be non-empty grids in* $\mathbb{G}_n$ *such that* $\mathcal{L}_1 \subseteq \mathcal{L}_2$*. If* $\mathcal{G}_1$ *and* $\mathcal{G}_2$ *are pivot equivalent generator systems in minimal form, then* $\mathcal{L}_1 = \mathcal{L}_2$*.*

Proof. Let $\mathcal{C}_1 = (\mathcal{F}_1, \mathcal{E}_1)$ and $\mathcal{C}_2 = (\mathcal{F}_2, \mathcal{E}_2)$ be congruence systems for $\mathcal{L}_1$ and $\mathcal{L}_2$ respectively, as constructed in Lemma 1 from the generator systems $\mathcal{G}_1$ and $\mathcal{G}_2$, respectively. Then by properties (2) and (3) in Lemma 1, $\mathcal{C}_1$ is pivot equivalent to $\mathcal{C}_2$. By hypothesis, $\mathcal{L}_1 \subseteq \mathcal{L}_2$. Thus, by Proposition 7, $\mathcal{L}_1 = \mathcal{L}_2$ as required. □

Proof (of Proposition 6). In order to show that '$\nabla_{\mathcal{G}}$' is a widening operator, we prove that conditions (1) and (2) in Definition 4 hold. Let $\mathcal{L}_1 = \mathrm{ggen}(\mathcal{G}_1)$, $\mathcal{L}_2 = \mathrm{ggen}(\mathcal{G}_2) \in \mathbb{G}_n$, where $\mathcal{L}_1 \subseteq \mathcal{L}_2$, $\mathcal{G}_1$ is in minimal form and $\mathcal{G}_2$ is in strong minimal form.

By Definition 10, if $\mathcal{L}_1 = \varnothing$ or $\dim(\mathcal{L}_1) < \dim(\mathcal{L}_2)$, then $\mathcal{L}_1 \nabla_{\mathcal{G}} \mathcal{L}_2 = \mathcal{L}_2$. Therefore, in this case, condition (1) holds. Clearly, the empty grid can occur only as the first element of a strictly increasing chain of grids; moreover, if $\mathcal{L}$ and $\mathcal{L}'$ are any two successive and distinct grids in the increasing chain of condition (2) in Definition 4, then $0 \leq \dim(\mathcal{L}) \leq \dim(\mathcal{L}') \leq n$. Hence, the case when $\mathcal{L}_1 = \varnothing$ or $\dim(\mathcal{L}_1) < \dim(\mathcal{L}_2)$ hold can occur no more than a finite number of times in such a chain.

Suppose now that $\mathcal{L}_1 \neq \varnothing$ and $\dim(\mathcal{L}_1) = \dim(\mathcal{L}_2)$, so that the second case of the widening computation applies (note that, due to the inclusion hypothesis, $\dim(\mathcal{L}_1) > \dim(\mathcal{L}_2)$ cannot hold), and let $\mathcal{G}_S = (L_S, Q_S, P_S)$ where

$$P_S = P_2, \quad Q_S = \{\, \boldsymbol{v} \in Q_2 \mid \exists \boldsymbol{u} \in Q_1 \,.\, \boldsymbol{u} \Downarrow \boldsymbol{v} \,\}, \quad L_S = L_2 \cup (Q_2 \setminus Q_S).$$

Then, if $\mathrm{ggen}(\mathcal{G}_S) \subseteq \mathrm{ggen}(\mathcal{G}_2)$, we have that $L_S \subseteq L_2$, but by Definition 10, $L_2 \subseteq L_S$, therefore $L_2 = L_S$. Hence we must have that for all $\boldsymbol{q}_2 \in Q_2, \exists \boldsymbol{q}_1 \in Q_1$ such that $\boldsymbol{q}_1 \Downarrow \boldsymbol{q}_2$, so condition (1) holds. Now as Q_1 is pivot equivalent to Q_2, then $\# Q_1 = \# Q_2$. Let $\mathcal{C}_1 = (\mathcal{F}_1, \mathcal{E}_1)$ and $\mathcal{C}_2 = (\mathcal{F}_2, \mathcal{E}_2)$ be congruence systems for $\mathcal{L}_1$ and $\mathcal{L}_2$ respectively, as constructed in Lemma 1. Then by Lemma 2, we get that $\# \mathcal{E}_1 \geq \# \mathcal{E}_2$, but since $\dim(\mathcal{L}_1) = \dim(\mathcal{L}_2)$ and $\mathcal{L}_1, \mathcal{L}_2$ are both in minimal form we must have $\# \mathcal{E}_1 = \# \mathcal{E}_2$, hence by Lemma 1, $\# L_1 = \# L_2$. Now as $\mathcal{L}_1 \subseteq \mathcal{L}_2$ we must have that for all lines $\boldsymbol{\ell}_1 \in L_1$ such that $\mathrm{piv}_>(\boldsymbol{\ell}_1) = k$, there exists $\boldsymbol{\ell}_2 \in L_2$ such that $\mathrm{piv}_>(\boldsymbol{\ell}_2) = k$. Therefore $\mathcal{G}_1 \Downarrow \mathcal{G}_2$. Hence by Proposition 8, $\mathcal{L}_1 = \mathcal{L}_2$. Thus, if $\mathcal{L}_1 \neq \mathcal{L}_2$, we know $\# L_1 \leq \# L_2$ and $\mathrm{ggen}(\mathcal{G}_2) \subseteq \mathrm{ggen}(\mathcal{G}_S)$ so condition (1) holds and we have three cases to consider. The first is if $\# Q_1 < \# Q_2$, then $\# L_S \geq \# L_2 + \# Q_2 - \# Q_1 > \# L_1$. The second case is if $\# Q_1 = \# Q_2$, then there exists $\boldsymbol{u} \in Q_1$ with $\mathrm{piv}_>(\boldsymbol{u}) = k$ such that for all $\boldsymbol{v} \in Q_2$ if $\mathrm{piv}_>(\boldsymbol{v}) = k$, then $u_k \neq v_k$, hence $\boldsymbol{v} \in L_S$, therefore $\# L_S > \# L_1$. Finally for the last case suppose $\# Q_1 > \# Q_2$, then there exists $\boldsymbol{u} \in Q_1$ with $\mathrm{piv}_>(\boldsymbol{u}) = k$ such that for all $\boldsymbol{v} \in Q_2$, $\mathrm{piv}_>(\boldsymbol{v}) \neq k$, hence $\boldsymbol{v} \in L_S$, therefore $\# L_S > \# L_1$. Therefore condition (2) of Definition 4 holds. □