ÉCOLE POLYTECHNIQUE

THÈSE

présentée par

François Masdupuy

pour obtenir le titre de

DOCTEUR de L'ÉCOLE POLYTECHNIQUE

Spécialité : INFORMATIQUE

ANALYSE SÉMANTIQUE RELATIONNELLE DES INDICES DE TABLEAUX PAR CONGRUENCES ET TRAPÉZOÏDES RATIONNELS

ARRAY INDICES RELATIONAL SEMANTIC ANALYSIS USING RATIONAL COSETS AND TRAPEZOIDS

Soutenue le 21 Décembre 1993, devant le jury composé de :

Patrice Quinton,	Président.
Chris Hankin,	
Pierre Jouvelot,	Rapporteurs.
Patrick Cousot,	
Nicolas Halbwachs,	
Alain Lichnewski,	Examinateurs.

RÉSUMÉ. L'analyse sémantique des variables numériques d'un programme consiste à déterminer statiquement et automatiquement des propriétés vérifiées par celles-ci à l'exécution. Différentes classes de propriétés (relations d'égalité, d'inégalité, de congruence) ont été étudiées. Cette thèse propose la généralisation d'une partie des modèles précédents. Plus particulièrement, en utilisant le cadre formel fourni par l'interprétation abstraite, nous proposons, d'une part, un ensemble de propriétés généralisant les intervalles et les classes de congruences de $\mathbb{Z}$ et, d'autre part, une généralisation des trapézoïdes et des systèmes d'équation linéaires de congruence de $\mathbb{Z}^n$. La définition d'une abstraction rationnelle de ces differentes propriétés permet d'obtenir des approximations, dont la complexité reste polynomiale en le nombre de variables considérées, des opérateurs sur les propriétés entières. Ces analyses, en général plus précises que la combinaison de celles dont elles sont issues, permettent de choisir dynamiquement le type de propriétés (entre relation d'inégalité ou de congruence) fournissant une information pertinente sur le programme considéré. Le modèle relationnel mis au point correspond à de nombreux motifs décrits par les indices des tableaux utilisés dans le domaine du calcul scientifique. Il est donc particulièrement bien adapté à l'analyse d'indices de tableaux, voire à la représentation abstraite de tableaux d'entiers.

Semantic analysis of program numerical variables consists in statically and automatically discovering properties verified at execution time. Different sets of properties (equality, inequality and congruence relations) have already been studied. This thesis proposes a generalization of some of the below patterns. More specifically, the abstract interpretation is used to design on the one hand a set of properties generalizing intervals and cosets on $\mathbb{Z}$ and on the other hand, a generalization of trapezoids and linear congruence equation systems on $\mathbb{Z}^n$. A rational abstraction of these properties is defined to get safe approximations, with a polynomial complexity in the number of the considered variables, of the integer properties operators. Those analyses, more precise than the combination of the analysis they come from in general, allow to dynamically choose the kind of properties (inequality or congruence relations) leading to relevant information for the considered program. The described relationnal analysis corresponds to numerous patterns encountered in the field of scientific computation. It is very well adapted to the analysis of array indices variables and also to the abstract description of integer arrays.

Je remercie

Messieur Chris Hankin, Professeur à l'Imperial College de Londres et Pierre Jouvelot, Chargé de Recherches à l'École des Mines de Paris d'avoir examiné en détail ce travail et de m'avoir fait part de leurs remarques constructives et cela malgré la sécheresse de l'organisation du mémoire.

Messieur Nicolas Halbwachs, Directeur de Recherches au CNRS, Alain Lichnewsky, Directeur Software de la société ACRI et Patrice Quinton, Directeur de Recherches au CNRS de m'avoir fait l'honneur de participer à ce jury.

Monsieur Patrick Cousot, Professeur d'Informatique à l'École Polytechnique et à l'École Normale Supérieure, qui, en tant que directeur de thèse a toujours été très attentif à mon travail et m'a permis de développer un sujet aride.

Que soient aussi remerciés mes collègues du Laboratoire d'Informatique de l'École Polytechnique et du Centre de Recherche de l'École des Mines de Paris, Alain Deutsch, Philippe Granger, Jan Stransky, François Irigoin et Pierre Jouvelot pour nos échanges scientifiques et leurs soutiens de tous ordres.

Je remercie tout particulièrement Laurence, mon épouse ainsi que Clara et Aure-Anne, mes enfants de l'enthousiasme qu'elles m'ont apporté.

Merci enfin à Jean Aymes et à Philippe Paules qui m'ont permis de partager leurs passions.

INTRODUCTION

La partie la plus importante du temps nécessaire à l'exécution de la plupart des programmes de calculs scientifiques est attribuée aux boucles effectuant des opérations sur des tableaux de données. La transformation et l'optimisation de ces boucles [LKK85, AK84, AK87, FW91, AN88, WL91b] en vue de la génération du code adapté à la machine-cible nécessite une bonne compréhension, lors de la compilation, de la structure des accès aux tableaux qui y sont effectués, lorsque ceux-ci ne sont pas considérés comme des scalaires [CCK90]. Des études pragmatiques [SLY89, EHLP91] ont été menées; elles justifient les méthodes plus systématiques parmi lesquelles on trouve par exemple la reconnaissance par idiomes [JD89, PP91, AHI90]. De très nombreuses analyses de dépendances qui permettent de valider la correction des transformations de boucles proposées ont été mises au point dans [GS90, Fea88a, Wal88, D'H89, BK89, MHL91]. D'autres méthodes analysent la localité des données référencées lors d'un accès à un tableau afin d'améliorer l'adéquation du code généré à la distribution et à la hiérarchie de la mémoire de la machine-cible dans [TP93, WL91a, GJG87, HKT92, KLS90, Ger89]. Toutes ces analyses reposent sur l'observation que la majorité des accès aux éléments des tableaux sont généralement des fonctions linéaires des indices des boucles les englobant [SLY89], du moins c'est le seul problème traitable de façon exacte [Dow90] et sont donc mises en échec par l'utilisation de tableaux d'indirections. C'est pour combler cette lacune que nous nous proposons de définir une méthode efficace qui permette d'analyser statiquement les tableaux.

Le tout premier choix à effectuer pour mettre au point notre analyse est celui du modèle utilisé pour trouver une approximation de la valeur exacte d'un tableau. La représentation d'un tableau par une fonction, qui est intuitivement la plus évidente, est malheureusement un mauvais point de départ car elle mène à des algorithmes de coût exponentiel [Jou87]. Nous avons donc choisi de représenter un tableau par une relation entre la valeur de ce tableau et son indice (éventuellement de dimension supérieure à un).

Le second choix, tout autant guidé par un souci d'efficacité, consiste à utiliser des relations sur les rationnels au lieu de relations sur les entiers (rappelons que les valeurs et indices d'un tableau d'indirection sont des entiers).

Pour ce qui est de la forme des relations utilisées, elles doivent au moins pouvoir exprimer les matrices bandes, triangulaires et autres caractérisations fréquentes de la localisation des valeurs des éléments d'un tableau [Cox88, BK93]. D'autre part, des analyses relationnelles désormais classiques existent. C'est le cas des égalités [Kar76] et inégalités [CH78] linéaires entre variables et des relations de congruences linéaires entre variables [Gra91b]. Nous avons choisi de nous baser, d'une part, sur un sous ensemble des polyèdres convexes et, d'autre part, sur les relations de congruences linéaires.

L'analyse par inégalités linéaires, autrement dit par polyèdres convexes, peut être simplifiée en restreignant les orientations possibles des différentes faces du polyèdre. Par exemple, en considérant que ces faces doivent être parallèles deux à deux et que pour la moitié d'entre elles leurs normales sont linéairement indépendantes, on obtient un cas particulier que nous nommons *trapézoïde*. Le modèle sur lequel s'appuie l'analyse que nous nous proposons de construire dans la partie 2 est une généralisation du trapézoïde et de la classe de congruence rationnelle relationnelle (solution d'un système d'équations linéaires de congruences rationnelles) et correspond donc aux solutions rationnelles d'un système d'équation de congruence à résidu borné de la forme :

$$\alpha_1 x_1 + \alpha_2 x_2 + \ldots + \alpha_n x_n \equiv \lambda \mod (q) \qquad \lambda \in [a, b] \tag{1}$$

dont tous les coefficients sont rationnels. Par interprétation abstraite [CC92b], on obtient donc une première analyse relationnelle de congruence de trapézoïdes concernant les variables entières d'un programme.

Cette analyse rationnelle des variables entières peut être rendue plus précise en considérant pour chaque congruence de trapézoïdes l'ensemble de points entiers qu'elle contient. Ce travail est effectué dans la partie 2 dans le cas $n = 1$ de la définition (1). On s'aperçoit que l'ensemble des points entiers d'un ensemble d'intervalles à extrémités rationnelles de la forme

$$\{[a, b], [a + q, b + q], [a + 2q, b + 2q], \ldots, [a + kq, b + kq], \ldots\}$$

est la réunion de classes de congruences entières de modulos identiques

$$\{l + mZ, l + \theta + mZ, \ldots, l + k\theta + mZ, \ldots, u + mZ\}$$

où l, u, m et θ sont entiers et θ et m sont premiers. Ce passage au modèle entier ne doit être utilisé au cours de l'analyse que dans les phases critiques (par exemple pour tester si un objet contient des points entiers ou non) de manière à ne pas augmenter le coût de l'analyse. Ce modèle étend celui des intervalles de [CC76] et des congruences de [Gra89] et l'analyse est par exemple plus précise qu'une analyse de flot de données comme [Gup90], ce qui n'est pas surprenant puisque le modèle des intervalles est déjà plus puissant que [Gup90] (voir l'exemple traité dans [CC92a]).

Mise à part son intégration dans un certain nombre de méthodes de détermination approchée des dépendances de données ou bien d'estimation de la localité des données comme annoncé initialement et décrit dans le chapitre VII, notre analyse permet en outre d'automatiser l'instanciation de programmes généraux à des cas de figure particuliers. Indirectement, elle est aussi intéressante pour des analyses qui peuvent se formuler numériquement comme le partage de données ou bien l'analyse des programmes communicants.

Ce travail est constitué de trois parties distinctes. Tout d'abord, des rappels concernant l'analyse sémantique par interprétation abstraite et plus particulièrement les analyses sémantiques des propriétés de congruences y sont donnés; on y trouvera d'une part la description du cadre de travail général ainsi qu'un rappel des analyses classiques développées dans la littérature concernant les variables numériques et fondées sur une interprétation abstraite, et d'autre part les propriétés spécifiques aux analyses de congruences qui sont utilisées dans la suite de la thèse.

La seconde partie de notre travail est dévolue à la construction de l'analyse sémantique des congruences d'intervalles, elle est elle-même divisée en deux chapitres. Dans un premier temps, nous construisons deux ensembles de propriétés caractérisant des ensembles d'entiers puis de rationnels et décrivons les relations fondamentales de comparaison et d'équivalence sur ces ensembles. Une fois ces contructions effectuées, nous établissons la connexion entre ces deux ensembles de propriétés; celle-ci permet de calculer, en temps constant, dans l'ensemble des propriétés rationnelles une approximation des opérations d'un coût non constant sur les propriétés entières. La construction de cette interprétation abstraite est complétée par la définition des instructions (ou primitives) abstraites et illustrée par un exemple.

La troisième partie de notre thèse correspond à la construction de l'analyse sémantique par congruence de trapézoïdes. Le plan de cette construction est en tout point semblable à celui de la partie précédente. Cette analyse relationnelle généralise l'analyse non relationnelle par congruence d'intervalles, de nombreuses opérations relationnelles sont réduites à des opérations non relationnelles construites auparavant. Quelques applications originales, notamment pour la représentation abstraite de tableaux d'entiers, sont données dans un dernier chapitre.

Part 1

SEMANTIC ANALYSIS OF NUMERICAL VARIABLES

CHAPTER I

STATIC ANALYSIS BY ABSTRACT INTERPRETATION

We introduce in this chapter the basic features of static program analysis based on operational semantics, called *abstract interpretation* and designed by P. and R. Cousot [CC77]. The abstract interpretation framework [CC92b] is here instantiated to the very special case for which it will be used in the rest of this work. The main characteristic that makes abstract interpretation a very powerful generalization of classical data flow analysis [MR88] is that its semantic bases provide analyses that can be easily proved correct and that the use of widening and narrowing operators allow to deal with infinite domains. Abstract interpretation is now widely used for static analysis in a great number of other fields than numerical variables analysis, for example logic program analysis [CC92a], type inference [Mon92] and alias analysis [Deu92].

The first part of this chapter briefly exposes the abstract interpretation framework while the second gives examples of such analyses in the field of program numerical variables.

1. The global design of the analysis

The first choice concerns the description of the meaning of a program. Two orthogonal concepts that are denotational (with functions that map program inputs to program outputs) and operational (with transition systems that describe every small step of the program) semantics are designed for this goal. Following [CC79], we take as standard semantics an operational semantics consisting of the transition system

$$(S, \tau, \iota, \varsigma)$$

where S is a set of program states, τ a transition relation binding a state to its possible successors, $\iota \subseteq S$ a set of initial states and $\varsigma \subseteq S$ a set of final states. Every program is associated with a transition system (for example, the set S of states of a program with m control points operating on n distinct integer variables is $[1, m] \times \mathbb{Z}^n$).

Then the forward collecting semantics is the sequences of finite partial execution traces, starting with an initial state, in which two consecutive states satisfy the transition relation. In order to discuss program invariance properties, we approximate the forward collecting semantics by the descendant states of the initial states, considering sets of states occurring in the original sequences of finite partial execution traces (indeed, program invariance properties do not deal with the execution order).

The so-called *concrete semantic domain* is the powerset $\mathbb{P}(S)$ of the set S. The concrete semantic function, which is used for associating its concrete semantics to each program, is the strongest post-condition operator

$$\begin{array}{rcccl} sp_\iota^\tau & : & \mathbb{P}(S) & \rightarrow & \mathbb{P}(S) \\ & & I & \mapsto & \iota \cup \{s | \exists s' \in I \ : \ (s, s') \in \tau\} \end{array}$$

More precisely, the meaning of a program associated to the transition system $(S, \tau, \iota, \varsigma)$ is the least fixpoint of the operator sp_ι^τ. Unfortunately, most of the time this fixpoint is uncomputable, and here, abstract interpretation introduces the fundamental concept of approximation. The idea is to introduce a new domain somehow connected to $\mathbb{P}(S)$ instead of the semantic domain, on which an approximation of the fixpoint equation is computable, providing an approximation of the exact solution. The connection is modeled by the use of semi-dual Galois connections between posets[1] ([O.44] for an inverse order on the abstract domain $L^\sharp$). For more precisions and definitions about the lattice theory see [Bir67].

Definition 1 (Galois connection (α, γ)). Let L and $L^\sharp$ be two posets. The pair of maps $(\alpha, \gamma) \in (L \rightarrow L^\sharp) \times (L^\sharp \rightarrow L)$ is a *semi-dual Galois connection* if α and γ are monotonic and

$$\mathcal{I} \sqsubseteq \gamma \circ \alpha \quad \wedge \quad \alpha \circ \gamma \sqsubseteq \mathcal{I}$$

where $\mathcal{I}$ is the identity function (either on L or on $L^\sharp$). α is called the abstraction function and γ the concretization function. Moreover, if α is surjective then $L^\sharp$ is isomorphic to a Moore family of L.

Hence the approximation is defined by a semi-dual Galois connection or dually by a Moore family (a meet closed subset of the semantic domain). The next theorem establishes the possibility of computing a safe approximation of the wanted least fixpoint on the concrete domain by a fixpoint computation on the abstract domain.

Theorem 2 (Fixpoint approximation [Cou81]). *Let L and $L^\sharp$ be two complete lattices, (α, γ) a semi-dual Galois connection, F a monotonic operator on L and $F^\sharp$ a monotonic operator on $L^\sharp$ greater than $\alpha \circ F \circ \gamma$. The least fixpoint of F is less than (or safely approximated by) the concretization of the least fixpoint of $F^\sharp$.*

This property is generalized to complete partial orders in [CC92b]. If the abstract domain is infinite or has too long ascending chains, we might be interested in approximating the least fixpoint computation in the abstract domain itself. The widening operator extrapolates the iteration process.

---

[1] Partial ordered sets.

Definition 3 (Widening operator ∇ [CC76]). Let L be a complete lattice. The operator $\nabla : L \times L \to L$ is a *widening* if

(1) it is greater than the least upper bound on L and
(2) for all increasing chain $(x_i)_{i\in\mathbb{N}}$ of elements of L, the series defined by $y_0 = x_0$ and $y_{n+1} = y_n \nabla x_{n+1}$ is stationary after a finite number of steps.

Practically, instead of a single fixpoint equation (shown to be the exact invariant of the program), a fixpoint equation system is considered, where the original equation domain is partitioned (with respect to the program control points for example). Each elementary program statement is then approximated by a monotonic operator on the abstract domain.

The design of an abstract interpretation is divided into three steps. First an abstract domain (with the corresponding abstraction and/or concretization) is extracted from the concrete semantic domain, possibly with an isomorphism to its machine representation if it is not directly implementable. Secondly, a set of abstract operators approximating as close by as possible (when the best one is not computable) the program statements are provided. Finally, the convergence of the iteration process for computing the fixpoint approximation is ensured, possibly introducing a widening operator.

Only the integer variables are of interest for our analysis, hence in a first approximation the set of considered states will be $\mathbb{Z}^n$ where n is the number of variables in the program and the concrete semantic domain is the powerset $\mathbb{P}(\mathbb{Z}^n)$ (standard semantics). For the presented analyses, the characterized states are either relationally approximated — and the semantic domain is really $\mathbb{P}(\mathbb{Z}^n)$ — or they are non relationally approximated and hence $\mathbb{P}(\mathbb{Z}^n)$ is replaced by $\mathbb{P}(\mathbb{Z})^n$. The next approximation introduces a set CP of properties of specific interest on integers, hence $\mathbb{P}(\mathbb{Z}^n)$ is now approximated by CP. Then for machine representation requirements, the integer properties are denoted as rational subsets (using the set AP), the intersection of which with $\mathbb{Z}^n$ will consist of the preceding integer properties. Two abstractions are considered that are the one between $\mathbb{P}(\mathbb{Z}^n)$ and CP and the other between CP and AP. CP is the abstract domain of the first approximation although it is the concrete domain of the second. The first abstraction is modeled by a single concretization function $\gamma_0 : CP \to \mathbb{P}(\mathbb{Z}^n)$ giving the meaning of an integer property in terms of integer tuples (in fact γ_0 is the extension of the identity to CP), the approximation ordering is therefore induced by the set inclusion relation on the powerset of $\mathbb{Z}^n$. The latter connection between concrete and abstract domain is established via a pair of abstraction and concretization function (α, γ). Examples of such connections appear in Chapters IV and VI where the concrete domain is CC (respectively RCC) and the abstract one is IC (respectively TC) with the particularity that (α, γ) (respectively $(\alpha^{\bowtie}, \gamma^{\bowtie})$) are not Galois connections.

2. Numerical variables analyses

This section presents some of the existing static analyses described in the literature dealing with program numerical variables. These are partitioned between the non relational and the relational ones.

2.1. Non relational analyses. The program numerical variables are considered separately. The interval analysis generalizes (i.e. is more precise than) the constant propagation and the sign analysis; the congruence analysis generalizes the parity analysis and the constant propagation. All these analyses concern either integer or rational values and have $\mathbb{P}(\mathbb{Z})$ as semantic domain for their integer valued version. The abstract operators are generally the best ones for the considered approximation.

Analysis of signs
The considered Moore family is here

$$\{\emptyset, \mathbb{Z}^{-*}, \mathbb{Z}^{+*}, \{0\}, \mathbb{Z}^{-}, \mathbb{Z}^{+}, \mathbb{Z}\}$$

An example of abstract statement is given for the abstract sum operator $\oplus$; it is point by point defined by:

$$\emptyset \oplus x = \emptyset,\ \mathbb{Z}^{-*} \oplus \mathbb{Z}^{+*} = \mathbb{Z},\ \mathbb{Z}^{-*} \oplus \{0\} = \mathbb{Z}^{-*}, \ldots$$

There is no need here for a widening because the lattice is finite and of height 3.

Constant propagation [Kil73]
The abstract lattice is here the set of all integer singletons, the empty set and $\mathbb{Z}$ ordered by inclusion. The abstract sum operator $\oplus$ is defined by:

$$\{x\} \oplus \{y\} = \{x + y\}, \emptyset \oplus \{y\} = \emptyset, \mathbb{Z} \oplus \{y\} = \mathbb{Z}$$

and is commutative. The height of the lattice is 2.

Interval analysis [CC76]
The abstract lattice is the set of possibly infinite integer intervals $[a, b]$ where $a, b \in \mathbb{Z} \cup \{-\infty, +\infty\}$ and $a \leq b$, completed with the emptyset and ordered by the set inclusion induced order. The abstract sum operator $\oplus$ is defined by:

$$[a, b] \oplus [c, d] = [a + c, b + d], \emptyset \oplus x = \emptyset$$

and is commutative. This lattice has an infinite height and a widening is needed, which extrapolates the increase of the interval bounds

$$[a, b]\nabla[c, d] = [\text{if } c < a \text{ then } -\infty \text{ else } a, \text{if } d > b \text{ then } +\infty \text{ else } b]$$

and its result is $\emptyset$ when at least one operand is $\emptyset$.

Parity analysis
The abstract domain is the four element lattice $\{\emptyset, 2\mathbb{Z}, 1+2\mathbb{Z}, \mathbb{Z}\}$. The abstract sum operator $\oplus$ is defined by:

$$\emptyset \oplus x = \emptyset,\ \mathbb{Z} \oplus x = \mathbb{Z},\ 2\mathbb{Z} \oplus 2\mathbb{Z} = (1 + 2\mathbb{Z}) \oplus (1 + 2\mathbb{Z}) = 2\mathbb{Z},\ (1 + 2\mathbb{Z}) \oplus 2\mathbb{Z} = (1 + 2\mathbb{Z})$$

and is commutative. There is no need of widening here.

Congruence analysis [Gra89]
The rational version of this analysis is exposed in the next chapter.

All these non relational analyses are rather simple but do not provide much information. They are not of interest for representing general array indexes, which is our purpose.

2.2. Relational analyses. These analyses compute approximations of the exact invariants where all the numerical variables are considered simultaneously, hence relationally. The linear constraints analysis generalizes the non relational interval analysis and the linear equalities analysis, while the linear congruences analysis generalizes the non relational congruence analysis.

Linear equalities [Kar76]
It considers the systems of equations such as

$$\sum_{i=1}^{n} \lambda_i x_i = \beta$$

Linear inequalities [CH78]
The semantic domain is $\mathbb{P}(\mathbb{Q}^n)$ and the abstract domain is the set of convex polyedras of $\mathbb{Q}^n$ represented by systems of equations of the kind

$$\sum_{i=1}^{n} \lambda_i x_i \leq \beta$$

The widening operator which is very frequently needed in such an analysis is based on experimentation and on the specific representation of a convex polyhedron by its system of generators.

Linear congruences [Gra91b]
The semantic domain is $\mathbb{P}(\mathbb{Z}^n)$ or $\mathbb{P}(\mathbb{Q}^n)$. The abstract domain corresponds to the solutions of the systems of linear congruence equations of the kind[2]

$$\sum_{i=1}^{n} \lambda_i x_i \equiv c \mod (q)$$

in $\mathbb{Z}^n$ or $\mathbb{Q}^n$. More details are given on this analysis in the next chapter and very often in the rest of this work.

The motivation for designing a new non relational integer semantic analysis is first to be able, using only one analysis, to discover program invariant approximations which would have been determined either by the interval or by the congruence analysis. This corresponds to automatically deciding, during the static analysis, which one of these analyses is convenient for every program point. The second goal is to determine invariant approximations when both interval and congruence analyses would have failed.

[2] $x \equiv y \mod (q)$ means $\exists k \in \mathbb{Z} \;\; x = y + kq$

```
   ind := 1;
   while nn > ind do begin
      for ii := 1 to ind do begin
         m := 2*ii -1;
         for j := 0 to ((2*nn - m) div 4*ind) do
            i := m + 4*ind*jj;
{S}         ... := data[i] + ... ;
{T}         ... := data[i+1] + ...      end end
      ind := 2*ind end;
```

FIGURE I.1. An extract of Fast Fourier Transform algorithm.

Let us consider the Fast Fourier Transform algorithm of figure I.1 coming from [PFTV86]. The accesses to the array `data` in statements {`S`} and {`T`} are summarized by the relation

$$\alpha - m \equiv 0 \mod (4) \quad \vee \quad \alpha - m \equiv 1 \mod (4)$$

where α stands for the indexes of accessed elements of the array `data`. Typically, congruence analysis will fail to summarize such information because of the consecutiveness of the two accessed elements, while interval analysis will fail because of the congruence character of the loop indices `jj` and `ind` in the expression of `i`. Other interesting information in order to parallelize the execution of these three nested loops could result of the parity of variable `m` and of the bounds on variable `ii`. This shows the need for an ambivalent analysis.

CHAPTER II

CONGRUENCE SEMANTIC ANALYSIS

In his PhD thesis [Gra91a], Granger has designed, using a common algebraic framework, four semantic analyses dealing with congruence properties of numerical variables. These analyses are classified into relational and non relational ones on one hand and into integer and rational ones on the other hand. In order to build our analyses, we are going to use many properties of Granger's rational analyses. The goal of this chapter is to recall the general framework of congruence semantic analyses and the main properties that are used in the rest of this work. All the properties figuring in this chapter are proved in [Gra91a]. After the formal definition of general cosets, first a special kind of cosets of the group of rational numbers $\mathbb{Q}$ are considered and, then, properties of the linear analysis based on the use of a special kind of cosets of $\mathbb{Q}^n$ are recalled.

DEFINITION 4 (COSETS). Let G be an abelian group and H be a subgroup of G. The equivalence classes of the equivalence relation of the kind $x - y \in H$ are called *cosets* modulo H. They have the form

$$a + H \quad = \quad \{x \in G \ / \ \exists h \in H, x = a + h\}$$

where a is an element of the coset.

The set of cosets of an abelian group is a lattice and hence fits the semantic analysis framework.

1. Rational arithmetical congruence analysis

The usual way to build a set of congruence properties is first to characterize a set of relevant subgroups of the considered original abelian group and then to consider the corresponding lattice of cosets. This is the purpose of the theorem 5 and the definition 6.

THEOREM 5 (FINITELY GENERATED SUBGROUPS OF $\mathbb{Q}$). *The finitely generated subgroups of $\mathbb{Q}$ have the form $q\mathbb{Z}$ (noted $\langle q \rangle$) where $q \in \mathbb{Q}$.*

THEOREM & DEFINITION 6 (RATIONAL ARITHMETICAL COSETS). *The join of $\{\emptyset, \mathbb{Q}\}$ and of the cosets $p + q\mathbb{Z}$ (noted $p\langle q \rangle$ where p and q are rational numbers) of $\mathbb{Q}$ modulo finitely generated subgroups is a Moore family of $\mathbb{P}(\mathbb{Q})$; it is called the* lattice of rational arithmetical cosets.

Before we specify the operations on this lattice, we extend the arithmetical operators to rational numbers. Based on the divisibility notion stating that given two rational numbers p and q, p is a divisor of q if and only if there exists an integer k such that $q = kp$ the following extensions of the arithmetical operators hold.

DEFINITION 7 (ARITHMETICAL OPERATORS EXTENSIONS). The euclidean division, the modulo, the greatest common divisor and the least common multiple are defined by

$$\begin{array}{lllll} \mathrm{div} & : & \mathbb{Q} \times \mathbb{Q}^{+*} & \rightarrow & \mathbb{Z} \\ & & (\frac{a}{b}, \frac{c}{d}) & \mapsto & \mathrm{sgn}(bc)ad \ \mathrm{div}\ |bc| \\ \mathrm{gcd} & : & \mathbb{Q}^{+} \times \mathbb{Q}^{+} & \rightarrow & \mathbb{Q}^{+} \\ & & (\frac{a}{b}, \frac{c}{d}) & \mapsto & \frac{\mathrm{gcd}(ad,bc)}{bd} \end{array} \qquad \begin{array}{lllll} \mathrm{mod} & : & \mathbb{Q} \times \mathbb{Q}^{+*} & \rightarrow & \mathbb{Q} \\ & & (\frac{a}{b}, \frac{c}{d}) & \mapsto & \frac{\mathrm{sgn}(bc)ad \ \mathrm{mod}\ |bc|}{|bd|} \\ \mathrm{lcm} & : & \mathbb{Q}^{+} \times \mathbb{Q}^{+} & \rightarrow & \mathbb{Q}^{+} \\ & & (\frac{a}{b}, \frac{c}{d}) & \mapsto & \frac{\mathrm{lcm}(ad,bc)}{bd} \end{array}$$

Now we characterize the operations on the complete lattice of rational arithmetical cosets: the comparison, the least upper and the greatest lower bounds.

PROPOSITION 8 (LATTICE OPERATIONS). *Let p_1, q_1, p_2 and q_2 be four rational numbers.*

$$\begin{array}{rcl} p_1\langle q_1 \rangle \subseteq p_2\langle q_2 \rangle & \Leftrightarrow & p_1 - p_2 \in \langle q_2 \rangle \wedge q_1 \in \langle q_2 \rangle \\ p_1\langle q_1 \rangle \sqcap p_2\langle q_2 \rangle \neq \emptyset & \Leftrightarrow & p_1 - p_2 \in \langle \mathrm{gcd}(q_1, q_2) \rangle \\ c \in p_1\langle q_1 \rangle \sqcap p_2\langle q_2 \rangle \neq \emptyset & \Rightarrow & p_1\langle q_1 \rangle \sqcap p_2\langle q_2 \rangle = c\langle \mathrm{lcm}(q_1, q_2) \rangle \\ p_1\langle q_1 \rangle \sqcup p_2\langle q_2 \rangle & = & p_1\langle \mathrm{gcd}(q_1, q_2, p_1 - p_2) \rangle \end{array}$$

The operations are extended to deal with the extremal elements.

Since the height of the lattice is very big, a widening operator is generally used in this analysis; several different ones are proposed in [Gra91a]. They are all based on the idea of a jump to $\mathbb{Q}$ in a possibly infinite increasing chain of rational cosets. The different strategies result from the predicates taken under consideration in order to do this jump to $\mathbb{Q}$. The simplest predicate is that two consecutive cosets in the increasing chain have non zero distinct modulos.

2. Rational linear congruence analysis

When E is an element of $\{\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}\}$, E^t is the set of tuples of elements of E and $E^{n,p}$ is the set of matrices of elements of E with n rows and p columns. The notation of $M^{i,j}$ as the matrix corresponding to the columns of M of ranks greater than i and less than j is used in the following, when i is 1 it will be omitted giving M^j. M_i denotes the column of rank i of the matrix M. M_i possibly denotes a matrix too, the context indicates which semantics is chosen. The "." operator is used to denote either the product of one scalar with a tuple, or the scalar product of two tuples. The vector named O denotes the null vector and the matrix $I(d)$ the identity matrix of dimension d; it is simply noted I if there is no ambiguity on d.

Following the same approach as in the preceding section, first a set of subgroups of $\mathbb{Q}^n$ is characterized, then the set of the corresponding cosets is exhibited.

Theorem & Definition 9 (Linear subgroup of $\mathbb{Q}^n$ [Gra91a]). *Let p and r be two non negative integers and $M \in \mathbb{Q}^{n,p+r}$ a rational coefficients matrix.* A linear subgroup $\langle M\rangle_{(p,r)}$ *of $\mathbb{Q}^n$ is the set $M^p\mathbb{Z}^p + M^{p+1,p+r}\mathbb{Q}^r$ of the elements*

$$k_1.M_1 + k_2.M_2 + \cdots + k_p.M_p + \alpha_1.M_{p+1} + \alpha_2.M_{p+2} + \cdots + \alpha_r.M_{p+r}$$

where $(k_i)_{i\in[1,p]} \in \mathbb{Z}^p$ and $(\alpha_i)_{i\in[1,r]} \in \mathbb{Q}^r$. It is the sum of a finitely generated subgroup $M^p\mathbb{Z}^p$ of $\mathbb{Q}^n$ and of a subspace $M^{p+1,p+r}\mathbb{Q}^r$ of $\mathbb{Q}^n$.
If $p + r = 0$ then the convention is that the corresponding linear subgroup is the null vector singleton.

A linear subgroup $\langle M\rangle_{(p,r)}$ is possibly denoted using the collection of the columns of the matrix M instead of the matrix itself, giving $\langle M_1, M_2, \ldots, M_{p+r}\rangle_{(p,r)}$.

Theorem & Definition 10 (Linear cosets [Gra91a]). *A* linear coset $A\langle M\rangle_{(p,r)}$ *of $\mathbb{Q}^n$ is a coset of $\mathbb{Q}^n$ modulo a linear subgroup $\langle M\rangle_{(p,r)}$ of $\mathbb{Q}^n$; it has the form*

$$A\langle M\rangle_{(p,r)} \stackrel{\text{def}}{=} \{A + MK, K \in \mathbb{Z}^p\mathbb{Q}^r\}$$

where $A \in \mathbb{Q}^n$ is the representative, $\langle M\rangle_{(p,r)}$ ($M \in \mathbb{Q}^{n,p+r}$) the modulo, $p \in \mathbb{N}$ the integer rank and $r \in \mathbb{N}$ the rational rank of the linear coset $A\langle M\rangle_{(p,r)}$. The complete lattice of linear cosets of $\mathbb{Q}^n$ is obtained by adding the empty set.

The lattice of linear cosets is now shown to exactly correspond to the set of solution sets of rational linear congruence equation systems. The process of getting a linear coset from such an equation system is exposed, at least the part of the process that will be used in appendix D.

An example of a linear coset of $\mathbb{Q}^2$ is given on the figure II.2. It is the solution of the linear congruence equation $x - 2y \equiv 2 \mod (6)$ corresponding to the linear coset

$$\left(\begin{smallmatrix} 2 \\ 0 \end{smallmatrix}\right)\left\langle\begin{smallmatrix} 0 & 2 \\ 3 & 1 \end{smallmatrix}\right\rangle_{(1,1)}$$

First, a method for finding the coset of $\mathbb{Z}^n$ which is the solution of a linear congruence equation in $\mathbb{Z}^p$ is needed; the complete method is given in [Gra91a]; it is too long to be recalled here although it is needed in the implementation of our analyses. Then the resolution of a linear

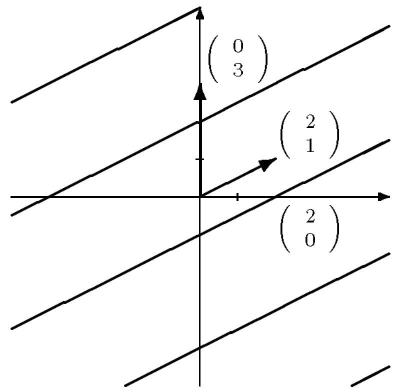


FIGURE II.2. Rational linear congruence equation solution set.

---

congruence equation in $\mathbb{Z}^p\mathbb{Q}^r$ with $r \neq 0$ is given in propositions 11 and 12.

PROPOSITION 11 (LINEAR EQUATION IN $\mathbb{Z}^p\mathbb{Q}^r$ [Gra91a]). *Let λ_{p+r} be a non zero rational number. The solution set of the linear equation*

$$\lambda_1 x_1 + \lambda_2 x_2 + \ldots + \lambda_{p+r} x_{p+r} \quad = \quad a$$

in the linear coset $O\langle I\rangle_{(p,r)}$ with $0 \leq p \leq p+r-1$ is the linear coset

$$C \quad = \quad \frac{a}{\lambda_{p+r}} I_{p+r} \left\langle I_1 - \frac{\lambda_1}{\lambda_{p+r}} I_{p+r}, \ldots, I_{p+r-1} - \frac{\lambda_{p+r-1}}{\lambda_{p+r}} I_{p+r} \right\rangle_{(p,r-1)}$$

The columns of the modulo of C are linearly independent.

PROPOSITION 12 (LINEAR CONGRUENCE EQUATION IN $\mathbb{Z}^p\mathbb{Q}^r$ [Gra91a]). *Let q and λ_{p+1} be non zero rational numbers. The solution set of the linear congruence equation*

$$\lambda_1 x_1 + \lambda_2 x_2 + \ldots + \lambda_{p+1} x_{p+1} + \ldots + \lambda_{p+r} x_{p+r} \quad \equiv \quad a \mod (q)$$

in the linear coset $O\langle I\rangle_{(p,r)}$ with $0 \leq p \leq p+r-1$ is the linear coset

$$\begin{aligned} C \quad = \quad \frac{a}{\lambda_{p+1}} I_{p+1} \Bigg\langle I_1 - \frac{\lambda_1}{\lambda_{p+1}} I_{p+1}, \ldots, I_p - \frac{\lambda_p}{\lambda_{p+1}} I_{p+1}, \frac{|q|}{\lambda_{p+1}} I_{p+1}, \\ I_{p+2} - \frac{\lambda_{p+2}}{\lambda_{p+1}} I_{p+1}, \ldots, I_{p+r} - \frac{\lambda_{p+r}}{\lambda_{p+1}} I_{p+1} \Bigg\rangle_{(p+1,r-1)} \end{aligned}$$

The columns of the modulo of C are linearly independent.

Then a method that reduces the resolution of a linear congruence equation in a linear coset to the resolution of another linear congruence equation in a special kind of linear cosets of the form $\mathbb{Z}^p\mathbb{Q}^r$ is given (of which only a special case is explicated here because the other cases are not used by our algorithm).

PROPOSITION 13 (LINEAR CONGRUENCE EQUATION IN A COSET OF $\mathbb{Z}^n$ [Gra91a]). *The solution of the linear congruence equation*

$$\lambda_1 x_1 + \lambda_2 x_2 + \cdots + \lambda_n x_n \equiv a \mod (q) \tag{2}$$

in the linear coset $A\langle M\rangle_{(p,0)}$ *is the coset*

$$(A + MB)\langle MN\rangle_{(p',0)}$$

where $B\langle N\rangle_{(p',0)}$ *is the solution of the linear congruence equation*

$$(\lambda_1, \lambda_2, \ldots, \lambda_n)M \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{pmatrix} \equiv (a - (\lambda_1, \lambda_2, \ldots, \lambda_n).A) \mod (q) \tag{3}$$

in $\mathbb{Z}^p$*, if the equation (3) has a non empty solution set. Otherwise, the solution set of equation (2) is empty.*

Finally, the solution of a linear congruence equation system in $\mathbb{Q}^n$ is obtained iteratively, solving first an equation in $\mathbb{Q}^n$ and then each other equation in the linear coset resulting from the preceding resolution.

THEOREM 14 (LINEAR COSET REPRESENTATIONS EQUIVALENCE [Gra91a]). *The set of solution sets in* $\mathbb{Q}^n$ *of linear congruence equation systems coincides with the set of linear cosets of* $\mathbb{Q}^n$*.*

The operators on the lattice of linear cosets (least upper bound, greatest lower bound and comparison) are not used in the following and hence not detailed here; only operators concerning linear subgroups comparison are given.

PROPOSITION 15 (LINEAR SUBGROUP COMPARISON [Gra91a]). *Let* $\langle M\rangle_{(p,r)}$ *and* $\langle M'\rangle_{(p',r')}$ *be two linear subgroups of* $\mathbb{Q}^n$*.*

$$\langle M\rangle_{(p,r)} \subseteq \langle M'\rangle_{(p',r')} \quad \Leftrightarrow \quad \left\{ \begin{array}{rcl} M^p\mathbb{Z}^p & \subseteq & M'\mathbb{Z}^{p'}\mathbb{Q}^{r'} \\ M^{p+1,p+r}\mathbb{Q}^r & \subseteq & M'^{p'+1,p'+r'}\mathbb{Q}^{r'} \end{array} \right.$$

$\langle M'\rangle_{(p',r')}$ *is said to divide* $\langle M\rangle_{(p,r)}$*.*

The greatest common divisor of two linear subgroups always exists.

For more details about congruence analysis, see the work of Granger in [Gra89, Gra90, Gra91b]. We end this chapter with some examples illustrating both analyses we sketched above.

Suppose that the program

```
        for i := 1 to n do
{1:}       z := i + 1/(2*i);
{2:}       x := x + z;
{3:}       y := y - 2*z;
{4:}    od;
```

is analyzed, using the lattice of linear cosets, with the initial abstract context

$$x \equiv 0 \mod (\frac{1}{10}) \;\wedge\; y \equiv 0 \mod (\frac{1}{10})$$

After five iterations, it is automatically discovered that at program points {1:}, {2:} and {4:} the program variables satisfy

$$\left\{ \begin{array}{rcl} i & \equiv & 0 \bmod (1) \\ 2x + y & \equiv & 0 \bmod (\frac{1}{10}) \end{array} \right.$$

Using the lattice of rational arithmetical congruences, it is automatically found that in the program

```
        x := 2.8542
{1:}    while condition do
{2:}       x := x + 1/500;
{3:}    od;
{4:}
```

the program variable x verifies

$$x \equiv \frac{1}{5000} \mod (\frac{1}{500})$$

Part 2

SEMANTIC ANALYSIS OF RATIONAL INTERVAL CONGRUENCES

CHAPTER III

DESIGN OF INTEGER AND RATIONAL MODELS

The analysis of interval congruences requires two different domains, a first one of integer properties for a matter of precision and a second one of rational properties for the efficiency of its basic algorithms. Although the coset congruence domain is presented before the interval congruence one, we see in Chapter IV that the integer coset congruences are naturally deduced from the rational interval congruences. The content of this chapter and the next one corresponds to [Mas93].

1. Notations

The notations of Chapter II are used. In addition, we have $\mathbb{Q}_{-\infty} \stackrel{\text{def}}{=} \mathbb{Q} \cup \{-\infty\}$, $\mathbb{Q}_{+\infty} \stackrel{\text{def}}{=} \mathbb{Q} \cup \{+\infty\}$, and $\mathbb{Z}_{-\infty} \stackrel{\text{def}}{=} \mathbb{Z} \cup \{-\infty\}$, $\mathbb{Z}_{+\infty} \stackrel{\text{def}}{=} \mathbb{Z} \cup \{+\infty\}$ where $-\infty$ and $+\infty$ are considered as limits on $\mathbb{Q}$ and $\mathbb{Z}$. The usual operators (sum, product,...) on $\mathbb{Q}$ and $\mathbb{Z}$ are canonically extended to $\mathbb{Q}_{-\infty}$, $\mathbb{Q}_{+\infty}$, $\mathbb{Z}_{-\infty}$, $\mathbb{Z}_{+\infty}$ and $\lceil -\infty \rceil = \lfloor -\infty \rfloor = -\infty$ and $\lceil +\infty \rceil = \lfloor +\infty \rfloor = +\infty$. Following the context $[-\infty, +\infty]$ is $\mathbb{Q}_{-\infty} \cup \mathbb{Q}_{+\infty}$ or $\mathbb{Z}_{-\infty} \cup \mathbb{Z}_{+\infty}$. The greatest common divisor is always non negative. The integer coset $a\,\langle q \rangle$ with integer representative a and modulo q is the set $\{a + kq,\ k \in \mathbb{Z}\}$. The rational coset $a\,\langle q \rangle$ corresponds to the set $\{a + kq,\ k \in \mathbb{Z}\}$ where $(a, q) \in \mathbb{Q}^2$. The relation $l \equiv u \mod m$, which is equivalent to $u - l \in \langle m \rangle$, is shortened to $l \stackrel{m}{=} u$. An inverse of the integer θ with respect to the integer m, when it exists, is noted $\theta^{-1}_{(m)}$ and satisfies $\theta\theta^{-1} \in 1\,\langle m \rangle$. $\theta^{-1}_{(m)}$ is noted θ^{-1} when there is no ambiguity on the modulo. An inverse of θ with respect to m exists when $\gcd(\theta, m) = 1$; it is a direct consequence of Bezout's theorem[1]. For the rest of this chapter, the convention is that an inverse $\theta^{-1}_{(m)}$ of an offset θ is always taken with respect to the modulo m of their coset congruence, if there is no possible ambiguity (see definition 16 for the definitions of coset congruence and offset).

[1] Let a and b be two integers; there exist integers u and v such that

$$u.a + v.b = \gcd(a, b)$$

2. The set CC of coset congruences on $\mathbb{Z}$

Interval analysis of [CC76] and congruence analysis of [Gra89, Gra91b] are quite orthogonal concepts. This leads to the definition of a third analysis with the basic idea of generalizing the first two to the notion of coset congruence. The basic components of coset congruences (and two degenerate cases of the general definition) are integer intervals and integer cosets. To fill the gap between these two kinds of elements, general coset congruences are introduced. A coset congruence is a set of arithmetical cosets with the same modulo and whose representatives are separated by an offset such that the common modulo and the offset are prime numbers.

2.1. Definition.

Definition 16 (Coset congruence $\theta.[l,u]\langle m\rangle$). Let $l \in \mathbb{Z}_{-\infty}$, $u \in \mathbb{Z}_{+\infty}$ and $m, \theta \in \mathbb{Z}$ be integers such that $\gcd(\theta, m) = 1$ and $m = 0$ implies $\theta = 1$. The *coset congruence* $\theta.[l,u]\langle m\rangle$ of offset θ, lower bound l, upper bound u and modulo m is defined by

$$\theta.[l,u]\langle m\rangle \stackrel{\text{def}}{=} \begin{cases} [-\infty, u] \cup [l, +\infty] & \text{if } l > u \text{ and } m = 0, \quad (4)\\ \bigcup\limits_{l \le \kappa \le u} \kappa\theta\langle m\rangle & \text{otherwise.} \quad (5) \end{cases}$$

CC is the set of coset congruences.

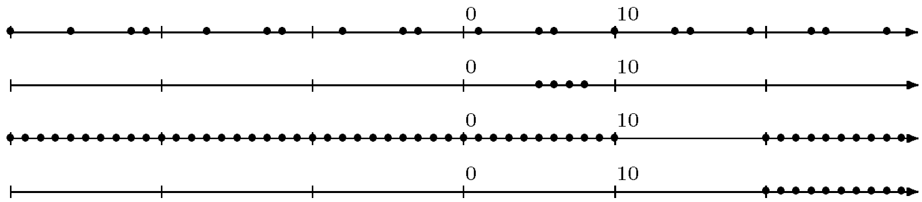
A very important remark for the rest of the discussion about coset congruences is that, when the modulo is non zero, since the offset and modulo are prime numbers by definition, the single cosets $\kappa\theta\langle m\rangle$ used in definition case (5) are all distinct for m consecutive values of κ. Hence taking a sufficiently wide interval $[l, u]$ provides a way to represent $\mathbb{Z}$ (see lemma 17 for details).

One motivation to define such a surprising integer model is that a coset congruence is exactly the intersection with $\mathbb{Z}$ of a much more intuitive model defined on the set of rational numbers: the interval congruences that are defined in section 3. In particular, we will see that the primality between the common modulo and the offset separating the different representatives comes from that intersection. Another, more practical, reason that leads us to approximate CC with rational interval congruences is that the comparison (set inclusion) test on CC is for the moment particularly inefficient[2].

The coset congruences of offset equal to one intuitively correspond to usual integer intervals regularly dispersed following a pattern of length the value of the modulo. The different other kinds of integer sets considered in the preceding definition are illustrated by the following figure:

[2] No efficient (constant time) algorithm has been found by me

(6)
(7)
(8)
(9)



The general case (6), corresponding to $5.[1,3]\langle 9\rangle$, is the set of the three integer cosets $5\langle 9\rangle$, $10\langle 9\rangle = 1\langle 9\rangle$ and $15\langle 9\rangle = 6\langle 9\rangle$. The case (7), where the modulo is zero and the representative bounds well ordered, is noted $1.[5,8]\langle 0\rangle$ and corresponds to the integer interval $[5,8]$. The case (8), where the modulo is zero and the representative bounds inverse ordered, corresponds to definition case (4) and is noted $1.[20,10]\langle 0\rangle$. Finally, the case (9) represents the set of integers greater than 20 and is the coset congruence $1.[20,+\infty]\langle 0\rangle$. Following these four coset congruence schemes, we see that the representation of $\mathbb{Z}$ using such a model is possible in the case (6) when there is enough distinct cosets, in cases (7) and (9) when the single integer interval is $[-\infty,+\infty]$ and in the case (8) when the lower bound is the successor of the upper bound.

The characterization of coset congruences equal to $\mathbb{Z}$ or to $\emptyset$ is described by lemmas 17 and 18.

LEMMA 17 (COSET CONGRUENCE EQUAL TO $\mathbb{Z}$). *Let θ, m and c be three integers, $l \in \mathbb{Z}_{-\infty}$ and $u \in \mathbb{Z}_{+\infty}$, then*

$$0 < |m| \leq u - l + 1 \quad \Leftrightarrow \quad \mathbb{Z} = \begin{cases} 1.[c, c-1]\langle 0\rangle \\ 1.[-\infty,+\infty]\langle 0\rangle \\ \theta.[l,u]\langle m\rangle \end{cases}$$

PROOF. Clearly the case (4) of coset congruence definition leads to $\mathbb{Z}$ if and only if $u = l-1$. Two subcases based on the nullity of its modulo must be considered for the case (5).
First if the modulo is zero[3] then the offset is one by definition and the corresponding coset congruence is $\mathbb{Z}$ if and only if $u = -l = +\infty$.
Otherwise the modulo is non zero and then the corresponding coset congruence is $\mathbb{Z}$ if and only if the number of its constituent distinct integer cosets is greater than the absolute value of its modulo. Notice that since $\gcd(\theta, m) = 1$, for m consecutive values of i, all the cosets $i\theta\langle m\rangle$ are distinct and the result follows. $\square$

LEMMA 18 (COSET CONGRUENCE EQUAL TO $\emptyset$). *Let θ and m be integers, $l \in \mathbb{Z}_{-\infty}$ and $u \in \mathbb{Z}_{+\infty}$, then*

$$m \neq 0 \wedge u < l \quad \Leftrightarrow \quad \theta.[l,u]\langle m\rangle = \emptyset$$

[3]This case exactly corresponds to the usual integer, possibly infinite, intervals

PROOF. The first case (4) of the coset congruence definition never provides the empty set. The latter case (5) leads to the same conclusion when $m = 0$ but when $m \neq 0$ the empty set is obtained for $u < l$. $\square$

Remark that the nullity of the modulo implies that the offset is one in the definition of coset congruences.

2.2. Equivalence Relation. For the relation $\subseteq$ induced by the set inclusion relation, CC is a preorder. An equivalence relation $\approx\ =\ \subseteq \wedge \supseteq$ is defined to build a partial order on the quotient set $CC/_{\approx}$ (for example $2.[7,9]\langle -11\rangle \approx 9.[2,4]\langle 11\rangle$).

A characterization of coset congruence equivalence is now given. This algorithm determines if two coset congruences represent the same integer set and is used to implement the relation $\approx$; it is proven correct in appendix A.

THEOREM 19 (COSET CONGRUENCE EQUIVALENCE $\approx$). *Let* $C_1 = \theta_1.[l_1, u_1]\langle m_1\rangle$ *and* $C_2 = \theta_2.[l_2, u_2]\langle m_2\rangle$ *be two coset congruences.* $C_1 \approx C_2$ *if and only if*

$$\left.\begin{array}{c} 0 < |m_1| \le u_1 - l_1 + 1 \\ \vee \\ u_1 = l_1 - 1 \ \wedge\ m_1 = 0 \\ \vee \\ u_1 = -l_1 = +\infty \wedge m_1 = 0 \end{array}\right\} \wedge \left\{\begin{array}{c} 0 < |m_2| \le u_2 - l_2 + 1 \\ \vee \\ u_2 = l_2 - 1 \ \wedge\ m_2 = 0 \\ \vee \\ u_2 = -l_2 = +\infty \wedge m_2 = 0 \end{array}\right. \tag{10}$$

$$\vee$$

$$m_1 \neq 0 \ \wedge\ u_1 < l_1 \ \wedge\ m_2 \neq 0 \ \wedge\ u_2 < l_2 \tag{11}$$

$$\vee$$

$$|m_1| = |m_2| \ \wedge\ u_1 - l_1 = u_2 - l_2 \ \wedge \left\{\begin{array}{l} \left\{\begin{array}{c} w = 1 \ \vee m = 0 \\ \theta_1 l_1 \overset{m}{=} \theta_2 l_2 \end{array}\right. \\ \vee \\ \left\{\begin{array}{c} w = m - 1 \\ \theta_1 l_1 - \theta_2 l_2 \overset{m}{=} \theta_1 - \theta_2 \end{array}\right. \\ \vee \\ \left\{\begin{array}{c} 2 \le w \le m - 2 \\ \theta_1 \overset{m}{=} -\theta_2 \\ \theta_1 l_1 \overset{m}{=} \theta_2 u_2 \end{array}\right. \\ \vee \\ \left\{\begin{array}{c} 2 \le w \le m - 2 \\ \theta_1 \overset{m}{=} \theta_2 \\ \theta_1 l_1 \overset{m}{=} \theta_2 l_2 \end{array}\right. \end{array}\right. \tag{12}$$

where $m = |m_1|$ *and* $w = u_1 - l_1 + 1$.

These three cases respectively correspond to $C_1 \approx C_2 = \mathbb{Z}$, $C_1 \approx C_2 = \emptyset$ and to the general case. The redundancies figuring in the above formula are not eliminated for a matter of formulation simplicity.

The quotient set $CC/_{\approx}$ is abusively called the set of coset congruences.

2.3. Normalization. If we arbitrary choose a representation of the empty set $(1.[1,0]\langle 1\rangle)$ and of the set of integers $(1.[0,0]\langle 1\rangle)$ by a coset congruence, if we remark that apart from them, coset congruences of zero modulo are equivalence classes with only one element, and finally if we consider the coset congruences with positive modulo, offset and lower bound positive and smaller than the modulo (recall that for example $2.[7,9]\langle -11\rangle \approx 9.[2,4]\langle 11\rangle$), we obtain the following normalization algorithm.

COROLLARY 20 (COSET CONGRUENCE NORMALIZATION $\|\ \|$). *Let $I = \theta.[l,u]\langle m\rangle$ be a coset congruence, $\|I\|$ is defined by*

$$
\begin{array}{lll}
\textit{if } \mathbb{Z} \subseteq I & \textit{then} & 1.[0,0]\langle 1\rangle \\
\textit{else if } I \subseteq \emptyset & \textit{then} & 1.[1,0]\langle 1\rangle \\
\textit{else if } m = 0 & \textit{then} & 1.[l,u]\langle 0\rangle \\
\textit{else if } u = l & \textit{then} & \zeta(1.[l\theta, u\theta]\langle m\rangle) \\
\textit{else if } u - l = m - 2 & \textit{then} & \zeta(1.[l\theta - \theta + 1, l\theta - \theta + m - 1]\langle m\rangle) \\
\textit{else if } |m| \text{ div } 2 < \theta^{-1} \bmod |m| < |m| & \textit{then} & \zeta(-\theta.[-u,-l]\langle m\rangle) \\
\textit{else} & & \zeta(\theta.[l,u]\langle m\rangle)
\end{array}
$$

where $\zeta(\theta.[l,u]\langle m\rangle) = (\theta \bmod |m|).[l \bmod |m|, u - (l \text{ div } |m|)|m|]\langle |m|\rangle$ is compatible with the equivalence relation $\approx$ and is a normalization operator on $CC/_{\approx}$.

PROOF. The equivalence between $C \in CC$ and $\|C\|$ comes from theorem 19 and the normalization character ($\forall C_1, C_2 \in CC\ C_1 \approx C_2 \Leftrightarrow \|C_1\| = \|C_2\|$) is provided by theorem 19 too[4] (successively considering all the cases where two coset congruences are equivalent and choosing one representation). □

The normalization of elements representing $\mathbb{Z}$ and $\emptyset$ is provided in order to simplify the expressions in the rest of the work; there is no canonical representation for these elements; for example $0.[0,0]\langle 1\rangle$ could represent $\mathbb{Z}$ as well. The choice between the offset and its opposite comes from the consideration of the abstraction function, see section IV.1.2. Since most of the operators that are defined on $CC/_{\approx}$ are not compatible[5] with the equivalence relation $\approx$, we cannot denote an equivalence class by one of its representatives and have to use the normalization operator on $CC/_{\approx}$ defined in corollary 20.

Note that this normalization algorithm could have been used as a concretization application from CC into $\mathbb{P}(\mathbb{Z})$ since it gives the unique subset of $\mathbb{Z}$ represented by the original coset congruence. It is not the case because the concretization giving the meaning of an interval congruence is used instead (see the construction of the abstract interpretation in Chapter IV).

[4]The unicity of the choice between an offset and its opposite is a consequence of the following equivalence for θ and θ' prime with m

$$\theta + \theta' \stackrel{m}{=} 0 \Leftrightarrow \theta^{-1} + \theta'^{-1} \stackrel{m}{=} 0$$

ensuring that our normalization algorithm is idempotent.

[5]Operator o is compatible with relation $\approx$ if and only if $\forall C_1, C_2, C_1', C_2' \in CC \quad (C_1 o C_2) \wedge (C_1 \approx C_1') \wedge (C_2 \approx C_2') \Rightarrow (C_1' o C_2')$

The next lemma is used to define the concretization function in the relational analysis in section VI.1.3.

LEMMA 21 (INTERSECTION WITH AN ARITHMETICAL COSET). *Let $\theta.[l,u]\langle m\rangle \in CC/_{\approx}$ be a normalized coset congruence such that $m \neq 0$ or $l \leq u$, and g a non negative divisor of its modulo m.*

$$\theta.[l,u]\langle m\rangle \cap \langle g\rangle \quad = \quad \begin{cases} 1.[1,0]\langle 1\rangle & \text{if } \left\lceil \frac{l}{g}\right\rceil > \left\lfloor \frac{u}{g}\right\rfloor \\ g * \left(\theta.\left[\left\lceil \frac{l}{g}\right\rceil, \left\lfloor \frac{u}{g}\right\rfloor\right]\left\langle \frac{m}{g}\right\rangle\right) & \text{otherwise} \end{cases}$$

PROOF.

$$\begin{aligned}\theta.[l,u]\langle m\rangle \cap \langle g\rangle &= \left(\bigcup_{l\leq\kappa\leq u} \kappa\theta\langle m\rangle\right) \cap \langle g\rangle \\ &= \bigcup_{l\leq\kappa\leq u} \left((\kappa\theta\langle m\rangle) \cap \langle g\rangle\right)\end{aligned}$$

since $\gcd(g,\theta) = 1$

$$\begin{aligned} &= \bigcup_{g\left\lceil \frac{l}{g}\right\rceil \leq \kappa \leq g\left\lfloor \frac{u}{g}\right\rfloor \ \wedge\ \kappa \overset{g}{=} 0} \kappa\theta\langle m\rangle \\ &= \bigcup_{\left\lceil \frac{l}{g}\right\rceil \leq \kappa' \leq \left\lfloor \frac{u}{g}\right\rfloor} \kappa' g\theta\langle m\rangle \end{aligned}$$

factorizing g in the cosets

$$= g * \left(\bigcup_{\left\lceil \frac{l}{g}\right\rceil \leq \kappa' \leq \left\lfloor \frac{u}{g}\right\rfloor} \kappa'\theta\left\langle \frac{m}{g}\right\rangle\right)$$

which is decomposed according to its emptyness and provides the result. □

2.4. Complementation operator. Let us define two auxiliary functions giving respectively the successor of a possibly positive infinite integer and the predecessor of a possibly negative infinite integer. Their definitions result from the simplification of the complementary operator definition.

$$\begin{aligned} \sigma \ :\ \mathbb{Z}_{+\infty} &\to \mathbb{Z}_{-\infty} \\ n &\mapsto \begin{cases} -\infty & \text{if } n = +\infty \\ n+1 & \text{otherwise} \end{cases} \end{aligned}$$

$$\begin{aligned} \pi \ :\ \mathbb{Z}_{-\infty} &\to \mathbb{Z}_{+\infty} \\ n &\mapsto \begin{cases} +\infty & \text{if } n = -\infty \\ n-1 & \text{otherwise} \end{cases} \end{aligned}$$

Now the complementary operator which provides the negation of a coset congruence property is defined.

THEOREM & DEFINITION 22 (COMPLEMENTATION $\overline{}$). *Let* $C = \theta.[l,u]\langle m\rangle$ *be a normalized coset congruence. Its* complementary *in* $\mathbb{Z}$ *is*

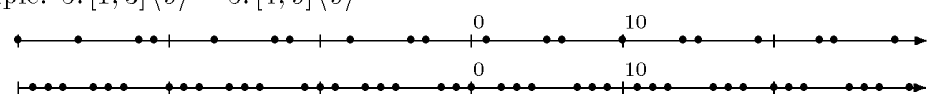
$$\overline{C} = \theta.[\sigma(u), \pi(l+m)]\langle m\rangle$$

and verifies:

$$\begin{array}{rcl} \overline{C} \cap C & = & \emptyset \\ \overline{C} \cup C & = & \mathbb{Z} \end{array}$$

PROOF. The reader can easily establish the correctness of the complementation formula in the case where the modulo is zero or the lower bound greater than the upper one.
Now if the modulo is not zero and $l \leq u$, there are exactly m distinct cosets of modulo m and they are reached if we consider the m cosets of representatives $\theta l, \theta(l+1), \ldots, \theta u, \theta(u+1), \ldots, \theta(l+m-1)$ since θ and m are prime. Hence the coset congruences of constitutive representatives $\theta l, \theta(l+1), \ldots, \theta u$ and $\theta(u+1), \ldots, \theta(l+m-1)$ are complementary of each other[6]. $\square$

For example: $\overline{5.[1,3]\langle 9\rangle} = 5.[4,9]\langle 9\rangle$



and

$$\begin{array}{rcl} \overline{1.[5,9]\langle 0\rangle} & = & 1.[10,4]\langle 0\rangle \\ \overline{\overline{5.[1,3]\langle 9\rangle}} & = & 5.[10,12]\langle 9\rangle \approx 5.[1,3]\langle 9\rangle \\ \overline{\mathbb{Z}} = \overline{1.[0,0]\langle 1\rangle} & = & 1.[1,0]\langle 1\rangle = \emptyset \\ \overline{1.[5,+\infty]\langle 0\rangle} & = & 1.[-\infty,4]\langle 0\rangle \end{array}$$

Only few analyses like parity, sign or logic program groundness analysis [CC92a] provide such a complementation characteristic and it will be shown to be very useful in the section IV.3 on abstract primitives. Although such a property necessitates the consideration of complement of finite integer intervals and hence complicates the expressions concerning coset congruences, such a characteristic feature is kept for analysis accuracy motives.

The use of normalized coset congruences leads to simpler expressions than if we had to generalize the complementation operator to CC. The complementary of a coset congruence corresponds to the set of integer cosets not contained in the original one, hence only the representative has to be inverted; the resulting expression is not always normalized (see the examples below) although the property $\forall C \in CC/_{\approx}\ \overline{\overline{C}} \approx C$ holds.

[6]Recall that for a normalized coset congruence, the difference between its greater and lower bounds is less than its modulo.

2.5. Set inclusion induced order. Comparison on CC is not provided for the general case because no constant time algorithm had been found by us; instead, only a special case where one operand is an arithmetical coset is dealt with.

PROPOSITION 23 (PARTIAL-ORDER ON COSET CONGRUENCES). *Let* $C_1 = 1.[l_1, l_1]\langle m_1\rangle$ *and* $C_2 = \theta_2.[l_2, u_2]\langle m_2\rangle$ *be two normalized coset congruences non empty and non equal to* $\mathbb{Z}$ *such that* $m_1 m_2 \neq 0$. $C_1 \subseteq C_2$ *if and only if*

$$\left\lfloor \frac{u_2 - \theta_2^{-1} l_1}{\gcd(m_1, m_2)} \right\rfloor = \left\lceil \frac{l_2 - \theta_2^{-1} l_1}{\gcd(m_1, m_2)} \right\rceil + \frac{m_2}{\gcd(m_1, m_2)} - 1 \tag{13}$$

PROOF. Let $d = \gcd(m_1, m_2)$ and $q_2 = \frac{m_2}{d}$.
From the proof of lemma 36 we know that $C_1 \subseteq C_2$ is equivalent to

$$l_1\langle d\rangle \subseteq C_2$$

which is the same as

$$(l_1 + d)\langle m_2\rangle \;\cup\; (l_1 + 2d)\langle m_2\rangle \;\cup\; \dots \;\cup\; (l_1 + q_2 d)\langle m_2\rangle \subseteq C_2$$

and multiplying all its representatives and the ones of C_2 by θ_2^{-1} (such that $\theta_2\theta_2^{-1} \overset{m_2}{=} 1$), we get the equivalent inclusion:

$$\theta_2^{-1}(l_1 + d)\langle m_2\rangle \;\cup\; \theta_2^{-1}(l_1 + 2d)\langle m_2\rangle \;\cup\; \dots \;\cup\; \theta_2^{-1}(l_1 + q_2 d)\langle m_2\rangle \subseteq 1.[l_2, u_2]\langle m_2\rangle$$

By identifying identical integer cosets, there is a one to one mapping from the integer coset set $\{d\langle m_2\rangle, \dots, q_2 d\langle m_2\rangle\}$ onto $\{\theta_2^{-1} d\langle m_2\rangle, \dots, \theta_2^{-1} q_2 d\langle m_2\rangle\}$, indeed, since $\gcd(\theta_2^{-1}, m_2) = 1$, these two coset sets are equal to the set of all cosets of modulo m_2 and of representative a multiple of d. Hence permuting the left hand side representatives we get

$$(\theta_2^{-1} l_1 + d)\langle m_2\rangle \;\cup\; (\theta_2^{-1} l_1 + 2d)\langle m_2\rangle \;\cup\; \dots \;\cup\; (\theta_2^{-1} l_1 + q_2 d)\langle m_2\rangle \subseteq 1.[l_2, u_2]\langle m_2\rangle$$

which is equivalent to

$$\theta_2^{-1} l_1\langle d\rangle \subseteq 1.[l_2, u_2]\langle m_2\rangle$$

Then the problem amounts to characterizing that q_2 consecutive representatives of the integer coset $\theta_2^{-1} l_1\langle d\rangle$ are in the interval $[l_2, u_2]$. This is equivalent to the existence of an integer i such that

$$\begin{cases} \theta_2^{-1} l_1 + (i-1)d & < & l_2 & \leq & \theta_2^{-1} l_1 + id \\ \theta_2^{-1} l_1 + (i + q_2 - 1)d & \leq & u_2 & < & \theta_2^{-1} l_1 + (i + q_2)d \end{cases}$$

that is equivalent to

$$\begin{cases} i - 1 < \frac{l_2 - \theta_2^{-1} l_1}{d} \leq i \\ \\ i \leq \frac{u_2 - \theta_2^{-1} l_1}{d} - q_2 + 1 < i + 1 \end{cases}$$

and finally

$$\left\lfloor \frac{u_2 - \theta_2^{-1} l_1}{d} \right\rfloor = \left\lceil \frac{l_2 - \theta_2^{-1} l_1}{d} \right\rceil + q_2 - 1$$

□

Special inclusion cases where coset congruences are empty, equal to $\mathbb{Z}$ or of zero modulo are very easy to deal with. Hence the present proposition provides a characterization of the coset congruence inclusion in the particular case where the smallest one is a simple coset. Since I have not been able to establish a simple property concerning general coset congruences inclusion, a new distinct order is introduced to model the precision on the coset congruences set. It is the goal of the next section. Of course, the naive algorithm consisting of using $u - l$ times (when it is finite, the other cases take constant time to deal with) the preceding algorithm to test the inclusion of $\theta.\,[l, u]\,\langle m \rangle$ in an other coset congruence is possible but very expensive (except if in practice the lower and upper bounds are very close).

2.6. Precision concrete order. Because we are not able to efficiently compare coset congruences and, moreover, for the purpose of the approximate join operator (in section IV.2.2), we need to choose between non comparable ones, a measure of accuracy ι is defined. It partially orders CC using an approximation of the cardinal of the integer set where this size is close to the probability that an integer is in the coset congruence. This is an arbitrary order defined on $CC/_{\approx}$; it is used by the approximate join operator to make an arbitrary choice between two rational interval congruences based on the ratio of information (their corresponding coset congruence) they are associated to. Note, however, that this process ensures that the approximate join of two integer intervals is the least upper bound in the lattice of intervals and that the approximate join of two integer cosets is the least upper bound in the lattice of cosets.

DEFINITION 24 (ACCURACY ι). The *accuracy function* ι associates with each coset congruence a rational number in the following way:

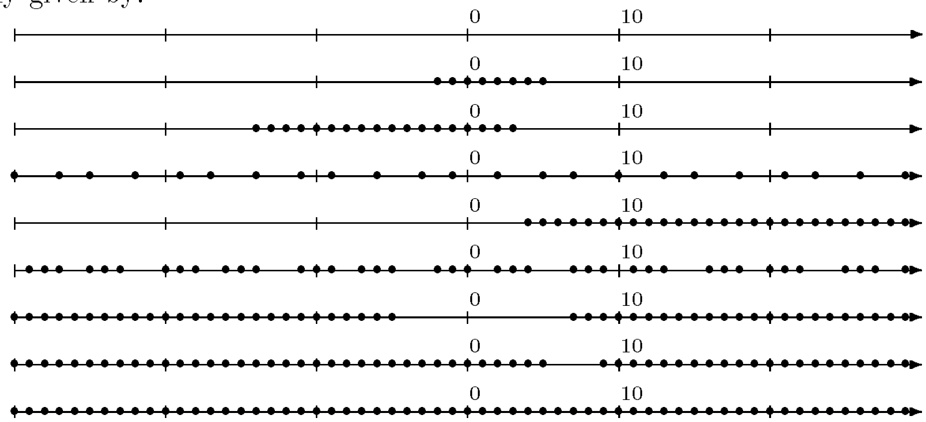
$$\begin{array}{rccl} \iota \ : & CC/_{\approx} & \to & \mathbb{Q} \\ & \theta.\,[l, u]\,\langle m \rangle & \mapsto & \begin{cases} 0 & \text{if } \theta.\,[l, u]\,\langle m \rangle = \emptyset \\ \frac{3(u-l)}{u-l+1} & \text{if } m = 0 \text{ and } -\infty < l \leq u < +\infty \\ \frac{1}{2} & \text{if } m = 0 \text{ and } (l = -\infty \text{ or } u = +\infty) \\ \frac{u-l+1}{m} & \text{if } m \neq 0 \\ 1 + \frac{1}{l-u} & \text{if } m = 0 \text{ and } u < l \\ 3 & \text{if } \theta.\,[l, u]\,\langle m \rangle = \mathbb{Z} \end{cases} \end{array}$$

Intuitively, ι arranges the coset congruences in the following informative order:

(1) the empty set;
(2) the half lines (without any ordering) in the middle of the sets of cosets with non zero modulo "density" (ratio between the number of representatives and modulo) order;
(3) the complementary of finite sets in their complementary size reverse order;

(4) the set of all integers.

and the finite sets in size order. An example of ascending chain for that partial order is graphically given by:



and corresponds to

$$\iota(1.\,[1,0]\,\langle 1\rangle) \le \iota(1.\,[-2,5]\,\langle 0\rangle) \le \iota(1.\,[-14,3]\,\langle 0\rangle) \le \iota(5.\,[1,3]\,\langle 8\rangle) \le \iota(1.\,[4,+\infty]\,\langle 0\rangle) \le \\ \iota(5.\,[4,9]\,\langle 9\rangle) \le \iota(1.\,[7,-5]\,\langle 0\rangle) \le \iota(1.\,[9,5]\,\langle 0\rangle) \le \iota(1.\,[0,0]\,\langle 1\rangle)$$

The determination of an accuracy function is not unique and has been chosen to be simple. ι could have been given without using a numerical function (for example by giving directly the comparison algorithm).

The set $CC/_{\approx}$ of coset congruences described above has only few interesting algebraic properties; it is a complete partial order with an infimum and a supremum. Its major drawback is a lack of least upper bound and of an efficient comparison algorithm between its elements. In addition $CC/_{\approx}$ is not a Moore family (see definition 1) and cannot be completed by intersecting its elements (because of the size of the resulting set). These are good motivations to introduce a new approximation, the rational sets of IC, which provide efficient algorithms.

3. The set IC of interval congruences on $\mathbb{Q}$

The goal of this section is to define a rational model based on the use of a set of rational arithmetical cosets with consecutive representatives.

3.1. Two equivalent definitions.

DEFINITION 25 (INTERVAL CONGRUENCE $[a,b]\langle q\rangle$). Let $a \in \mathbb{Q}_{-\infty}$, $b \in \mathbb{Q}_{+\infty}$ and $q \in \mathbb{Q}$ be rational numbers. The *interval congruence* $[a,b]\langle q\rangle$ of lower bound a, upper bound b, and modulo q is defined by

$$[a,b]\langle q\rangle \overset{\text{def}}{=} \begin{cases} [a,+\infty] \cup [-\infty,b] & \text{if } a > b \text{ and } q = 0 \quad (14) \\ \{x,\ \exists x_0 \in \mathbb{Q}, x = x_0 + kq,\ a \leq x_0 \leq b,\ k \in \mathbb{Z}\} & \text{otherwise} \quad (15) \end{cases}$$

IC is the set of interval congruences.

In the following, when we need to consider an interval congruence $[a,b]\langle \frac{\nu}{\delta}\rangle$, we implicitly take non negative integers ν and δ such that $\gcd(\nu,\delta) = 1$.

Dually, let us define a set of appropriate congruence equations.

DEFINITION 26 (ARCEBR). Let $a \in \mathbb{Q}_{-\infty}$, $b \in \mathbb{Q}_{+\infty}$ and $q \in \mathbb{Q}$. The *arithmetical rational congruence equation with bounded representative*

$$x \equiv [a,b]\langle q\rangle$$

is defined by the system with the rational unknown x

$$x \equiv [a,b]\langle q\rangle \overset{\text{def}}{=} \begin{cases} \displaystyle\bigvee_{\beta \geq a, \beta \leq b} x = \beta & \text{if } a > b \text{ and } q = 0 \quad (16) \\ \displaystyle\bigvee_{a \leq \beta \leq b} x \equiv \beta \mod (q) & \text{otherwise} \quad (17) \end{cases}$$

Let us note ARCEBR the set of such equations.

Clearly, IC corresponds to the solution sets of the elements of ARCEBR. For example the interval congruence $[2,5]\langle 6\rangle$ corresponds to the solution of the equation $x \equiv [2,5]\langle 6\rangle$.

THEOREM 27 (REPRESENTATION EQUIVALENCE). *The set IC of interval congruences on $\mathbb{Q}$ is the set of the solution sets of the elements of ARCEBR.*

PROOF. The natural map from ARCEBR to IC

$$\begin{array}{rccl} \mu & : & ARCEBR & \rightarrow & IC \\ & & x \equiv [a,b]\langle q\rangle & \mapsto & [a,b]\langle q\rangle \end{array}$$

provides an isomorphism between the solution sets of equations (16) and (17) and expressions (14) and (15) respectively. $\square$

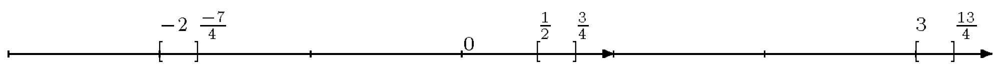
An interval congruence is either an infinitely and regularly dispersed set of rational intervals, or equivalently a set of rational cosets with "consecutive" representatives; therefore $[a, b]$ is called the representative of the interval congruence. For any non negative rational q, $IC(q)$ is the set of interval congruences of modulo q. Two interval congruences with different representatives may denote the same rational set.

Notice that the set of interval congruences contains the set of rational cosets (where the lower and upper bounds are equal) and the set of rational intervals (where the modulo is zero and the upper bound greater than the lower bound).

An example of such a rational set is given below:

$$\left[\frac{1}{2}, \frac{3}{4}\right] \left\langle \frac{5}{2} \right\rangle = \bigcup_{k \in \mathbb{Z}} \left[\frac{1+5k}{2}, \frac{3+10k}{4}\right]$$

and illustrated by



In the following, we implicitly consider the usual operators on interval congruences of zero modulo (usual rational intervals) that are the sum, the difference of two intervals and the product of an interval with a scalar.

The two following lemmas are quite simple to verify.

Lemma 28 (Interval congruence equal to $\mathbb{Q}$). *Let $I = [a, b] \langle q \rangle$ be an interval congruence.*

$$I = \mathbb{Q} \quad \Leftrightarrow \quad \begin{cases} & q = 0 \quad \wedge \quad a = -b = -\infty \\ \vee & \\ & q \neq 0 \quad \wedge \quad b - a \geq |q| \end{cases}$$

Lemma 29 (Interval congruence equal to $\emptyset$). *Let $I = [a, b] \langle q \rangle$ be an interval congruence.*

$$I = \emptyset \quad \Leftrightarrow \quad \begin{cases} q \neq 0 \\ b < a \end{cases}$$

The definition of complementation on interval congruences does not fit with the usual meaning of a complementation operator because the intersection of an element with its complementary is not empty. The notion is only used to compare interval congruences.

Definition 30 (Complementation on IC). Let $I_1 = [a_1, b_1] \langle q \rangle$ be an interval congruence. The interval congruence $I_2 = [a_2, b_2] \langle q \rangle$ is called its *complementary* iff

$$I_1 \cup I_2 = \mathbb{Q}$$

and $I_1 \cap I_2$ is the join of at most two rational cosets of modulo q.

For example, the complementary of $[\frac{2}{3}, 5]\langle 29\rangle$ is $[-24, \frac{2}{3}]\langle 29\rangle$ (their intersection is $\frac{2}{3}\langle 29\rangle \cup 5\langle 29\rangle$) and the one of $[-\infty, \frac{9}{43}]\langle 0\rangle$ is $[\frac{9}{43}, +\infty]\langle 0\rangle$ (their intersection is $\frac{9}{43}\langle 0\rangle$).

3.2. Comparison on IC. In contrast to CC, an efficient comparison algorithm is provided here. Let us first redefine the order relation.

Definition 31 (Interval congruence comparison $\subseteq_\sharp$). The *comparison relation* $\subseteq_\sharp$ on IC is the extension to IC of the partial order relation on $\mathbb{P}(\mathbb{Q})$ induced by set inclusion.

$\subseteq_\sharp$ is a preorder relation. The following theorem reduces the general comparison to the particular case where the first of the compared elements is of zero modulo; the next theorem deals with this special case. In addition to the lemma characterizing interval congruences equal to $\mathbb{Q}$ or to $\emptyset$, they provide an algorithm to compare interval congruences which is implicitly given here.

Theorem 32 (Comparison with non zero first modulo). *Given $q_1 \neq 0$ and two interval congruences $I_1 = [a_1, b_1]\langle q_1\rangle$ and $I_2 = [a_2, b_2]\langle q_2\rangle$ neither empty nor equal to $\mathbb{Q}$, $I_1 \subseteq_\sharp I_2$ if and only if*

$$\left\{\begin{array}{ll} & q_2 = 0 \ \wedge \ b_2 < a_2 \ \wedge \ [b_2, a_2]\langle 0\rangle \subseteq_\sharp [b_1, a_1 + q_1]\langle q_1\rangle \qquad (18) \\ \vee & \\ & q_2 \neq 0 \ \wedge \ \left\lceil \frac{a_2 - a_1}{\gcd(q_1, q_2)} \right\rceil \ = \ \left\lfloor \frac{b_2 - b_1 - |q_2|}{\gcd(q_1, q_2)} \right\rfloor + 1 \qquad (19) \end{array}\right.$$

Proof. Some points of the following proof, which are very close to the one of the proof of proposition 23, are not fully explicated.
If the modulo of the second interval congruence is zero (case (18)), I_2 is infinite and its complementary must be in the complementary of I_1.
Otherwise, the modulo of the second interval congruence is not zero (case (19)).
Let $d = \gcd(q_1, q_2)$ and $q_2' = \frac{|q_2|}{d}$.
For all a in the rational interval $[a_1, b_1]$, the smallest set of rational cosets of modulo q_2 containing the interval congruence $[a, a]\langle q_1\rangle$ is

$$\{(a + d)\langle q_2\rangle, (a + 2d)\langle q_2\rangle, \ldots, (a + q_2'd)\langle q_2\rangle\}$$

Hence $[a, a]\langle q_1\rangle \subseteq_\sharp I_2$ is equivalent to

$$[a, a]\langle d\rangle = \bigcup_{i\in\mathbb{Z}} (a + id)\langle q_2\rangle \quad \subseteq \quad I_2$$

We follow now a reasoning similar to the end of the proof of the proposition 23 considering rational instead of integers.
It is equivalent to say that q_2' consecutive representatives of $[a, a]\langle d\rangle$ are in $[a_2, b_2]$ (recall that $q_2'd = |q_2|$) iff there exists an integer i such that

$$\begin{aligned} a_2 &\leq a + id \\ a + id + q_2'd - d &\leq b_2 \end{aligned}$$

and since this system is valid for any rational a in the interval $[a_1, b_1]$ we get

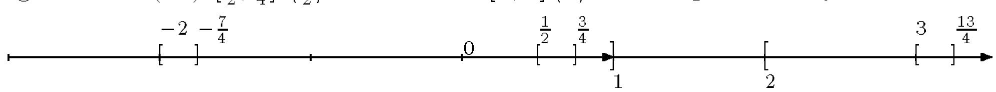
$$\begin{aligned} a_2 &\leq a_1 + id \\ b_1 + id + |q_2| - d &\leq b_2 \end{aligned}$$

which implies

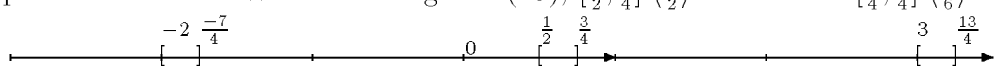
$$\left\lceil \frac{a_2 - a_1}{d} \right\rceil = \left\lfloor \frac{b_2 - b_1 - |q_2|}{d} + 1 \right\rfloor$$

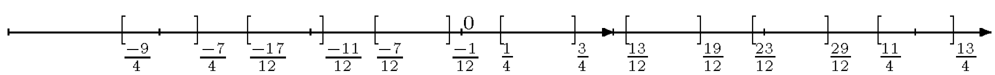
□

Examples
Following the rule (18) $[\frac{1}{2}, \frac{3}{4}]\,\langle\frac{5}{2}\rangle$ is less than $[2, 1]\,\langle 0\rangle$ as it is pictured by



where the big braces correspond to the interval congruence with zero modulo and the small ones represent the other. When following rule (19), $[\frac{1}{2}, \frac{3}{4}]\,\langle\frac{5}{2}\rangle$ is less than $[\frac{1}{4}, \frac{3}{4}]\,\langle\frac{5}{6}\rangle$





The following theorem assumes that the comparison of two intervals, both with zero modulo, is well known.

THEOREM 33 (COMPARISON WITH NULL FIRST MODULO). *Given two interval congruences* $I_1 = [a_1, b_1]\,\langle 0\rangle$ *and* $I_2 = [a_2, b_2]\,\langle q_2\rangle$ *neither empty nor equal to* $\mathbb{Q}$, $I_1 \subseteq_\sharp I_2$ *if and only if*

$$\left\{\begin{array}{ll} \quad q_2 = 0 \;\wedge\; I_1 \subseteq_\sharp I_2 & \quad (20) \\ \vee & \\ \quad q_2 \neq 0 \;\wedge\; -\infty < a_1 \leq b_1 < +\infty \;\wedge\; \left\lceil \frac{a_2 - a_1}{|q_2|} \right\rceil = \left\lfloor \frac{b_2 - b_1}{|q_2|} \right\rfloor & \quad (21) \end{array}\right.$$

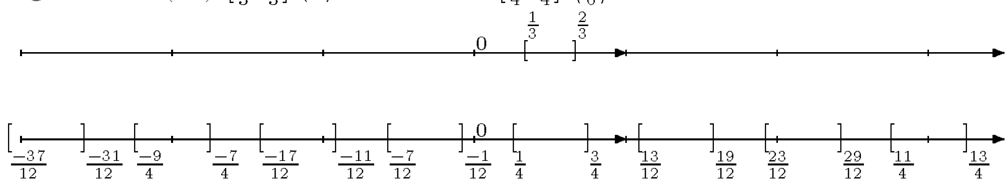
PROOF. The comparison of two interval congruences of zero modulo (case (20)) being quite trivial is not detailed here.
If the greatest interval congruence is of non zero modulo (case (21)), then I_1 is finite (because I_2 is not equal to $\mathbb{Q}$) and in one representative of I_2, which results in the existence of an integer i such that

$$\begin{array}{rcl} a_2 + i|q_2| & \leq & a_1 \\ b_1 & \leq & b_2 + i|q_2| \end{array}$$

and the result follows. □

Example

Following the rule (21) $[\frac{1}{3}, \frac{2}{3}]\,\langle 0\rangle$ is less than $[\frac{1}{4}, \frac{3}{4}]\,\langle \frac{5}{6}\rangle$



3.3. Equivalence relation. An algorithm for deciding the equivalence of interval congruences is provided. It does not rely upon the comparison algorithm. Notice the difference of complexity with respect to the equivalence algorithm on CC of theorem 19.

THEOREM & DEFINITION 34 (EQUIVALENCE $\approx_\sharp$). *The interval congruences* $I_1{=}[a_1, b_1]\,\langle q_1\rangle$ *and* $I_2 = [a_2, b_2]\,\langle q_2\rangle$ *represent the same rational set* ($I_1 \subseteq_\sharp I_2 \wedge I_2 \subseteq_\sharp I_1$)*, noted* $I_1 \approx_\sharp I_2$*, if and only if they are either both empty* ($q_i \neq 0, b_i < a_i$ *for* $i \in \{1,2\}$) *or the set of rational numbers* ($(q_i = 0 \wedge b_i = -a_i = +\infty) \vee (q_i \neq 0 \wedge b_i - a_i \geq |q_i|)$ *for* $i \in \{1,2\}$) *or have modulos with the same absolute value* $|q_1|$ *and satisfy* $b_2 - b_1 = a_2 - a_1 \in \langle |q_1| \rangle$*.* $\approx_\sharp$ *is an equivalence relation on* IC*.*

PROOF. Only the case where both interval congruences are neither empty nor equal to $\mathbb{Q}$ has to be explicated. It is easy to see that, in the other cases, an interval congruence with zero modulo and one with non zero modulo cannot be equivalent and that the theorem characterizes the equivalence between zero modulo interval congruences.

Now, suppose we have two non empty, non equal to $\mathbb{Q}$ interval congruences with non zero modulo. They are equivalent if the case (19) of theorem 32 is verified for both $I_1 \subseteq_\sharp I_2$ and $I_2 \subseteq_\sharp I_1$ which gives:

$$\left\lceil \frac{a_2 - a_1}{d} \right\rceil \;=\; \left\lfloor \frac{b_2 - b_1 - |q_2|}{d} \right\rfloor + 1 \quad\wedge\quad \left\lceil \frac{a_1 - a_2}{d} \right\rceil \;=\; \left\lfloor \frac{b_1 - b_2 - |q_1|}{d} \right\rfloor + 1$$

where $d = \gcd(q_1, q_2)$. Suppose d does not divide $a_2 - a_1$ then[7]

$$\left\lceil \frac{a_2 - a_1}{d} \right\rceil = -\left\lceil \frac{a_1 - a_2}{d} \right\rceil + 1$$

hence

$$\left\lfloor \frac{b_2 - b_1 - |q_2|}{d} \right\rfloor + 1 = -\left\lfloor \frac{b_1 - b_2 - |q_1|}{d} \right\rfloor$$

[7] A non integer rational number a verifies $\lceil a \rceil = -\lceil -a \rceil + 1$.

and there exists an integer i such that

$$\begin{array}{rcl} i \leq & \dfrac{b_2 - b_1 - |q_2|}{d} + 1 & < i + 1 \\ -i \leq & \dfrac{b_1 - b_2 - |q_1|}{d} & < -i + 1 \end{array}$$

and

$$\begin{array}{rcl} i - 1 + q'_2 \leq & \dfrac{b_2 - b_1}{d} & < i + q'_2 \\ i - 1 - q'_1 < & \dfrac{b_2 - b_1}{d} & \leq i - q'_1 \end{array}$$

where $|q_1| = dq'_1$ and $|q_2| = dq'_2$. The existence of $\frac{b_2-b_1}{d}$ implies (following footnote 3 on page 45)

$$\begin{array}{rcl} -1 - q'_1 & < & q'_2 \\ -1 + q'_2 & \leq & -q'_1 \end{array}$$

The latter inequality implies that the sum of the positive integers q'_1 and q'_2 is less than one which is impossible. Hence d divides $a_1 - a_2$ and we have

$$\left\lfloor \frac{b_2 - b_1 - |q_2|}{d} \right\rfloor + 1 \quad = \quad - \left\lfloor \frac{b_1 - b_2 - |q_1|}{d} \right\rfloor - 1$$

and, following the same scheme as above, it is established that there exists an integer i such that

$$\left\{ \begin{array}{rcccl} i - 1 + q'_2 & \leq & \frac{b_2-b_1}{d} & < & i + q'_2 \\ i - q'_1 & < & \frac{b_2-b_1}{d} & \leq & i + 1 - q'_1 \end{array} \right. \tag{22}$$

and

$$\begin{array}{rcl} -q'_1 & < & q'_2 \\ -1 + q'_2 & \leq & 1 - q'_1 \end{array}$$

Hence $q'_1 = q'_2 = 1$ and $|q_1| = |q_2| = d$. The substitution of 1 to q'_1 and to q'_2 in the system (22) provides

$$\frac{b_2 - b_1}{d} \quad = \quad i$$

hence d divides $b_2 - b_1$ too. The proof of the equality of the distances separating upper and lower bounds is trivial. □

The preceding theorem states that $(IC, \subseteq_\sharp)$ is a preorder and we use the equivalence classes on IC induced by $\approx_\sharp$ for the next steps of the construction of our abstraction. Hence from now on and to avoid notational complications, we note an equivalence class of $IC/_{\approx_\sharp}$ by one of its representative and the partial order on $IC/_{\approx_\sharp}$ by $\subseteq_\sharp$. There is no need here for a normalization operator as in CC because operators on IC are compatible with the equivalence relation.

APPENDIX A

Equivalence relation on CC

Before stating that coset congruences with distinct modulos are distinct, two lemmas concerning the conversion of coset congruences to a new modulo are established. Let us first show that given a coset congruence of non zero modulo, non empty and non equal to $\mathbb{Z}$, the smallest (for set inclusion induced order) coset congruence, with a fixed new modulo, containing it, is equal to $\mathbb{Z}$, if the number of consecutive integer cosets of the initial coset congruence is greater than the gcd of the two modulos.

LEMMA 35 (COSET CONGRUENCE CONVERSION GIVING $\mathbb{Z}$). *Let* $C_1 = \theta_1.[l_1, u_1]\langle m_1\rangle$ *and* $C_2 = \theta_2.[l_2, u_2]\langle m_2\rangle$ *be two coset congruences.*

$$\left.\begin{array}{c} 0 < u_1 - l_1 + 1 < |m_1| \\ u_1 - l_1 + 1 \geq \gcd(m_1, m_2) \\ C_1 \subseteq C_2 \end{array}\right\} \Rightarrow C_2 = \mathbb{Z}$$

PROOF. Let us recall that C_1 is non empty and non equal to $\mathbb{Z}$ and has a non zero modulo. We are going to show that the smallest coset congruence of modulo m_2 containing C_1 is $\mathbb{Z}$. First $\theta_1 l_1\langle m_1\rangle, \theta_1(l_1+1)\langle m_1\rangle, \ldots, \theta_1 u_1\langle m_1\rangle \subseteq C_1$ hence

$$\theta_1 l_1\langle m_1\rangle \;\cup\; \theta_1(l_1+1)\langle m_1\rangle \;\cup\; \ldots \;\cup\; \theta_1 u_1\langle m_1\rangle \quad\subseteq\quad C_2$$

Since C_2 is of modulo m_2 and by proposition 8 $\langle m_1\rangle \sqcup \langle m_2\rangle = \langle \gcd(m_1, m_2)\rangle$, then

$$\theta_1 l_1\langle \gcd(m_1, m_2)\rangle \;\cup\; \theta_1(l_1+1)\langle \gcd(m_1, m_2)\rangle \;\cup\; \ldots \;\cup\; \theta_1 u_1\langle \gcd(m_1, m_2)\rangle \quad\subseteq\quad C_2$$

$\gcd(\theta_1, \gcd(m_1, m_2)) = 1$ implies that the left hand side of the latter inclusion is the coset congruence $\theta_1.[l_1, u_1]\langle \gcd(m_1, m_2)\rangle$ and lemma 17, under the hypothesis $u_1 - l_1 + 1 \geq \gcd(m_1, m_2)$, shows that it is $\mathbb{Z}$. $\square$

Now, if we negate the condition comparing the number of distinct cosets constituting the original coset congruence with the gcd of modulos, a lower bound on the number of distinct cosets constituting the resulting coset congruence is determined.

LEMMA 36 (REPRESENTATIVE WIDTH OF CONVERTED COSET CONGRUENCE). *Let* $C_1 = \theta_1.[l_1, u_1]\langle m_1\rangle$ *and* $C_2 = \theta_2.[l_2, u_2]\langle m_2\rangle$ *be two coset congruences.*

$$\left.\begin{array}{c} 0 < u_1 - l_1 + 1 < |m_1| \\ u_1 - l_1 + 1 < \gcd(m_1, m_2) \\ C_1 \subseteq C_2 \end{array}\right\} \Rightarrow u_2 - l_2 + 1 \geq \frac{|m_2|}{\gcd(m_1, m_2)}(u_1 - l_1 + 1)$$

PROOF. Let $d = \gcd(m_1, m_2)$ and $q = \frac{|m_2|}{d}$. From the proof of lemma 35 we have

$$\theta_1.[l_1, u_1]\langle d\rangle \quad \subseteq \quad C_2$$

Since $\theta_1.[l_1, u_1]\langle d\rangle$ is the join of $u_1 - l_1 + 1$ distinct integer cosets of modulo d, each of which satisfies (for the corresponding integer r)

$$\begin{array}{rcl} \theta_1.[r, r]\langle d\rangle & = & \theta_1 r\langle m_2\rangle \;\cup\; (\theta_1 r + d)\langle m_2\rangle \;\cup\; (\theta_1 r + 2d)\langle m_2\rangle \\ & & \cup\; \ldots \;\cup\; (\theta_1 r + (q-1)d)\langle m_2\rangle \end{array}$$

where the q single integer cosets are distinct, then $\theta_1.[l_1, u_1]\langle d\rangle$ is the join of $q(u_1 - l_1 + 1)$ distinct cosets of modulo m_2, hence $q(u_1 - l_1 + 1) \leq u_2 - l_2 + 1$. □

LEMMA 37 (COSET CONGRUENCES OF DISTINCT MODULO ARE DISTINCT). *Let* $C_1 = \theta_1.[l_1, u_1]\langle m_1\rangle$ *and* $C_2 = \theta_2.[l_2, u_2]\langle m_2\rangle$ *be two coset congruences*

$$\left.\begin{array}{c} 0 < u_1 - l_1 + 1 < |m_1| \\ 0 < u_2 - l_2 + 1 < |m_2| \\ |m_1| \neq |m_2| \end{array}\right\} \quad \Rightarrow \quad C_1 \not\approx C_2$$

PROOF. Let $q_1 = \frac{|m_1|}{\gcd(m_1, m_2)}$ and $q_2 = \frac{|m_2|}{\gcd(m_1, m_2)}$.
If $u_1 - l_1 + 1 \geq \gcd(m_1, m_2)$ lemma 35 shows that the only way for C_2 to contain C_1 is to be $\mathbb{Z}$ which is impossible by hypothesis.
If $u_1 - l_1 + 1 < \gcd(m_1, m_2)$ lemma 36 implies that $u_2 - l_2 + 1 \geq q_2(u_1 - l_1 + 1)$ and that $u_1 - l_1 + 1 \geq q_1(u_2 - l_2 + 1)$ leading to $u_1 - l_1 + 1 \geq q_1 q_2(u_1 - l_1 + 1)$ and $q_1 = q_2 = 1$ and finally $|m_1| = |m_2|$ which is incompatible with the hypotheses. □

The preceding lemma is extensible to coset congruences with zero modulos but which are non empty and non equal to $\mathbb{Z}$[1]. The following lemma characterizes equivalence between coset congruences of identical modulo when one of them has its offset equal to one[2].

[1] It states that two coset congruences non empty and non equal to $\mathbb{Z}$ of distinct modulo absolute value are distinct.

[2] These coset congruences intuitively correspond to usual integer intervals regularly dispersed following a pattern of length the value of the modulo.

LEMMA 38 (EQUIVALENCE TO A COSET CONGRUENCE OF OFFSET ONE). *Let m be a positive integer, l, θ and n three integers such that* $\gcd(\theta, m) = 1$ *and* $2 \leq n+1 \leq |m| - 2$.

$$\theta.[l, l+n]\langle m\rangle \approx 1.[0,n]\langle m\rangle \Leftrightarrow \left\{ \begin{array}{lll} & \theta \in 1\langle m\rangle & \wedge \quad l \in \langle m\rangle \\ \vee & & \\ & \theta \in -1\langle m\rangle & \wedge \quad l \in -n\langle m\rangle \end{array} \right. \tag{23}$$

PROOF. If $\theta \in 0\langle m\rangle = \langle m\rangle$ then $\gcd(\theta, m) = 1$ implies $|m| = 1$ which contradicts the hypothesis $|m| \geq 4$.
If $\theta \in -1\langle m\rangle$, we are going to build a one to one correspondence between the n distinct cosets constituting $\theta.[l, l+n]\langle m\rangle$ and $1.[0,n]\langle m\rangle$ by identifying the identical cosets. Recall that $\theta \overset{m}{=} -1$, so the cosets of $\theta.[l, l+n]\langle m\rangle$ are

$$-l\langle m\rangle, (-l-1)\langle m\rangle, \ldots, (-l-n)\langle m\rangle$$

and in reverse order

$$(-l-n)\langle m\rangle, (-l-n+1)\langle m\rangle, \ldots, -l\langle m\rangle$$

The cosets of $1.[0,n]\langle m\rangle$ are

$$0\langle m\rangle, 1\langle m\rangle, \ldots, n\langle m\rangle$$

and the only way to build the correspondence between identical cosets is to associate $(-l-n+i)\langle m\rangle$ to $i\langle m\rangle$ for $0 \leq i \leq n$. Indeed, if the correspondence associates $(-l-n+i)\langle m\rangle$ to $i'\langle m\rangle$ (with $i' - i \notin \langle m\rangle$), it should associate $(-l-n+i+k)\langle m\rangle$ to $(i'+k)\langle m\rangle$ for $n+1$ consecutive integer values of k which is impossible because it would associate some coset of one set to some coset that does not appear in the other set. In particular, for $i = 0$, the correspondence requires $(-l-n)\langle m\rangle = 0\langle m\rangle$ and $l \in -n\langle m\rangle$ that provides the result.
Now we can suppose that $\theta \notin 1.[-1, 0]\langle m\rangle$. It is sufficient to show that $\theta \notin 1.[2, |m|-2]\langle m\rangle$ and hence the only solution is $\theta \in 1\langle m\rangle$ and clearly $l \in \langle m\rangle$.
Suppose that $\theta \in 1.[2, |m|-2]\langle m\rangle$ (where $|m| > 3$). Since $n+1 \leq |m|-2$, the coset congruence $1.[n+1-\theta, |m|-2-\theta]\langle m\rangle$ is non empty. For every value of θ, there exists an integer k such that $n+1-\theta \leq n-1+km$ and $|m|-2-\theta \geq km$ (just applying the definition of coset congruences to $\theta \in 1.[2, |m|-2]\langle m\rangle$). For that k, we have[3] $[km, n-1+km] \cap [n+1-\theta, |m|-2-\theta] \neq \emptyset$ hence

$$M = 1.[0, n-1]\langle m\rangle \cap 1.[n+1-\theta, |m|-2-\theta]\langle m\rangle \neq \emptyset$$

Let $\mu \in M$, hence $\mu \in 1.[0, n-1]\langle m\rangle \subset 1.[0,n]\langle m\rangle$. The hypothesis implies that there exist $\kappa \in [l, l+n]$ and $k \in \mathbb{Z}$ such that $\mu = \kappa\theta + km$. On the other hand $\mu + 1 \in 1.[0,n]\langle m\rangle$; similarly there exist $\kappa' \in [l, l+n]$ and $k' \in \mathbb{Z}$ such that $\mu + 1 = \kappa'\theta + k'm$. We have $\kappa\theta + km + 1 = \kappa'\theta + k'm$. Suppose $\kappa - \kappa' \in \langle m\rangle$, then $1 = (\kappa' - \kappa)\theta + (k'-k)m \in \langle m\rangle$ which is impossible for $|m| \geq 4$; hence $\kappa\langle m\rangle \neq \kappa'\langle m\rangle$. Hence at least one of the cosets $\kappa\langle m\rangle$ and $\kappa'\langle m\rangle$ is different from $(l+n)\langle m\rangle$; first, suppose $\kappa \neq l+n$. Then we have $(\mu+\theta)\langle m\rangle = \theta(\kappa+1)\langle m\rangle$ because $\mu = \kappa\theta + km$ and $\theta(\kappa+1)\langle m\rangle \subseteq 1.[0,n]\langle m\rangle$ because

[3] Recall that $[a,b] \cap [c,d] \neq \emptyset \Leftrightarrow c \leq b \ \wedge \ d \geq a$

of $\kappa \in [l, l+n-1]$ and of the left hand-side of 23. $(\mu+\theta)\langle m\rangle \subseteq 1.[0,n]\langle m\rangle$ contradicts the choice of $\mu+\theta$ in $1.[n+1,|m|-2]\langle m\rangle$ in the definition of M. In second place suppose $\kappa' \neq l+n$. Then we have $(\mu+1+\theta)\langle m\rangle = \theta(\kappa'+1)\langle m\rangle$ because $\mu+1 = \kappa'\theta + km$ and $\theta(\kappa'+1)\langle m\rangle \subseteq 1.[0,n]\langle m\rangle$ because of $\kappa' \in [l, l+n-1]$ and of the left hand-side of 23. $(\mu+1+\theta)\langle m\rangle \subseteq 1.[0,n]\langle m\rangle$ contradicts the choice of $\mu+1+\theta$ in $1.[n+2,|m|-1]\langle m\rangle$ in the definition of M. The result follows. $\square$

The general case for testing the equivalence of coset congruences of identical modulo is now provided.

THEOREM 39 (EQUIVALENCE OF COSET CONGRUENCES WITH IDENTICAL MODULO). *Let* $C_1 = \theta_1.[l_1, u_1]\langle m_1\rangle$ *and* $C_2 = \theta_2.[l_2, u_2]\langle m_2\rangle$ *be two coset congruences such that*

$$m = |m_1| = |m_2| \neq 0$$
$$1 \leq w = u_2 - l_2 + 1 = u_1 - l_1 + 1 \leq m-1$$

$C_1 \approx C_2$ *if and only if*

$$\begin{array}{c}
\left\{\begin{array}{c} w = 1 \\ \theta_1 l_1 \overset{m}{=} \theta_2 l_2 \end{array}\right. \\
\vee \\
\left\{\begin{array}{c} 2 \leq w \leq m-2 \\ \theta_1 \overset{m}{=} \theta_2 \\ \theta_1 l_1 \overset{m}{=} \theta_2 l_2 \end{array}\right. \\
\vee \\
\left\{\begin{array}{c} 2 \leq w \leq m-2 \\ \theta_1 \overset{m}{=} -\theta_2 \\ \theta_1 l_1 \overset{m}{=} \theta_2 u_2 \end{array}\right. \\
\vee \\
\left\{\begin{array}{c} w = m-1 \\ \theta_1(l_1-1) \overset{m}{=} \theta_2(l_2-1) \end{array}\right.
\end{array} \tag{24}$$

PROOF. The considered cosets are neither empty nor equal to $\mathbb{Z}$.
If $w=1$, we have to compare integer cosets which results in comparing their representatives $\theta_1 l_1$ and $\theta_2 l_2$.
If $2 \leq w \leq m-2$, we are going to show that $C_1 \approx C_2$ is equivalent to

$$\theta_2^{-1}\theta_1.[l_1 - \theta_1^{-1}\theta_2 l_2, u_1 - \theta_1^{-1}\theta_2 l_2]\langle m\rangle \approx 1.[0, u_2 - l_2]\langle m\rangle \tag{25}$$

by showing that the equality of the two cosets $\kappa_1\theta_1\langle m\rangle$ and $\kappa_2\theta_2\langle m\rangle$ respectively in C_1 and in C_2 is equivalent to the equality of the two cosets $\theta_2^{-1}\theta_1(\kappa_1 - \theta_1^{-1}\theta_2 l_2)\langle m\rangle$ and $(\kappa_2 - l_2)\langle m\rangle$, where θ_1^{-1} and θ_2^{-1} are chosen such that $\theta_1\theta_1^{-1} \overset{m}{=} 1$ and $\theta_2\theta_2^{-1} \overset{m}{=} 1$.
The relation

$$\kappa_1\theta_1 - \kappa_2\theta_2 \in \langle m\rangle$$

is equivalent to

$$\kappa_1\theta_1\theta_2^{-1} - \kappa_2\theta_2\theta_2^{-1} \in \langle m \rangle$$

since $\gcd(\theta_2, m) = 1$, which is in turn equivalent to

$$\kappa_1\theta_1\theta_2^{-1} - \kappa_2 + l_2 - l_2 \underbrace{(\theta_1\theta_1^{-1})}_{\overset{m}{=}1} \underbrace{(\theta_2\theta_2^{-1})}_{\overset{m}{=}1} \in \langle m \rangle$$

and finally

$$\theta_2^{-1}\theta_1(\kappa_1 - \theta_1^{-1}\theta_2 l_2)\langle m \rangle = (\kappa_2 - l_2)\langle m \rangle$$

which provides equality (25). Since $\gcd(\theta_2^{-1}\theta_1, m) = 1$ and $2 \leq u_2 - l_2 + 1 \leq m - 2$, lemma 38 provides the result.
If $w = m - 1$ the problem results in comparing the complementaries of C_1 and C_2 (which are simple cosets) that are respectively $\theta_1(l_1 - 1)\langle m \rangle$ and $\theta_2(l_2 - 1)\langle m \rangle$. □

PROOF. [of theorem 19] Notice that Lemma 17 (resp. lemma 18) provides the result when the considered integer set is $\mathbb{Z}$ (resp. $\emptyset$) and corresponds to case (10) (resp. case (11)).
Suppose that C_1 and C_2 are neither empty nor equal to $\mathbb{Z}$.
First lemma 37 implies that $|m_1| = |m_2|$ even if $m_1 = 0$. In addition, two coset congruences non empty and non equal to $\mathbb{Z}$ with the same absolute value of modulo are equal if and only if the differences between their upper and lower bounds are equal.
Now, if the modulo is not zero, theorem 39 has to be considered (for one part of case (12)) and, if the common modulo is zero, then both offsets are one by definition and the representative bounds have to be equal modulo m (which is indeed taken into account by case (12)). □

CHAPTER IV

ABSTRACT INTERPRETATION OF INTERVAL CONGRUENCES

This chapter is devoted to the design of some abstract interpretations using the two domains described in Chapter III. First the connection between these two domains is provided in section 1; its particular features are expressed in terms of the general abstract interpretation framework [CC92b]. Then the approximate operators on the abstract domain are determined together with the widening operator in the section 2. Finally, section 3 provides the abstract statments and is ended with a complete analysis example.

1. Semantic operators

The concrete domain CC and the abstract one IC are designed in Chapter III; we now bind them using a pair of abstraction and concretization functions in order to give the meaning of the abstract elements and to prove that their respective orders are coherent.

1.1. Soundness relation. The definition of a soundness relation formalizes the intuitive concept that an integer set is well approximated by a rational one if the original integer set is included in that given rational set.

Definition 40 (The soundness relation σ). It is defined by

$$\sigma \stackrel{\text{def}}{=} \{(C, I) \in CC/_{\approx} \times IC/_{\approx_\sharp}, C \subseteq I\}$$

The order relation $\subseteq$ used in the definition is simply the usual inclusion between sets. The soundness relation is implied by the relation $\{(C, I) \in CC/_{\approx} \times IC/_{\approx_\sharp}, \alpha(C) \subseteq_\sharp I\}$; the reciprocal is false (see in the proof of proposition 47 an example illustrating that (α, γ) is not a Galois connection, i.e. an example of coset congruence contained in an interval congruence for which its abstraction is not contained in that interval congruence).

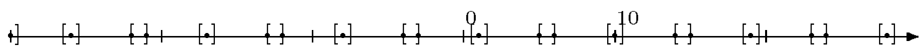
1.2. Abstraction. The choice of an interval congruence representing a given coset congruence is formalized by the abstraction function: the chosen abstract element is one of the minimal approximations of the concrete one. Given one coset congruence, many interval congruences contain it (they are provided by the soundness relation); there are still many containing exactly the integers corresponding to the original coset congruence; finally there are still many of these of minimum representative width (informally the difference between the upper and the lower bounds).

DEFINITION 41 (ABSTRACTION α). The *abstraction function* is the following:

$$\begin{array}{rcccl} \alpha & : & CC/_{\approx} & \to & IC/_{\approx_\sharp} \\ & & \theta.\,[l,u]\,\langle m\rangle & \mapsto & \left[\frac{l}{\theta^{-1}},\frac{u}{\theta^{-1}}\right]\left\langle\frac{m}{\theta^{-1}}\right\rangle \end{array}$$

where $0 < \theta^{-1} < |m|$ is an inverse of θ with respect to m and with the convention that the inverse of 0 with respect to 1 is 1.

Following Bezout's theorem (See footnote 1 on page 25), the abstraction function is always defined (θ^{-1} always exists).
The abstraction could have been defined as a relation if we had not chosen a unique inverse of θ but, since a normal form exists for θ^{-1} and is easily computable, we prefer to have a function. For example $\alpha\,(5.\,[1,3]\,\langle 9\rangle) = \left[\frac{1}{2},\frac{3}{2}\right]\left\langle\frac{9}{2}\right\rangle$ that is represented by



$\left[\frac{6}{7},\frac{8}{7}\right]\left\langle\frac{9}{7}\right\rangle$ is an other minimal interval congruence containing $5.\,[1,3]\,\langle 9\rangle$ and no more integers. It is of course non comparable with $\left[\frac{1}{2},\frac{3}{2}\right]\left\langle\frac{9}{2}\right\rangle$. This illustrates the lack of a best approximation of an element of CC with an interval congruence. It is optimal if $\gamma \circ \alpha$ is the identity (which is in fact ensured by theorem 44).

1.3. Concretization. The concretization function associates a concrete element with an abstract one giving its meaning.

DEFINITION 42 (CONCRETIZATION γ). The *concretization function* is defined by

$$\begin{array}{rcccll} \gamma & : & IC/_{\approx_\sharp} & \to & CC/_{\approx} & \\ & & [a,b]\left\langle\frac{\nu}{\delta}\right\rangle & \mapsto & \begin{cases} 1.\,[1,0]\,\langle 1\rangle & \text{if } \nu = 0 \text{ and } b \geq a \text{ and } \lceil a\rceil > \lfloor b\rfloor \\ \|\delta^{-1}.\,[\lceil a\delta\rceil,\lfloor b\delta\rfloor]\,\langle\nu\rangle\| & \text{otherwise} \end{cases} & \begin{array}{r}(26)\\(27)\end{array} \end{array}$$

where δ^{-1} is an inverse of δ with respect to ν.

The same remark as for the choice of the inverse of θ in the abstraction definition holds here for the choice of δ^{-1}, except that all the different possibilities reach here the same element of $CC/_{\approx}$ (because of the normalization on CC) and though there is in fact no choice. We see that considering rational interval congruences provides a much more powerful description of concrete properties than only considering integer interval congruences the definition of which would have been quite similar to the definition 25 replacing $\mathbb{Q}$ by $\mathbb{Z}$. This is a direct consequence of the strict inclusion of these integer interval congruences in IC. An example of concretization is:

$$\gamma\left(\left[\frac{3}{4},\frac{3}{2}\right]\left\langle\frac{9}{4}\right\rangle\right) = 7.\,[3,6]\,\langle 9\rangle = (1\,\langle 9\rangle)\cup(3\,\langle 9\rangle)\cup(6\,\langle 9\rangle)\cup(8\,\langle 9\rangle)$$

To prove the fundamental property about γ, we first need to show a sufficient condition for two interval congruences to have the same integer subset.

LEMMA 43 (EQUAL INTEGER SUBSET WITH IDENTICAL MODULO). *Let ν be a non zero positive integer and $I_1 = [a_1, b_1]\left\langle\frac{\nu}{\delta}\right\rangle$ and $I_2 = [a_2, b_2]\left\langle\frac{\nu}{\delta}\right\rangle$ two interval congruences such that $\lceil a_1\delta\rceil \le \lfloor b_1\delta\rfloor$ and $\lceil a_2\delta\rceil \le \lfloor b_2\delta\rfloor$. I_1 and I_2 have the same integer subset if*

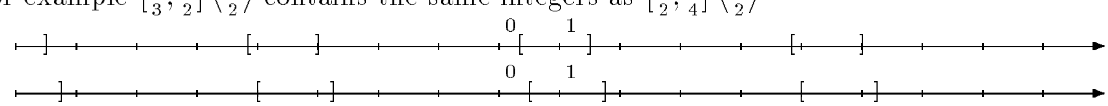
$$\lceil a_1\delta\rceil - \lceil a_2\delta\rceil \;=\; \lfloor b_1\delta\rfloor - \lfloor b_2\delta\rfloor \;\in\; \langle\nu\rangle \tag{28}$$

PROOF. Theorem 27 associates to I_1 the equation $x \equiv [a_1, b_1]\left\langle\frac{\nu}{\delta}\right\rangle$, which has the same integer solutions[1] as the equation

$$x \equiv \left[\frac{\lceil a_1\delta\rceil}{\delta}, \frac{\lfloor b_1\delta\rfloor}{\delta}\right]\left\langle\frac{\nu}{\delta}\right\rangle$$

An equivalent deduction is satisfied for I_2 and equation (28) proves the equality of equations. $\square$

For example $\left[\frac{1}{3},\frac{3}{2}\right]\left\langle\frac{9}{2}\right\rangle$ contains the same integers as $\left[\frac{1}{2},\frac{7}{4}\right]\left\langle\frac{9}{2}\right\rangle$



The next step establishes that the concretization function corresponds to our initial goal to express the integer subset of an interval congruence.

THEOREM 44 (CORRECTNESS OF γ). *The meaning $\gamma(I)$ of an interval congruence I is its intersection with $\mathbb{Z}$.*

$$\forall I \in IC\ \gamma(I) = I \cap \mathbb{Z}$$

PROOF. We do not have to consider here the normalization step in the concretization process since it does not change the resulting integer set. Let us consider the different cases for an interval congruence $I = [a, b]\left\langle\frac{\nu}{\delta}\right\rangle$:

$a = b \wedge \nu = 0 \wedge \lceil a\delta\rceil > \lfloor b\delta\rfloor$: The resulting interval congruence is the interval $[a, a]$ and since $\lceil a\rceil > \lfloor b\rfloor$, $a \notin \mathbb{Z}$ and its integer subset is empty.

$a < b \wedge \nu = 0 \wedge \lceil a\delta\rceil > \lfloor b\delta\rfloor$: Since $\lceil a\rceil > \lfloor b\rfloor$, a and b are two distinct rational numbers without any integer between them; the resulting interval congruence is $[a, b]\,\langle 0\rangle$ and its integer subset is empty.

[1] A basic result for solving arithmetical congruence equations states that $\alpha x \equiv a \mod (q)$, $(\alpha, a, q \in \mathbb{Q})$ has an integer solution if and only if $\gcd(\alpha, q)$ divides a. $\frac{\lceil a_1\delta\rceil}{\delta}$ is the smallest rational representative greater than a_1 for which the preceding property is verified in the equation corresponding to I_1. The symmetrical result holds for $\frac{\lfloor b_1\delta\rfloor}{\delta}$.

$a > b \wedge \nu = 0 \wedge \lceil a\delta \rceil > \lfloor b\delta \rfloor$: The resulting interval congruence corresponds to $[a, +\infty] \cup [-\infty, b]$ and its concretization $1.[\lceil a \rceil, \lfloor b \rfloor]\langle 0 \rangle$ to $[\lceil a \rceil, +\infty] \cup [-\infty, \lfloor b \rfloor]$ which is exactly the integer subset of $[a, +\infty] \cup [-\infty, b]$.

$\nu \neq 0 \wedge \lceil a\delta \rceil > \lfloor b\delta \rfloor$: Lemma 18 proves the emptyness of $\gamma(I)$ and definition 25 the emptyness of I.

$\lceil a\delta \rceil \leq \lfloor b\delta \rfloor \wedge \nu = 0$: Then $\gamma(I) = \|1.[\lceil a \rceil, \lfloor b \rfloor]\langle 0 \rangle\|$; the concretization is here the intersection of a usual rational interval with the set of integers.

$\lceil a\delta \rceil \leq \lfloor b\delta \rfloor \wedge \nu \neq 0$: A direct consequence of lemma 43 is that

$$I \cap \mathbb{Z} = \left[\frac{\lceil a\delta \rceil}{\delta}, \frac{\lfloor b\delta \rfloor}{\delta}\right] \left\langle \frac{\nu}{\delta} \right\rangle \cap \mathbb{Z}$$

Then solving the resulting integer congruence equation

$$\delta x \equiv \kappa \mod \nu \ \wedge \ \lceil a\delta \rceil \leq \kappa \leq \lfloor b\delta \rfloor \tag{29}$$

provides an expression of $I \cap \mathbb{Z}$. The solution set of equation (29) is the union of a set of cosets with identical modulo ν and with representative a particular solution that is given for example by $x_0 = \kappa\theta$ such that $\delta\theta \in 1\langle \nu \rangle$ (hence $\gcd(\theta, \nu) = 1$) which is exactly the description of $\gamma(I)$, the coset congruence of offset θ, lower bound $\lceil a\delta \rceil$, upper bound $\lfloor b\delta \rfloor$ and modulo ν.

□

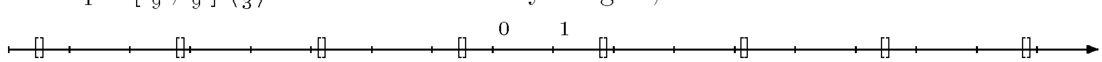
1.4. Characteristics of the connection (α, γ). Two classes of abstract properties are first characterized with respect to their meaning, then the structure of the abstraction–concretization connection is dealt with, which directly results from the fact that our domains are not Moore families.

PROPOSITION 45 (CHARACTERIZATION OF $\gamma^{-1}(\emptyset)$). *Let $I = [a, b]\langle \frac{\nu}{\delta} \rangle$ be an interval congruence. It is empty if and only if*

$$\begin{cases} \nu \neq 0 \ \vee \ a \leq b \\ \lceil a\delta \rceil > \lfloor b\delta \rfloor \end{cases}$$

PROOF. Considering lemma 18, the case (26) of the concretization definition always leads to the empty integer set. By lemma 18, the case (27) is the empty integer set if and only if $\nu \neq 0$ and $\lceil a\delta \rceil > \lfloor b\delta \rfloor$.
The last equality directly results from the theorem 44. □

For example $[\frac{16}{9}, \frac{17}{9}]\langle \frac{7}{3} \rangle$ does not contain any integers, as is visible on



PROPOSITION 46 (CHARACTERIZATION OF INTERVAL CONGRUENCES CONTAINING $\mathbb{Z}$). *Let $I = [a, b]\left\langle \frac{\nu}{\delta} \right\rangle$ be an interval congruence. It contains $\mathbb{Z}$ if and only if*

$$\begin{cases} \nu = 0 \ \wedge \ \lceil a \rceil = \lfloor b \rfloor + 1 \ \wedge \ b < a & (30) \\ \vee & \\ a = -\infty \ \wedge \ b = +\infty & (31) \\ \vee & \\ 0 < \nu \leq \lfloor b\delta \rfloor - \lceil a\delta \rceil + 1 & (32) \end{cases}$$

PROOF. Theorem 44 transforms the problem into characterizing $\gamma^{-1}(\mathbb{Z})$. The final result comes from lemma 17. $\square$

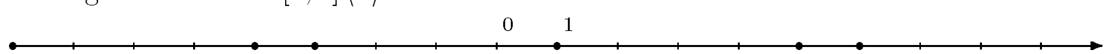
For example $\left[\frac{-4}{9}, \frac{16}{9}\right]\left\langle \frac{7}{3} \right\rangle$ and $\left[\frac{3}{4}, \frac{1}{2}\right]\langle 0 \rangle$ contain the set of integers.
These two propositions provide tests on the emptyness and the fullness of the meaning of an interval congruence which are very frequently used operators in the implementation of the program analyzer.

PROPOSITION 47 (STRUCTURE OF (α, γ)). *The pair of maps (α, γ) is not a Galois connection.*

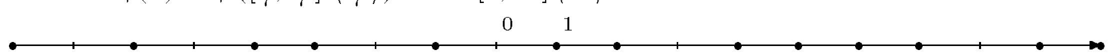
PROOF. Recall that (α, γ) would have been a Galois connection if

$$\forall C \in CC, I \in IC \ \alpha(C) \subseteq_{\sharp} I \Leftrightarrow C \subseteq \gamma(I)$$

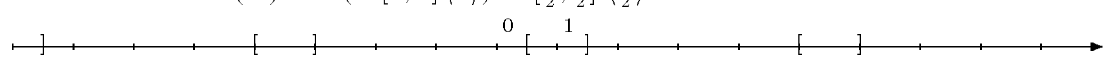
that is equivalent to stating that for every interval congruence I, $\alpha(\gamma(I)) = I$. However the coset congruence $C = 5.[1, 3]\langle 9 \rangle$



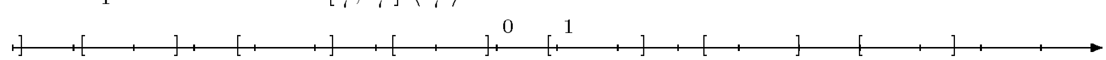
is less than $\gamma(I) = \gamma\left(\left[\frac{6}{7}, \frac{17}{7}\right]\left\langle \frac{18}{7} \right\rangle\right) = 13.[6, 17]\langle 18 \rangle$



but its abstraction $\alpha(C) = \alpha(5.[1, 3]\langle 9 \rangle) = \left[\frac{1}{2}, \frac{3}{2}\right]\left\langle \frac{9}{2} \right\rangle$



is not comparable with $I = \left[\frac{6}{7}, \frac{17}{7}\right]\left\langle \frac{18}{7} \right\rangle$



which contradicts the Galois connection character of the pair (α, γ). $\square$

Hence the usual framework of [CC77] cannot be used and [CC92b] shall be used instead.

1.5. Normalization on IC. A major consequence of the normalized feature of the concrete coset congruences is that $\gamma \circ \alpha$ is the identity operator. We are now going to consider the inverse operator $\alpha \circ \gamma$ as a normalization operator on IC.

PROPOSITION 48 (SEMANTIC MINIMIZATION). *Let $I = [a,b]\left\langle\frac{\nu}{\delta}\right\rangle$ be an interval congruence containing integers but not $\mathbb{Z}$.*

$$\left[\frac{\lceil a\delta\rceil}{\delta}, \frac{\lfloor b\delta\rfloor}{\delta}\right]\left\langle\frac{\nu}{\delta}\right\rangle$$

is the smallest interval congruence with the same concretization and modulo as I.

$$\forall I_1 \in IC\left(\frac{\nu}{\delta}\right) \quad \emptyset \neq I_1 \cap \mathbb{Z} = I \cap \mathbb{Z} \neq \mathbb{Z} \quad \Rightarrow \quad \left[\frac{\lceil a\delta\rceil}{\delta}, \frac{\lfloor b\delta\rfloor}{\delta}\right]\left\langle\frac{\nu}{\delta}\right\rangle \subseteq_\sharp I_1$$

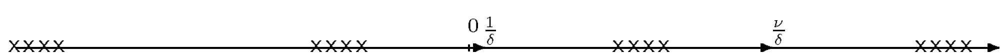
PROOF. If the modulo of I is zero, the result is easy to get.
Suppose now that I has a non zero modulo and does contain integer elements.
As stated in the proof of theorem 44 the coset congruence $\delta^{-1}.[\lceil a\delta\rceil, \lfloor b\delta\rfloor]\langle\nu\rangle$ is the integer subset of I; it is the collection

$$\delta^{-1}\lceil a\delta\rceil\langle\nu\rangle,\ \delta^{-1}(\lceil a\delta\rceil + 1)\langle\nu\rangle,\ \ldots,\ \delta^{-1}\lfloor b\delta\rfloor\langle\nu\rangle$$

of integer cosets, where δ^{-1} verifies $\delta^{-1}\delta \overset{\nu}{=} 1$. In order to find the smallest rational interval congruence with modulo $\frac{\nu}{\delta}$ containing this set of integer cosets, let us start by determining the smallest rational coset with modulo $\frac{\nu}{\delta}$ containing the integer coset $\delta^{-1}(\lceil a\delta\rceil + i)\langle\nu\rangle$, $0 \leq i \leq \lfloor b\delta\rfloor - \lceil a\delta\rceil$. It is easy to see that this rational coset is $\delta^{-1}(\lceil a\delta\rceil + i)\left\langle\frac{\nu}{\delta}\right\rangle$, which is equal[2] to $\frac{\lceil a\delta\rceil + i}{\delta}\left\langle\frac{\nu}{\delta}\right\rangle$. Since $\nu \neq 0$, $I \cap \mathbb{Z} \neq \emptyset$ and $I \cap \mathbb{Z} \neq \mathbb{Z}$, propositions 45 and 46 imply

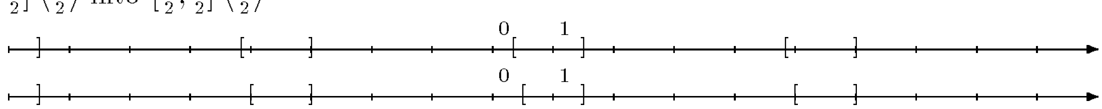
$$0 \leq \lfloor b\delta\rfloor - \lceil a\delta\rceil < \nu - 1$$

The set of representatives of the cosets of the collection $\left(\frac{\lceil a\delta\rceil + i}{\delta}\left\langle\frac{\nu}{\delta}\right\rangle\right)_{0\leq i\leq\lfloor b\delta\rfloor - \lceil a\delta\rceil}$ have the shape of aggregates of $\lfloor b\delta\rfloor - \lceil a\delta\rceil + 1$ values separated by $\frac{1}{\delta}$; the aggregates are separated from each other with a distance of $\frac{\nu}{\delta}$ following the scheme:



In order not to add new integer elements to the resulting interval congruence, its representative should not add other multiples of $\frac{1}{\delta}$ than the ones figuring in the rational coset collection and hence the smallest interval congruence containing them is $\left[\frac{\lceil a\delta\rceil}{\delta}, \frac{\lfloor b\delta\rfloor}{\delta}\right]\left\langle\frac{\nu}{\delta}\right\rangle$. $\square$

This first kind of normalization (not the one that will be finally considered) transforms $\left[\frac{1}{3}, \frac{3}{2}\right]\left\langle\frac{9}{2}\right\rangle$ into $\left[\frac{1}{2}, \frac{3}{2}\right]\left\langle\frac{9}{2}\right\rangle$



[2] For all integer i we have $\delta^{-1}\delta i \overset{\nu}{=} i$ which is equivalent to $\delta^{-1} i \overset{\frac{\nu}{\delta}}{=} \frac{i}{\delta}$.

We now state that γ selects the set of maximal concrete properties with respect to the soundness relation σ, that is here the greatest integer set contained in an abstract element (which is a coset congruence).

COROLLARY 49 (CONCRETE MAXIMALITY ASSUMPTION). *Let I be an interval congruence and C a coset congruence.*

$$C = \gamma(I) \;\Leftrightarrow \begin{cases} C \subseteq I \\ \\ \forall C' \in CC/_{\approx}\; C \subseteq C' \subseteq I \;\Rightarrow\; C' \subseteq C \end{cases} \tag{33}$$

PROOF. It results from the definition of γ as intersection with $\mathbb{Z}$. $\square$

In order to provide a unique representation of semantically equivalent abstract properties, a normalization is introduced.

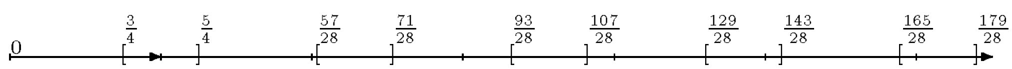
DEFINITION 50 (NORMALIZATION η). Let us define the *normalization operator* η on $IC/_{\approx^\sharp}$ by

$$\eta = \alpha \circ \gamma$$

The normalization operator replaces an abstract property by a more precise one or by a non comparable one, but without increasing the accuracy of the corresponding concrete elements. If the result is smaller than the original interval congruence, then the analysis will be more precise and, if it is non comparable, the experimentation justifies the use of such a normalization in practice. For example

$$\eta\left(\left[\frac{3}{4},\frac{5}{4}\right]\left\langle\frac{9}{7}\right\rangle\right) = \alpha\left(5.\,[1,3]\,\langle 9\rangle\right) = \left[\frac{1}{2},\frac{3}{2}\right]\left\langle\frac{9}{2}\right\rangle$$

graphically the interval congruence



is transformed into

The rational intervals not containing any integers have been removed by the normalization and the modulo has increased; this is the consequence of two processes that are part of η: the narrowing of interval bounds in order for these bounds to be in rational cosets containing integers (see proposition 48) and the choice by the normalization on CC of a particular offset (hence the increase of the modulo).

2. Abstract operators

The goal of this section is to deal with the operators on the abstract domain that are needed for the analysis. Exact meet and join algorithms are not definable since IC is not a complete lattice, hence only safe approximations of them are defined.

2.1. Conversion. As is illustrated below in the definition of the approximate join operator the only really needed conversion consists in finding the smallest interval congruence of $IC(q)$ containing a given interval congruence when the new modulo divides the one of the original congruence. For reasons that appear in the approximate join definition, the result of a conversion operation must have the new modulo (even in the degenerate cases).

DEFINITION 51 (CONVERSION TO A DIVISOR OF THE MODULO Conv). Let q' be a rational number and $I = [a, b]\langle q\rangle$ an interval congruence such that q' divides q. The *conversion* of I to modulo q' is defined by

$$\mathrm{Conv}_{q'}(I) \stackrel{\mathrm{def}}{=} \begin{cases} [a, a+q']\langle q'\rangle & \text{if } b < a \text{ and } q = 0 \text{ and } q' \neq 0 \\ [a, b]\langle q'\rangle & \text{otherwise} \end{cases}$$

This conversion algorithm is optimal in the sense that it gives the smallest interval congruence containing the original one and of given modulo.

2.2. Join. The goal of this section is to find an algorithm that determines, given two interval congruences, a minimal element containing both of them. If they are comparable, the problem has an optimal solution and will not be considered. Otherwise the interval congruences are converted to a common modulo and two different possible upper bounds are compared using the accuracy function ι on their meaning. Hence the main question is to find a minimal upper bound for two interval congruences with same modulo. Only one particular case (the interleaved relation (34)) leads to two non separable solutions and is arbitrarily solved at implementation time. The resulting join operator is not associative and a slightly different solution[3] to that latter problem would provide a commutative union but with a loss of information.

Join with constant modulo

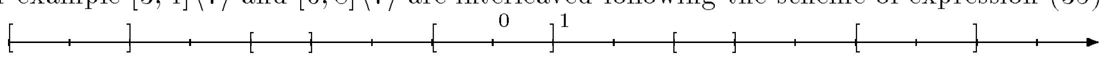
The interleaving of two interval congruences expresses the impossibility of finding a unique interval congruence containing the first one with the same common modulo and of minimal representative width (try and apply the definition below of interval-like join to one of the above examples of interleaved interval congruences).

[3]Just taking the optimum of IC to approximate this kind of union.

Definition 52 (Interleaved $\wr$). Two interval congruences $I_1 = [a_1, b_1]\langle q_1\rangle$ and $I_2 = [a_2, b_2]\langle q_2\rangle$ are said to be *interleaved*, noted $I_1 \wr I_2$, if they have the same modulo $q = |q_1| = |q_2|$ and

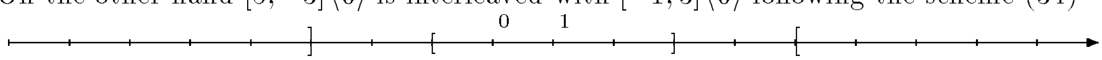
$$\begin{cases} q = 0 \ \wedge \ b_2 < a_1 \leq b_1 < a_2 \ \wedge \ a_1 + b_1 = a_2 + b_2 & (34) \\ \vee & \\ q \neq 0 \ \wedge \ 0 \leq b_1 - a_1 < q \ \wedge \ 0 \leq b_2 - a_2 < q \ \wedge \ a_2, b_2 \notin I_1 & \\ \wedge \ I_1 \not\subseteq_\sharp I_2 \ \wedge \ b_2 - a_1 \stackrel{q}{=} b_1 - a_2 & (35) \\ \vee & \\ I_2 \wr I_1 & \end{cases}$$

For example $[3,4]\langle 7\rangle$ and $[6,8]\langle 7\rangle$ are interleaved following the scheme of expression (35)



stating that the two interval congruences of non zero modulo are neither empty nor $\mathbb{Q}$, have no common elements and that, given one representative of one of them, the two nearest representatives of the other are at the same distance from the first one.

On the other hand $[5,-3]\langle 0\rangle$ is interleaved with $[-1,3]\langle 0\rangle$ following the scheme (34)



stating that the first interval congruence with zero modulo is finite when the later one is infinite; they have no common element and their bounds have the same center.

Definition 53 (Interval-like join $\sqcup_{[\,]}$). Given two non interleaved elements I_1 and I_2 of $IC(q)$, their *interval-like join* $I_1 \sqcup_{[\,]} I_2 = [\gamma, \gamma']\langle q\rangle$ is an interval congruence of modulo q containing I_1 and I_2 and of minimal value of the difference between its upper and lower bounds.

$$\forall K = [c, c']\langle q\rangle \in IC \left\{ \begin{array}{c} K \not\approx_\sharp I_1 \sqcup_{[\,]} I_2 \\ I_1 \subseteq_\sharp K \\ I_2 \subseteq_\sharp K \end{array} \right\} \Rightarrow \left\{ \begin{array}{l} K \not\subseteq_\sharp I_1 \sqcup_{[\,]} I_2 \\ c' - c > \gamma' - \gamma \end{array} \right.$$

Since when considering all the particular cases of interval congruences the only ones not providing a unique interval-like join as defined above are the interleaved ones, $\sqcup_{[\,]} : IC(q) \times IC(q) \to IC(q)$ is well defined. The existence of the interval-like join is proved by the algorithm given in appendix B.

Join to a divisor of the modulo

An alternative to the interval join $\sqcup_{[\,]}$ naturally defined for two interval congruences of same modulo is the congruence join $\sqcup_{\ldots}$ that first converts them to a divisor of the modulo following the definition 51 and then makes an interval join. The new modulo is chosen such that the converted representatives overlap.

Definition 54 (Congruence-like join $\sqcup_{\dots}$). Given two non comparable interval congruences $I_1 = [a_1, b_1]\langle q_1\rangle$ and $I_2 = [a_2, b_2]\langle q_2\rangle$ of same modulo $q = |q_1| = |q_2|$. Let r be the divisor of q that is the smallest rational closest to the distance d between I_1 and I_2 representative centers. The *congruence-like join* $I_1 \sqcup_{\dots} I_2$ is $[-\infty, +\infty]\langle 0\rangle$ if d is zero; it is defined by

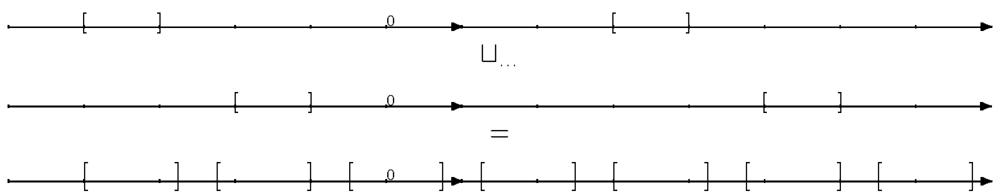
$$I_1 \sqcup_{\dots} I_2 \stackrel{\text{def}}{=} \text{Conv}_r(I_1) \sqcup_{[\,]} \text{Conv}_r(I_2)$$

if the negation of the interleaving condition (34) ($q \neq 0 \ \vee \ a_2 \leq b_1 \ \vee \ b_1 < a_1 \ \vee \ a_1 \leq b_2 \ \vee \ a_1 + b_1 \neq a_2 + b_2$) is verified and otherwise $[a_1, b_2]\langle 0\rangle$ or $[a_2, b_1]\langle 0\rangle$.

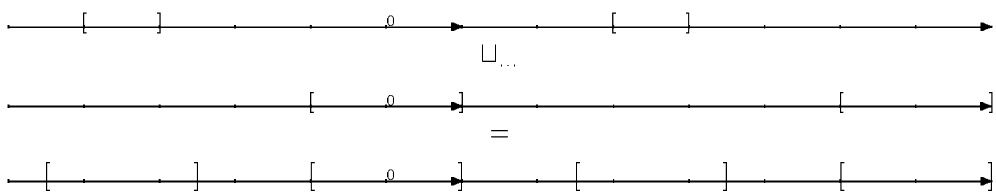
The concept of distance between two representatives denotes the smallest distance considering all possible representative pairs. Notice that if at least one of the interval congruence representative widths is infinite then the congruence-like join of the two interval congruences is $[-\infty, +\infty]\langle 0\rangle$ so that the mentioned distance between the representative centers is chosen as we want.

This kind of join is a good alternative to interval-like join for the case where the interval congruences are interleaved following expression (35). The only case we are not able to deal with is the interleaving of expression (34) where the exact join of interval congruences is approximated either by $[a_1, b_2]\langle 0\rangle$ or by $[a_2, b_1]\langle 0\rangle$ with the same precision. The following examples can be considered:

$$[3,4]\langle 7\rangle \sqcup_{\dots} [5,6]\langle 7\rangle \quad = \quad \left[\frac{5}{4}, \frac{5}{2}\right]\left\langle\frac{7}{4}\right\rangle \tag{36}$$

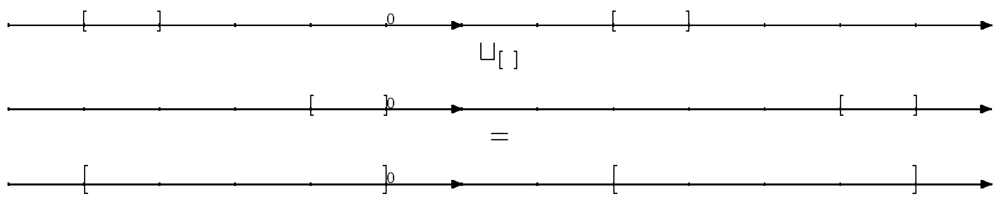


$$[3,4]\langle 7\rangle \sqcup_{\dots} [6,8]\langle 7\rangle \quad = \quad \left[\frac{5}{2}, \frac{9}{2}\right]\left\langle\frac{7}{2}\right\rangle \tag{37}$$

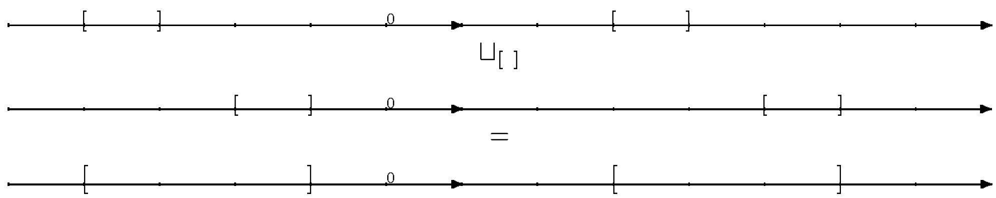


when

$$[3,4]\langle 7\rangle \sqcup_{[\,]} [6,7]\langle 7\rangle \quad = \quad [3,7]\langle 7\rangle \tag{38}$$



$$[3,4]\langle 7\rangle \sqcup_{[\,]} [5,6]\langle 7\rangle \quad = \quad [3,6]\langle 7\rangle \tag{39}$$



Intuitively comparing the examples (36) and (39), the interval join seems to be more adapted to this case, while comparing the examples (37) and (38) the congruence join seems to be closer to the exact join on rational sets. It is clear that no optimal join exists for the four examples considered above.

Precision abstract order

An operator $\downarrow$ is introduced that estimates, given two interval congruences, which one is the most informative of the two, in other words, which one contains the smallest density of integers. It is naturally defined using the accuracy function on coset congruences ι.

Definition 55 (Choice $\downarrow$). Given two interval congruences I and J, the result $I \downarrow J$ of the *choice* between I and J is one having the smallest value by $\iota \circ \gamma$.

For example

$$\left[\frac{1}{5},\frac{3}{5}\right]\left\langle\frac{8}{5}\right\rangle \downarrow \left[2,\frac{9}{2}\right]\left\langle\frac{9}{2}\right\rangle = \left[\frac{1}{5},\frac{3}{5}\right]\left\langle\frac{8}{5}\right\rangle$$

since

$$\begin{aligned}
\iota\left(\gamma\left(\left[\frac{1}{5},\frac{3}{5}\right]\left\langle\frac{8}{5}\right\rangle\right)\right) &= \iota(5.[1,3]\langle 8\rangle) \\
&= \frac{3}{8} \\
\iota\left(\gamma\left(\left[2,\frac{9}{2}\right]\left\langle\frac{9}{2}\right\rangle\right)\right) &= \iota(5.[4,9]\langle 9\rangle) \\
&= \frac{2}{3}
\end{aligned}$$

The reader can easily see that this precision order confirms the intuitive preferences between interval and congruence-like join at the end of the preceding paragraph.

Approximate least upper bound

Finally we get the following approximation of the least upper bound operator (the one defined on $\mathbb{P}(\mathbb{Q})$) on IC:

Definition 56 (Approximate join $\sqcup$). Given $I_1 = [a_1, b_1]\langle q_1\rangle$ and $I_2 = [a_2, b_2]\langle q_2\rangle$ two interval congruences, their *approximate* join $I_1 \sqcup I_2$ is equal to

$$\begin{cases} \quad I_1 & \text{if} \quad I_2 \subseteq_\sharp I_1 \\ \text{else} \quad I_2 & \text{if} \quad I_1 \subseteq_\sharp I_2 \\ \text{else} \quad I'_1 \sqcup_{\dots} I'_2 & \text{if} \quad I'_1 \wr I'_2 \\ \text{else} \quad (I'_1 \sqcup_{[\,]} I'_2) \downarrow (I'_1 \sqcup_{\dots} I'_2) & \end{cases}$$

where $I'_1 = \mathrm{Conv}_{\gcd(q_1,q_2)}(I_1)$ and $I'_2 = \mathrm{Conv}_{\gcd(q_1,q_2)}(I_2)$.

Of course, it is possible to refine this definition, especially in the case where the choice between the congruence and the interval joins is arbitrary (the accuracy of their concretizations are equal).

Let us look at a necessary refinement of the least upper bound that has to do with the initialization of the iteration process during the analysis. During the analysis of the program

```
      x := 1;
{1:}  while true do
{2:}     x := x + 3;
{3:}  od;
{4:}
```

it is determined at the first iteration and program point {2:} that `x` may be equal to 1, the second iteration indicates that `x` may be equal to 1 or to 4 hence resulting in the abstract join of $[1,1]\langle 0\rangle$ and $[4,4]\langle 0\rangle$. Following our definition, this join result in $[1,1]\langle 3\rangle$ and corresponds to what we expected. Nevertheless, it might be not always the case that the approximate join determines at the first iteration which of the two strategies is preferably chosen. The solution is to keep during a small number n of iterations the two join alternatives and then choosing among the resulting 2^n interval congruences with the choice operator.

2.3. Intersection. The goal of this section is to find an algorithm that determines, given two interval congruences, a minimal element containing their exact intersection. If they are comparable the problem has an optimal solution and will not be considered. Otherwise the interval congruences are converted to a common modulo and two different possible upper bounds are compared using the accuracy function ι on their meaning. Hence the main question is to find a minimal upper bound of the intersection of two interval congruences with same modulo. Only one particular case (the overlap relation (40)) leads to two non separable solutions and is arbitrarily solved at implementation time. This approximate intersection operator is not associative and a slightly different solution[4] to that latter problem would

[4]Just taking the optimum of IC to approximate this kind of intersection

provide a commutative intersection but with a loss of information.

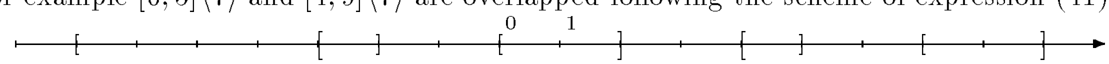
Intersection with constant modulo

The overlapping of two interval congruences expresses the impossibility of finding a unique interval congruence contained in the first ones with the same common modulo and of minimal representative width.

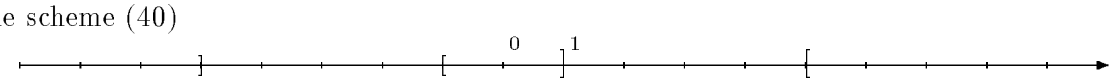
Definition 57 (Overlap $\sim$). Let $I_1 = [a_1, b_1]\langle q_1\rangle$ and $I_2 = [a_2, b_2]\langle q_2\rangle$ be two interval congruences, I_1 and I_2 *overlap*, which is noted $I_1 \sim I_2$ if they have the same modulo $q = |q_1| = |q_2|$ and

$$\left\{\begin{array}{ll} \quad q = 0 \ \wedge \ b_2 < a_2 \leq b_1 < a_1 \ \wedge \ a_2 + b_1 = a_1 + b_2 & (40) \\ \vee & \\ \quad q \neq 0 \ \wedge \ 0 \leq b_1 - a_1 < q \ \wedge \ 0 \leq b_2 - a_2 < q \ \wedge \ a_2, b_2 \in I_1 & \\ \quad \wedge \ I_2 \not\subseteq_\sharp I_1 \ \wedge \ b_2 - a_2 = b_1 - a_1 & (41) \\ \vee & \\ \quad I_2 \sim I_1 & \end{array}\right.$$

For example $[0, 5]\langle 7\rangle$ and $[4, 9]\langle 7\rangle$ are overlapped following the scheme of expression (41)



stating that the two interval congruences of non zero modulo are neither empty nor $\mathbb{Q}$, have common elements and that each representative of one of them intersects two distinct representatives of the other one. On the other hand $[-1, -5]\langle 0\rangle$ is overlapped with $[5, 1]\langle 0\rangle$ following the scheme (40)



stating that the two interval congruences with zero modulo are infinite, their join is $\mathbb{Q}$ and have the same representative width.

Definition 58 (Interval-like intersection $\sqcap_{[\,]}$). Given two non overlapped non comparable elements I_1 and I_2 of $IC(q)$, their *interval-like intersection* $I_1 \sqcap_{[\,]} I_2 = [\gamma, \gamma']\langle q\rangle$ is an interval congruence of modulo q containing the elements common to I_1 and I_2 and of minimal representative width $\gamma' - \gamma$.

$$\forall K = [c, c']\langle q\rangle \in IC \left\{\begin{array}{c} K \not\approx_\sharp I_1 \sqcap_{[\,]} I_2 \\ K \subseteq_\sharp I_1 \\ K \subseteq_\sharp I_2 \end{array}\right\} \Rightarrow \left\{\begin{array}{l} K \subseteq_\sharp I_1 \sqcap_{[\,]} I_2 \\ c' - c > \gamma' - \gamma \end{array}\right.$$

Since when considering all the particular cases of interval congruences the only ones not providing a unique interval-like intersection as defined above are the overlapped ones, $\sqcap_{[\,]} : IC(q) \times IC(q) \to IC(q)$ is well defined. The existence of the interval-like intersection is proved using its defining algorithm given in appendix C.

Intersection to a divisor of the modulo

An alternative to the interval intersection $\sqcap_{[\,]}$ naturally defined for two interval congruences of same modulo is the congruence intersection $\sqcap_{\dots}$ that first reduces the representative safely with respect to the exact intersection and then makes a congruence-like join which is safe with regard to exact intersection too.

Definition 59 (Congruence-like intersection $\sqcap_{\dots}$). Given two non comparable interval congruences $I_1 = [a_1, b_1]\langle q_1\rangle$ and $I_2 = [a_2, b_2]\langle q_2\rangle$ of same modulo $q = |q_1| = |q_2|$, then the *congruence-like intersection* $\sqcap_{\dots}$ is defined by

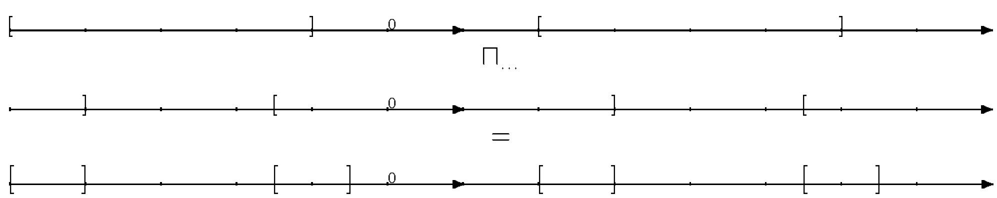
$$I_1 \sqcap_{\dots} I_2 \stackrel{\text{def}}{=} [a_1, b_2']\langle q\rangle \sqcup_{\dots} [a_2, b_1']\langle q\rangle$$

where b_1' (resp. b_2') is the smallest element of $\{b_1 + kq, k \in \mathbb{Z}\}$ (resp.$\{b_2 + kq, k \in \mathbb{Z}\}$) greater than a_2 (resp. a_1).

Like the interval-like intersection, the congruence-like intersection is a safe approximation of exact set intersection.

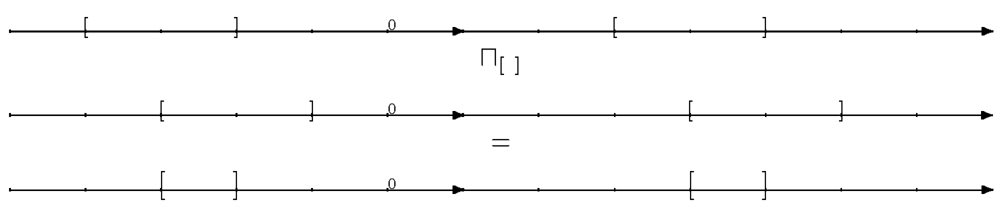
This kind of intersection is a good alternative to interval-like intersection for the case where the interval congruences are overlapped following expression (41). The only case we are not able to deal with is the overlap of expression (40) where the exact intersection of the interval congruences is either approximated by $[a_2, b_2]\langle 0\rangle$ or by $[a_1, b_1]\langle 0\rangle$ with the same precision. The following examples are considered:

$$\begin{aligned}[2,6]\langle 7\rangle \sqcap_{\dots} \left[\frac{11}{2}, 10\right]\langle 7\rangle &= [2,3]\langle 7\rangle \sqcup_{\dots} \left[\frac{11}{2}, 6\right]\langle 7\rangle \\ &= [2,3]\left\langle \frac{7}{2}\right\rangle\end{aligned}$$



provides only an approximation of the intersection on $\mathbb{P}(\mathbb{Q})$, while

$$[3,5]\langle 7\rangle \sqcap_{[\,]} [4,6]\langle 7\rangle = [4,5]\langle 7\rangle$$



corresponds to the exact intersection and hence is optimal.

Intuitively, the purpose of defining such intersection algorithms is to provide a more accurate approximation than just choosing one of the original interval congruences. As for the join operator, the two algorithms are complementary and are used in different situations (using the non adequate algorithm on the examples given above would only result in a loss of precision on the result).

Approximate greatest lower bound

Finally we get the following approximation of the greatest lower bound operator (the one defined on $\mathbb{P}(\mathbb{Q})$) on IC:

Definition 60 (Approximate intersection $\sqcap$). Let $I_1{=}[a_1, b_1]\langle q_1\rangle$ and $I_2{=}[a_2, b_2]\langle q_2\rangle$ be two interval congruences. Their *approximate intersection* $I_1 \sqcap I_2$ is equal to

$$\begin{cases} \quad\quad I_2 & \text{if} \quad I_2 \subseteq_\sharp I_1 \\ \text{else} \quad I_1 & \text{if} \quad I_1 \subseteq_\sharp I_2 \\ \text{else} \quad I'_1 \sqcap_{\dots} I'_2 & \text{if} \quad I'_1 \sim I'_2 \\ \text{else} \quad (I'_1 \sqcap_{[\,]} I'_2) \downarrow (I'_1 \sqcap_{\dots} I'_2) \downarrow I_1 \downarrow I_2 & \end{cases}$$

where $I'_1 = [a_1 + k_1 q_1, b_1 + (k_1 + l_1 - 1)q_1]\langle q\rangle$ and $I'_2 = [a_2 + k_2 q_2, b_2 + (k_2 + l_2 - 1)q_2]\langle q\rangle$ and $q = \mathrm{lcm}(q_1, q_2) = l_1 q_1 = l_2 q_2$, k_1 and k_2 are integers minimizing the value of $|a_1 + b_1 - a_2 - b_2 - q_1 + q_2 + q + 2(k_1 q_1 - k_2 q_2)|$.

The rather complex choice of I'_1 and I'_2 in the last definition simply is the expression of the conversion of I_1 and I_2 to a common modulo $\mathrm{lcm}(q_1, q_2)$ where the distance between their representative is as important as possible (hence the minimization of $|a_1 + b_1 - a_2 - b_2 - q_1 + q_2 + q + 2(k_1 q_1 - k_2 q_2)|$).

Of course, it is possible to refine this definition, especially in the case where the choice between the operands, the congruence and the interval intersections is arbitrary (the accuracy of their concretizations are equal).

2.4. Widening operator. Recall from [CC92b] that the three uses of widening operator are the following:

(1) A sound choice function, that is if a concrete property is soundly approximated by many abstract values the widening operator discriminates between all possibilities,
(2) A way to ensure convergence,
(3) An accelerator to guarantee rapid termination of the iteration process for fixpoint computation.

The first feature is part of the definition of the abstraction function α when the two last ones are explicated in the following operator derived from the widening operators on interval [CC76] and rational arithmetical cosets [Gra91a].

Definition 61 (Widening ∇). Let $I_1 = [a_1, b_1]\langle q_1\rangle$ and $I_2 = [a_2, b_2]\langle q_2\rangle$ be two interval congruences. Their *widening* $I_1 \nabla I_2$ is defined by

$$\begin{cases} \left[\frac{\lceil a_2\delta\rceil - 1}{\delta}, \frac{\lfloor b_2\delta\rfloor + 1}{\delta}\right]\left\langle \frac{\nu}{\delta}\right\rangle & \text{if} \quad q_1 = q_2 = \frac{\nu}{\delta} \neq 0 & (42) \\ [a_2, a_2 + q_2]\langle q_2\rangle & \text{if} \quad 0 \neq q_1 \neq q_2 \neq 0 & (43) \\ [a, b]\langle 0\rangle & \text{if} \quad \begin{cases} q_1 = q_2 = 0 \\ b_1 \geq a_1 \ \wedge \ b_2 \geq a_2 \end{cases} & (44) \\ I_1 \sqcup I_2 & \text{otherwise} & (45) \end{cases}$$

where if $a_2 < a_1$ then $a = -\infty$ else $a = a_1$ and if $b_1 < b_2$ then $b = +\infty$ else $b = b_1$.

Notice that in order to be more precise than a sign analysis, the widening on two finite rational intervals only has to jump to zero before extrapolating the infinite values if the infinite extrapolation value is not of the same sign as the original one. This additional feature does not figure in the widening definition as a matter of simplification.

The correctness of ∇ is a direct consequence of the correctness of classical widenings on intervals (case (44)) and rational arithmetical congruences (case (43) where moreover a particular interval congruence representing $\mathbb{Q}$ is chosen for technical reasons). In addition, it is sufficient to remark that

- the situation where q_1 is zero and q_2 is not (case (45)) has not to be considered since it cannot take place in an infinite increasing chain: an interval congruence of zero modulo must be of infinite width in order to be greater than an interval congruence of non zero modulo which in turn is greater than an interval congruence of null modulo only if the latter one is of finite width. Hence an infinite increasing chain containing interval congruence of null modulo will necessarily contain two consecutive such elements.
- in the case where the two original interval congruences have the same non zero modulo (case (42)), the widening ensures convergence in finite time by embedding the representative in a new one adding integer cosets in the corresponding coset congruence hence accelerating the termination of the iteration process.

First recall the classical widening operator used on intervals with the following examples:

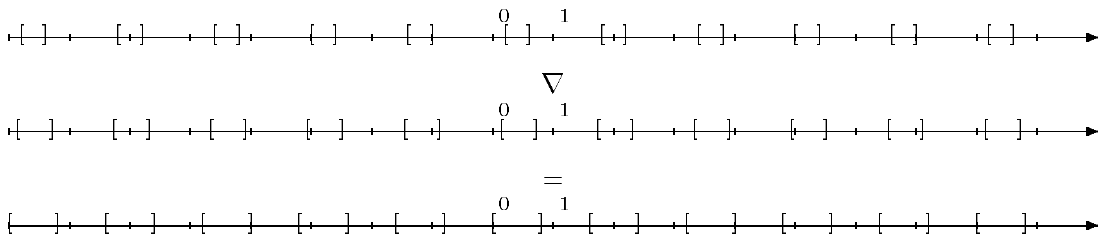
$$\begin{aligned} [2,3]\langle 0\rangle \nabla [2,7]\langle 0\rangle &= [2,+\infty]\langle 0\rangle \\ [3,10]\langle 0\rangle \nabla [1,10]\langle 0\rangle &= [0,10]\langle 0\rangle \\ [-10,-3]\langle 0\rangle \nabla [-10,3]\langle 0\rangle &= [-10,+\infty]\langle 0\rangle \end{aligned}$$

Then the congruence-like behavior of our widening operator is illustrated by:

$$\left[\frac{2}{3}, 6\right]\left\langle\frac{190}{77}\right\rangle \nabla \left[\frac{1}{3}, 1\right]\langle 5\rangle \quad = \quad \left[\frac{1}{3}, \frac{16}{3}\right]\langle 5\rangle \approx_\sharp [-\infty, +\infty]\langle 0\rangle$$

and finally the last kind of widening process (apart from the approximate join operator) is examplified in:

$$\left[\frac{1}{5},\frac{3}{5}\right]\left\langle\frac{8}{5}\right\rangle \nabla \left[\frac{1}{7},\frac{5}{7}\right]\left\langle\frac{8}{5}\right\rangle \quad = \quad \left[0,\frac{4}{5}\right]\left\langle\frac{8}{5}\right\rangle$$



where at most three more applications of ∇ lead to an interval congruence containing $\mathbb{Z}$ (look at the respective meaning of the originals and resulting interval congruences).

The widening operator is improvable by a slight modification of case (42). Instead of widening both of the interval bounds, the operator might modify only one of them; this is especially recommended when the other bound is the same in I_1 and in I_2. An other alternative to case (44) is enabled by the duality of the interval congruence model. Indeed, instead of keeping a zero modulo, a non zero is possibly introduced depending on program parameters.

3. Abstract primitives

Defining first abstractions of integer sum and product by a constant allows us to deal with assignments of affine expressions to integer variables. Then abstracting a given class of tests gives the possibility to take into account control flow information in the analysis. The entire design of an abstract interpretation requires also the definition of backward abstract primitives to deal with backward analysis and improve the accuracy of the resulting combimation of forward and backward analyses. Those primitives are easily deduced from their interval and congruence counterparts.

The following abstract primitives are chosen to be sound, i.e. if F is the concrete primitive and ϕ the abstract one, we have $F \leq \gamma \circ \phi \circ \alpha$.

3.1. Abstract sum.

Definition 62 (Abstract sum $\oplus$). Let $[a_1, b_1]\langle q_1\rangle$ and $[a_2, b_2]\langle q_2\rangle$ be two interval congruences, their *abstract sum*, noted $[a_1, b_1]\langle q_1\rangle \oplus [a_2, b_2]\langle q_2\rangle$, is $[1, 0]\langle 1\rangle$ if the concretization of one operand is the empty set and otherwise is defined by

$$\begin{cases} \eta\left([a_1 + a_2, b_1 + b_2]\langle \gcd(q_1, q_2)\rangle\right) & \text{if } \ q_i \neq 0 \vee a_i \leq b_i, \ i \in \{1, 2\} \\ [0, 0]\langle 1\rangle & \text{otherwise} \end{cases}$$

Proof. [of the soundness of $\oplus$] We need to prove that the abstract sum is safe, that is for every interval congruence I_1 and I_2

$$\gamma(I_1) + \gamma(I_2) \subseteq \gamma(I_1 \oplus I_2)$$

The result is trivial either if an operand has an empty meaning or if at least one operand is of null modulo with its lower bound greater than its upper bound. Suppose we are not in this case and show that

$$I_1 + I_2 \subseteq I_1 \oplus I_2$$

$x_1 \in I_1$ (resp. $x_2 \in I_2$) if and only if $x_1 = \alpha_1 + k_1 q_1$ (resp. $x_2 = \alpha_2 + k_2 q_2$) where $a_1 \leq \alpha_1 \leq b_1$ and $k_1 \in \mathbb{Z}$ (resp. $a_2 \leq \alpha_2 \leq b_2$ and $k_2 \in \mathbb{Z}$). Hence $x_1 + x_2 = \alpha_1 + \alpha_2 + (k_1 q_1' + k_2 q_2') \gcd(q_1, q_2)$ where $q_1 = q_1' \gcd(q_1, q_2)$ and $q_2 = q_2' \gcd(q_1, q_2)$. The mentioned inclusion of interval congruences follows. Then we have $(I_1 + I_2) \cap \mathbb{Z} \subseteq (I_1 \oplus I_2) \cap \mathbb{Z}$ and since $(I_1 \cap \mathbb{Z}) + (I_2 \cap \mathbb{Z}) \subseteq (I_1 + I_2) \cap \mathbb{Z}$ and γ is the intersection with $\mathbb{Z}$, the correctness is established. □

Notice that the definition of abstract sum is commutative which seems natural; unfortunately the abstract sum is not exact, i.e. generally $I_1 + I_2 \subset I_1 \oplus I_2$ and $I_1 \oplus I_2$ is not the smallest interval congruence containing $I_1 + I_2$; Generally, the smallest interval congruence containing $I_1 + I_2$ does not exist.

Examples
First illustrating the else branch of the definition, take

$$[2,-2]\langle 0\rangle \oplus [4,7]\langle 54\rangle = [0,0]\langle 1\rangle$$

Then an example of non zero modulo sum

$$\left[\frac{3}{2},\frac{9}{2}\right]\langle 14\rangle \oplus \left[\frac{11}{2},7\right]\langle 21\rangle = \eta\left(\left[0,\frac{9}{2}\right]\langle 7\rangle\right) = [0,4]\langle 7\rangle$$

where it is visible that normalizing the operands before doing the abstract sum would have led to a more precise result ($[1,4]\langle 7\rangle$) by not accumulating "errors" on the bounds of the interval congruences. That is why the results of the abstract statements (abstract expressions) are normalized.

3.2. Abstract product by an integer.

DEFINITION 63 (ABSTRACT PRODUCT $\odot$). Given an integer λ and an interval congruence $I = [a,b]\langle q\rangle$, their *abstract product* is $[1,0]\langle 1\rangle$ if the meaning of I is empty and otherwise

$$\lambda \odot [a,b]\langle q\rangle \stackrel{\text{def}}{=} \begin{cases} \eta([\lambda a, \lambda b]\langle \lambda q\rangle) & \text{if} \quad \lambda > 0 \\ [0,0]\langle 0\rangle & \text{if} \quad \lambda = 0 \\ \eta([\lambda b, \lambda a]\langle \lambda q\rangle) & \text{if} \quad \lambda < 0 \end{cases}$$

PROOF. [of the soundness of $\odot$] We need to prove that the abstract product is safe, that is for every interval congruence I

$$\lambda * \gamma(I) \subseteq \gamma(\lambda \odot I)$$

Cases where λ is zero or where the interval congruence meaning is empty are straightforward. Suppose $I = [a,b]\langle q\rangle$ and λ is strictly positive (and $\gamma(I) \neq 1.[1,0]\langle 1\rangle$), then $\lambda * ([a,b]\langle q\rangle)$ is equal to $[\lambda a, \lambda b]\langle \lambda q\rangle$ and $\lambda * \gamma(I) = \lambda * (I \cap \mathbb{Z}) \subseteq (\lambda * I) \cap \mathbb{Z} = \gamma(\eta([\lambda a, \lambda b]\langle \lambda q\rangle))$, since $\gamma \circ \eta = (\gamma \circ \alpha) \circ \gamma = \gamma$. The case where $\lambda < 0$ has a similar solution. □

Examples
$-2 \odot [-\infty, 5]\langle 0\rangle = [-10, +\infty]\langle 0\rangle$ while $2 \odot [2,4]\langle 6\rangle = [4,8]\langle 12\rangle$.
Other arithmetical abstract primitives could be defined such as product by a rational, modulo and euclidian division. But since only very special cases would lead to accurate results[5] and the other cases would be long, simple and not very useful (in a first approximation) to define, they are not given here.

[5]Think of $[a,b]\langle q\rangle$ *amod* r where *amod* is the abstract modulo function and r divides q, then $[a,b]\langle 0\rangle$ is a good approximation of the exact result.

3.3. Abstract test. The definition of the abstraction of the test statement is usually divided into two steps. First tests involving conditional expressions expressed by the approximate invariants of the analysis (here interval congruences) are considered. Then more general conditional expressions are safely approximated and the first step is applied.

DEFINITION 64 (ABSTRACT TEST WITH AN ARCEBR CONDITION). Let $I_1 = [a_1, b_1]\langle q_1\rangle$ be an abstract context preceding a test with the condition equation $x \equiv [a_2, b_2] \mod q_2$. The abstract entry context in the true branch of the conditional is

$$I_1 \sqcap [a_2, b_2]\langle q_2\rangle$$

while the abstract entry context in the false branch of the conditional is

$$I_1 \sqcap \alpha(\overline{\gamma([a_2, b_2]\langle q_2\rangle)})$$

PROOF. [of the soundness] Since for all interval congruences I and J in CC, $I \cap J \subseteq I \sqcap J$ and $\gamma(I) \cap \overline{\gamma(J)} = (I \cap \mathbb{Z}) \cap (\alpha(\overline{\gamma(J)}) \cap \mathbb{Z}) \subseteq \left(I \sqcap \alpha(\overline{\gamma(J)})\right) \cap \mathbb{Z} = \gamma\left(I \sqcap \alpha(\overline{\gamma(J)})\right)$, this abstract test is correct. $\square$

Notice that the test condition is easily extended to an equivalent linear equation by first approximating it with an arithmetical rational congruence equation with bounded representative.

A major improvement with respect to the existing analyses using congruence properties on integers is that the negation of the natural condition (here an arithmetical rational congruence equation with bounded representative) is also quite natural. Recall that the meaning of a rational interval congruence is its integer points.

3.4. Precision ordering with the related analyses. Though the operators on the set of interval congruences are inspired by the corresponding ones on the lattices of intervals and cosets, the resulting analysis is not comparable with these two. Let us have a look for example at the approximate join operator. On the example

$$[2,4]\langle 0\rangle \sqcup [3,6]\langle 0\rangle \quad = \quad [2,6]\langle 0\rangle$$

the join operator has the same behavior as (is as precise as) the one of the lattice of intervals while on the example

$$[-4,-3]\langle 0\rangle \sqcup [3,4]\langle 0\rangle \quad = \quad [3,4]\langle 7\rangle$$

they are clearly non comparable ($[-4,-3] \cup [3,4] = [-4,4]$). The same feature results from the consideration of the example

$$[0,0]\langle 29\rangle \sqcup [4,4]\langle 29\rangle \quad = \quad \left[4, \frac{29}{7}\right]\left\langle \frac{29}{7}\right\rangle$$

the concretization of which is $21.[28,29]\langle 29\rangle$ which is more precise than $\mathbb{Z}$ (the result of the application of the join operator on integer cosets) while

$$[0,0]\langle 30\rangle \sqcup [12,12]\langle 30\rangle \quad = \quad [0,2]\langle 10\rangle$$

which is clearly non comparable with the result of the same operation on the lattice of integer cosets. The same kind of behavior results from the definition of the other abstract operators and abstract statements.

point	initially	first iteration
{1:}	$(1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$	$(1.[0,0]\langle 0\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$
{2:}	$(1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$	$(1.[0,0]\langle 0\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$
{3:}	$(1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$	$(1.[0,0]\langle 0\rangle, 1.[0,0]\langle 0\rangle, 1.[1,0]\langle 1\rangle)$
{4:}	$(1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$	$(1.[0,0]\langle 0\rangle, 1.[0,0]\langle 0\rangle, 1.[1,0]\langle 1\rangle)$
{5:}	$(1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$	$(1.[0,0]\langle 0\rangle, 1.[0,0]\langle 0\rangle, 1.[1,1]\langle 0\rangle)$
{6:}	$(1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$	$(1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$
{7:}	$(1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$	$(1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$
{8:}	$(1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$	$(1.[0,0]\langle 0\rangle, 1.[0,0]\langle 0\rangle, 1.[1,1]\langle 0\rangle)$
{9:}	$(1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$	$(1.[1,1]\langle 0\rangle, 1.[0,0]\langle 0\rangle, 1.[1,1]\langle 0\rangle)$
{10:}	$(1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$	$(1.[0,1]\langle 0\rangle, 1.[0,0]\langle 0\rangle, 1.[0,1]\langle 0\rangle)$

point	second iteration	third iteration
{1:}	$(1.[0,0]\langle 0\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$	$(1.[0,0]\langle 0\rangle, 1.[1,0]\langle 1\rangle, 1.[1,0]\langle 1\rangle)$
{2:}	$(1.[0,+\infty]\langle 0\rangle, 1.[0,0]\langle 0\rangle, 1.[0,+\infty]\langle 0\rangle)$	$(1.[0,0]\langle 1\rangle, 1.[0,0]\langle 3\rangle, 1.[0,1]\langle 6\rangle)$
{3:}	$(1.[0,+\infty]\langle 0\rangle, 1.[0,0]\langle 3\rangle, 1.[0,+\infty]\langle 0\rangle)$	$(1.[0,0]\langle 1\rangle, 1.[0,0]\langle 3\rangle, 1.[0,1]\langle 6\rangle)$
{4:}	$(1.[0,0]\langle 2\rangle, 1.[0,0]\langle 3\rangle, 1.[0,+\infty]\langle 0\rangle)$	$(1.[0,0]\langle 2\rangle, 1.[0,0]\langle 3\rangle, 1.[0,1]\langle 6\rangle)$
{5:}	$(1.[0,0]\langle 2\rangle, 1.[0,0]\langle 3\rangle, 1.[1,1]\langle 6\rangle)$	$(1.[0,0]\langle 2\rangle, 1.[0,0]\langle 3\rangle, 1.[1,1]\langle 6\rangle)$
{6:}	$(1.[1,1]\langle 2\rangle, 1.[0,0]\langle 3\rangle, 1.[0,+\infty]\langle 0\rangle)$	$(1.[1,1]\langle 2\rangle, 1.[0,0]\langle 3\rangle, 1.[0,1]\langle 6\rangle)$
{7:}	$(1.[1,1]\langle 2\rangle, 1.[0,0]\langle 3\rangle, 1.[0,0]\langle 6\rangle)$	$(1.[1,1]\langle 2\rangle, 1.[0,0]\langle 3\rangle, 1.[0,0]\langle 6\rangle)$
{8:}	$(1.[0,0]\langle 1\rangle, 1.[0,0]\langle 3\rangle, 1.[0,1]\langle 6\rangle)$	$(1.[0,0]\langle 1\rangle, 1.[0,0]\langle 3\rangle, 1.[0,1]\langle 6\rangle)$
{9:}	$(1.[0,0]\langle 1\rangle, 1.[0,0]\langle 3\rangle, 1.[0,1]\langle 6\rangle)$	$(1.[0,0]\langle 1\rangle, 1.[0,0]\langle 3\rangle, 1.[0,1]\langle 6\rangle)$
{10:}	$(1.[0,0]\langle 1\rangle, 1.[0,0]\langle 3\rangle, 1.[0,1]\langle 6\rangle)$	$(1.[0,0]\langle 1\rangle, 1.[0,0]\langle 3\rangle, 1.[0,1]\langle 6\rangle)$

TABLE IV.1. Example of iteration process

3.5. Example. Let us consider the following program

```
{1:}     i := 0;
{2:}     while test_on_i do
{3:}       x := 3*i;
{4:}       if even(i) then
{5:}         y := 3*i + 1
{6:}       else
{7:}         y := 3*i + 3
{8:}       endif;
{9:}       A[x,y] := A[x+1,y+1] + A[x+2,y+2];
           i := i + 1
{10:}    endwhile;
```

where `i`,`x` and `y` are integer variables, `A` an array of dimension 2 and `test_on_i` a boolean expression that is not taken into account by the analysis.

The analyzed program, instead of being very complex or requiring all the subtilities of the interval congruence analysis, illustrates the basic idea of our analysis. The exact information to approximate in this program is congruence like, but not quite, since the test inserted in the loop makes it fail; only interval congruences can take this information into account.

The iteration process is summarized in table IV.1. In this table : (I, X, Y) at line {n:} and in column "i $^{\text{th}}$ iteration" stands for: during iteration i at program point {n:} the values of i, x and y are approximated respectively by the integer sets I, X and Y. The safe static approximation is given in the last column where the fixed point is reached. Each element of the represented tuples stands for the meaning uniquely associated with the corresponding abstract interval congruence in the iteration process. The normalization operator is essential here to describe the analysis results.

The iteration starts without knowing anything about the variables as it is stated in the "initially" column. Then the abstract primitives and the widening are used to determine the other columns values. Notice that the congruence behavior of the widening is preferred at point {2:} (it detects that {i} is in fact the loop index) when the interval behavior is preferably chosen at points {8:} and {10:}. The fourth iteration giving the same results as the third one (telling the analyzer that the fixpoint is reached) is done by the analyzer but is not represented here.

The important result of analyzing this program with interval congruences is that the three references to the array A are shown to be independent. It is easy to see that

$$\begin{aligned}
1.[0,0]\langle 3\rangle \times 1.[0,1]\langle 6\rangle \;\cap\; 1.[1,1]\langle 3\rangle \times 1.[1,2]\langle 6\rangle &= \emptyset \\
1.[0,0]\langle 3\rangle \times 1.[0,1]\langle 6\rangle \;\cap\; 1.[2,2]\langle 3\rangle \times 1.[2,3]\langle 6\rangle &= \emptyset \\
1.[1,1]\langle 3\rangle \times 1.[1,2]\langle 6\rangle \;\cap\; 1.[2,2]\langle 3\rangle \times 1.[2,3]\langle 6\rangle &= \emptyset
\end{aligned}$$

APPENDIX B

Interval-like join algorithm

Given $I_1 = [a_1, b_1]\langle q\rangle$ and $I_2 = [a_2, b_2]\langle q\rangle$ two non interleaved non comparable interval congruences, their interval-like join is determined as follows:

```
if q = 0 then
    if a1 ≤ b1 then
        if a2 ≤ b2 then [min(a1, a2), max(b1, b2)] ⟨0⟩
        else
            if b2 ≤ b1 < a2 then
                if a1 ≤ b2 then [a2, b1] ⟨0⟩
                else
                    if a2 − b1 > a1 − b2 then [a2, b1] ⟨0⟩
                    if a2 − b1 < a1 − b2 then [a1, b2] ⟨0⟩
            if b1 ≥ a2 then
                if a1 ≤ b2 then [−∞, +∞] ⟨0⟩
                else [a1, b2] ⟨0⟩
    else
        if a2 ≥ b2 then call the algorithm with permuted parameters
        else
            if a2 ≤ b1 then [−∞, +∞] ⟨0⟩
            if b1 < a2 < a1 ∧ b2 ≤ b1 then [a2, b1] ⟨0⟩
            if a2 ≥ a1 then
                if b1 < b2 < a1 then [a1, b2] ⟨0⟩
                if b2 ≥ a1 then [−∞, +∞] ⟨0⟩
else
    if a2 ∈ I1 then
        if b2 ∈ I1 then [a1, a1 + q] ⟨q⟩
        else U1
    else
        if b2 ∈ I1 then U2
        else
            if l(U1) > l(U2) then U2
            if l(U1) < l(U2) then U1
```

where $U_1 = [a_1, \min_{k\in\mathbb{Z}}\{b_2 + kq \geq a_1\}]\langle q\rangle$ and $U_2 = [a_2, \min_{k\in\mathbb{Z}}\{b_1 + kq \geq a_2\}]\langle q\rangle$ and $l([a, b]\langle q\rangle) = b - a$. Notice that all the missing cases correspond to comparable or interleaved interval congruences.

APPENDIX C

Interval-like intersection algorithm

Given $I_1 = [a_1, b_1]\langle q\rangle$ and $I_2 = [a_2, b_2]\langle q\rangle$ two non overlapped non comparable interval congruences, their interval-like intersection is determined as follows:

```
if q = 0 then
        if a₁ ≤ b₁ then
                if a₂ ≤ b₂ then
                        if max(a₁, a₂) ≤ min(b₁, b₂) then [max(a₁, a₂), min(b₁, b₂)] ⟨0⟩
                        else [1, 0] ⟨1⟩
                else
                        if b₂ ≤ b₁ < a₂ then
                                if a₁ ≤ b₂ then [a₁, b₂] ⟨0⟩
                                else [1, 0] ⟨1⟩
                        if b₁ ≥ a₂ then
                                if a₁ ≤ b₂ then I₁
                                else [a₂, b₁] ⟨0⟩
        else
                if a₂ ≥ b₂ then call the algorithm with permutted parameters
                else
                        if a₂ ≤ b₁ then I₁ ↓ I₂
                        if b₁ < a₂ < a₁ ∧ b₂ ≤ b₁ then [a₁, b₂] ⟨0⟩
                        if a₂ ≥ a₁ then [a₂, b₁] ⟨0⟩
else
        if a₂ ∈ I₁ then
                if b₂ ∈ I₁ then I₁ ↓ I₂
                else U₂
        else
                if b₂ ∈ I₁ then U₁
                else [1, 0] ⟨1⟩
```

where $U_1 = [a_1, \min_{k\in\mathbb{Z}}\{b_2 + kq \geq a_1\}]\langle q\rangle$ and $U_2 = [a_2, \min_{k\in\mathbb{Z}}\{b_1 + kq \geq a_2\}]\langle q\rangle$. Notice that all the missing cases correspond to comparable or overlapped interval congruences.

Part 3

SEMANTIC ANALYSIS OF TRAPEZOÏD CONGRUENCES

CHAPTER V

DESIGN OF A RATIONAL RELATIONAL MODEL

The analysis of trapezoid congruences requires two different domains: a first one of integer properties, for precision, and a second one of rational properties, for the efficiency of its basic algorithms. Although the relational coset congruence domain is presented before the trapezoid congruence one, we see in Chapter VI that the integer relational coset congruences are naturally deduced from the rational trapezoid congruences. The content of this chapter and the next one corresponds to a revision of [Mas92].

1. Notations

The notations of Chapter II are used. In addition, we need to extend some notations to rational intervals.

Definition 65 (Rational interval linear combination). Let $I_1 = [a_1, b_1]$ and $I_2 = [a_2, b_2]$ be two rational intervals of possibly positive infinite upper bound and possibly negative infinite lower bound and ρ a rational number. The sum of intervals, their product and sum with a constant are defined by

$$\rho + I_1 \stackrel{\text{def}}{=} [a_1 + \rho, b_1 + \rho]$$

$$\rho * I_1 \stackrel{\text{def}}{=} \begin{cases} [\rho a_1, \rho b_1] & \text{if } \rho \geq 0 \\ [\rho b_1, \rho a_1] & \text{otherwise} \end{cases}$$

$$I_1 + I_2 \stackrel{\text{def}}{=} [a_1 + a_2, b_1 + b_2]$$

The dot product is extended to deal with vectors of rational intervals

$$(\rho_1, \rho_2, \ldots, \rho_n).\begin{pmatrix} [a_1, b_1] \\ [a_2, b_2] \\ \vdots \\ [a_n, b_n] \end{pmatrix} \stackrel{\text{def}}{=} \rho_1 * [a_1, b_1] + \rho_2 * [a_2, b_2] + \cdots + \rho_n * [a_n, b_n]$$

2. The set RCC of relational coset congruences on $\mathbb{Z}^n$

The relations we are now interested in correspond to a generalization of both relational arithmetical cosets and integer trapezoids (a special case of polyhedron corresponding to a non singular[1] system of linear inequations of the form $AX \leq b \wedge a \leq AX$). An integer trapezoid is a set of relational arithmetical cosets of zero modulo and consecutive representatives. Hence, the following model consists in sets of relational arithmetical cosets of identical modulo and consecutive representatives. It is designed so to be the intersection with the set of rational tuples $\mathbb{Q}^n$ of the rational model of trapezoid congruences which is provided in section 3.

2.1. Definition. The notion of coset congruence is generalized to $\mathbb{Z}^n$. In fact only the set of coset congruences that are not a complementary of a finite interval is generalized.

Definition 66 (LCCE). Let $\theta.[l,u]\langle m\rangle \in CC/_{\approx}$ be a normalized coset congruence and $(\delta_1, \delta_2, \ldots, \delta_n) \in \mathbb{Z}^n$, such that $\gcd(\delta_1, \delta_2, \ldots, \delta_n, m) = 1$. The *Linear Coset Congruence Equation* (LCCE)

$$\delta_1 x_1 + \delta_2 x_2 + \cdots + \delta_n x_n \equiv \theta.[l,u]\langle m\rangle$$

is defined by the linear congruence equation system with integer unknowns

$$\bigvee_{l \leq \kappa \leq u} \delta_1 x_1 + \delta_2 x_2 + \cdots + \delta_n x_n \equiv \theta\kappa \mod (m)$$

Notice that, excepted when the modulo of the linear coset congruence equation is zero, the complementary of its solution set is the solution set of the LCCE with the same linear coefficients and the complementary of the initial coset congruence. When the modulo of the LCCE is zero, the only cases for which the set of LCCEs solution sets is closed under complementation are the cases where they are empty, equal to $\mathbb{Z}^n$, or half spaces.

It is possible to extend the preceding definition since the choice of coefficients of the equation prime with the modulo of the LCCE can be omitted (and the division of the whole equation by $\gcd(\delta_1, \delta_2, \ldots, \delta_n, m)$ provides an equivalent equation satisfying the primality condition[2]).

Now we are able to define our relational concrete model.

Definition 67 (Relational coset congruences). The solution sets of LCCEs non-singular systems are called *Relational Coset Congruences* of $\mathbb{Z}^n$. The set of Relational Coset Congruences is noted RCC.

A relational coset congruence is represented on the figure V.3. It corresponds to the relational arithmetical cosets $\left(\begin{smallmatrix}2\\0\end{smallmatrix}\right)\left\langle\begin{smallmatrix}2&0\\1&3\end{smallmatrix}\right\rangle_{(2,0)}$ and $\left(\begin{smallmatrix}3\\0\end{smallmatrix}\right)\left\langle\begin{smallmatrix}2&0\\1&3\end{smallmatrix}\right\rangle_{(2,0)}$ and to the single LCCE $x - 2y \equiv 1.[2,3]\langle 6\rangle$.

Now, we are going to build the parametric representation of relational coset congruences up to now equationally defined. For that purpose we start by intersecting solution sets of

[1] Recall that a non-singular system of linear equations $AX = b$ is such that all the rows of A are linearly independent.

[2] See the proof of the theorem 90 for the whole process of division.

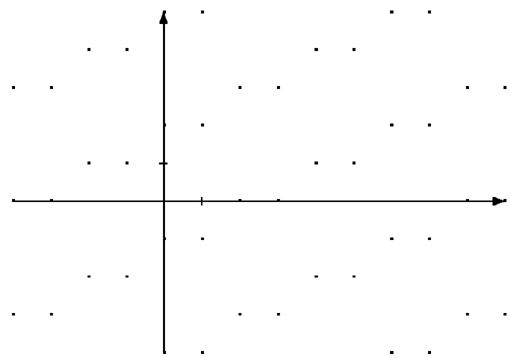


FIGURE V.3. Relational coset congruence.

LCCEs. The first step of the intersection process considers a special kind of LCCE in which one operand of the intersection is a rational linear congruence equation. Then we expect to generalize to general LCCEs. A direct extension of the proposition 13 deals with LCCE and follows.

PROPOSITION 68 (LCCE IN A COSET OF $\mathbb{Z}^n$). *The solution set of the LCCE*

$$\delta_1 x_1 + \delta_2 x_2 + \cdots + \delta_n x_n \equiv \theta.[l,u]\langle m\rangle \tag{46}$$

in the coset $A\langle M\rangle_{(p,0)}$ *is*

$$\bigcup_{l\leq\kappa\leq u}(A + (\theta\kappa - (\delta_1,\delta_2,\ldots,\delta_n).A)MB)\langle MN\rangle_{(q,0)}$$

where $B\langle N\rangle_{(q,0)}$ *is the solution of the LCCE*

$$(\delta_1,\delta_2,\ldots,\delta_n)M\begin{pmatrix} y_1\\ y_2\\ \vdots\\ y_p\end{pmatrix} \equiv 1 \mod (m) \tag{47}$$

in $\mathbb{Z}^p$ *if the equation* (47) *has a non empty solution set. Otherwise, the solution set of equation* (46) *is empty.*

Unfortunately, we did not find an algorithm to solve a LCCE in the solution set of an LCCE in $\mathbb{Z}^n$. Hence we do not provide a parametric representation of the relational coset congruences by incrementaly solving the LCCEs in the solution set of the preceding ones (the principle of that method is detailed in section 3.3 and provides a parametric representation of trapezoid congruences given an equational representation). But following [Gra91a] we have a good algorithm to solve a relational coset congruence when all the coset congruences of the LCCEs are reduced to single cosets. No extensions of that algorithm seem to be able to deal with general relational coset congruences. Hence the only solution in order to give a parametric representation of a general coset congruence is to enumerate its constitutive cosets, each of which corresponds to one linear congruence equation system. The proposition 13 implies that all these cosets have the same modulo. The only theoretical problem concerned with this enumeration process is the possibly infiniteness of the representative of a LCCE coset congruence with zero modulo. Methods like those of [Fea88b] provide a parametric representation of solution sets of systems of linear constraints. Hence the above mentioned enumeration is obtained by partitioning the LCCE system into two subsystems: one with non zero modulo equations and the other with zero modulo equations.

2.2. Equivalence relation. The only case where equivalent cosets (representing the same integer tuples set) are easily detectable is when they are equal to $\mathbb{Z}^n$.

PROPOSITION 69 (RELATIONAL COSET CONGRUENCES EQUAL TO $\mathbb{Z}^n$). *A relational coset congruence $C = \{\Delta_i.X \equiv \theta_i.[l_i, u_i]\langle m_i\rangle\}_{i\in[1,p]}$ is equal to $\mathbb{Z}^n$ if and only if*

$$\forall i \in [1,p] \quad l_i \leq u_i \text{ and } \theta_i.[l_i, u_i]\langle m_i\rangle = \mathbb{Z}$$

PROOF. $C = \mathbb{Z}^n$ is equivalent to say that every single LCCE solution set is equal to $\mathbb{Z}^n$. If it is the case for $\Delta_i.X \equiv \theta_i.[l_i, u_i]\langle m_i\rangle$, then let us show that $\theta_i.[l_i, u_i]\langle m_i\rangle = \mathbb{Z}$ and $l_i \leq u_i$. If $m_i = 0$, then $\gcd(\Delta_{i1}, \Delta_{i2}, \ldots, \Delta_{in}) = 1$ and knowing that the solution set of the LCCE is $\mathbb{Z}^n$, Bezout's theorem implies that $\theta_i.[l_i, u_i]\langle m_i\rangle = \mathbb{Z}$ ($l_i \leq u_i$ because otherwise the LCCE has no solutions at all). Suppose now $m_i \neq 0$, if $l_i > u_i$ then the LCCE has no solutions and if $\theta_i.[l_i, u_i]\langle m_i\rangle \neq \mathbb{Z}$ then there exists κ such that $\theta_i\kappa \notin \theta_i.[l_i, u_i]\langle m_i\rangle$. The solution set of $\Delta_i.X \equiv \theta_i\kappa \mod (m_i)$ is not empty since $\gcd(\Delta_{i1}, \Delta_{i2}, \ldots, \Delta_{in}, m_i) = 1$ and is not in the solution set of $\Delta_i.X \equiv \theta_i.[l_i, u_i]\langle m_i\rangle$ which is consequently not $\mathbb{Z}^n$. □

Of course, the equivalence of relational coset congruences is provided by comparing their parametric representations because comparing their common modulos and then comparing their representative sets is possible. We do not use so costly operations and shall only use operators that need constant time with respect to the number of cosets contained in its relational coset congruence operands. Hence contrary to the non relational model of coset congruences, no normalization is explicated here.

The set inclusion induced order is possibly defined with respect to the relational coset congruence parametric representation too, but once again it does not appear to be efficiently implementable.

2.3. Precision concrete order. As for the case of coset congruences, the definition of an accuracy function is needed, which implements a heuristics corresponding to the informative order.

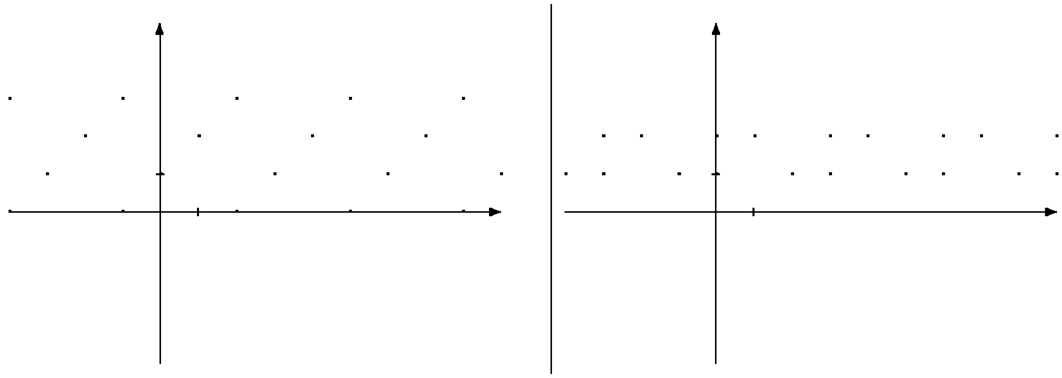


FIGURE V.4. Relational coset congruences with equal accuracy.

DEFINITION 70 (ACCURACY $\iota^{\bowtie}$). Let $RC = \{\Delta_i.X \equiv C_i\}_{i\in[1,p]}$ be a relational coset congruence. Its *accuracy* $\iota^{\bowtie}(RC)$ is defined by

$$\iota^{\bowtie}(RC) \quad = \quad \prod_{i\in[1,p]} \frac{\iota\,(C_i)}{3}$$

where ι is the coset congruence accuracy function.

The accuracy function estimates the density of integer points contained in a relational coset congruence. The most accurate relational coset congruence is of accuracy zero; it is a representation of the empty set. As is explicated below in the below definition of the precision concrete order, this definition of accuracy is only useful to compare two relational coset congruences with the same dimension (see below). Notice that adding to a relational coset congruence an LCCE whose coset congruence is equal to $\mathbb{Z}$ does not change its associated accuracy. Unfortunately, adding to a relational coset congruence an LCCE whose coset congruence is empty does not always provide a zero accuracy (think of LCCEs whose coset congruences have a greater lower bound than their upper bound). Hence a significant improvement to the accuracy measure consists in removing these equations and replacing them by equivalent LCCEs with empty solution set, before determining the accuracy of a relational coset congruence.

DEFINITION 71 (PRECISION CONCRETE ORDER $\preceq_\natural$). Let RC_1 and RC_2 be two relational coset congruences. $RC_1 \preceq_\natural RC_2$ if and only if

- the accuracy $\iota^{\bowtie}(RC_1)$ is zero or either
- the number of LCCEs with finite width representative and zero modulo is greater in RC_1 than in RC_2, or
- the numbers of LCCEs with finite width representative and zero modulo are equal and $\iota^{\bowtie}(RC_1) \leq \iota^{\bowtie}(RC_2)$

The elements of RCC are more precise if they are defined by more LCCEs with finite representative and zero modulo. The relational coset congruences RC_1 and RC_2 of the figure V.4 corresponding respectively to the LCCEs

$$(x - y \equiv 1.\,[2,2]\,\langle 3\rangle\,,\, y \equiv 1.\,[0,3]\,\langle 0\rangle)$$

and to

$$(x - y \equiv 1.\,[1,2]\,\langle 3\rangle\,,\, y \equiv 1.\,[1,2]\,\langle 0\rangle)$$

have the same accuracy and hence are equivalent for the precision concrete order. The relational coset congruence RC_1 of the figure V.4 is smaller for the precision order than C_0 of the figure V.3. Intuitively we see that RC_1 is of dimension one when C_0 is of dimension 2.

3. The set TC of trapezoid congruences on $\mathbb{Q}^n$

Before getting into the definition of trapezoid congruences, we need to define a componentwise partial order on $\mathbb{Q}^n$, given $n \geq 1$.

Definition 72 (Basis-relative partial order on $\mathbb{Q}^n$). Given an integer p such that $0 \leq p \leq n$ and a collection $Q = (Q_1, \ldots, Q_p)$ of p linearly independent vectors of $\mathbb{Q}^n$, the partial order $\underset{Q}{\leq}$ on $\mathbb{Q}^n$ is defined by:

$$\forall G, H \in \mathbb{Q}^n,\ G \underset{Q}{\leq} H \Leftrightarrow \exists(\lambda_1, \ldots, \lambda_p) \in \mathbb{Q}^p_+, H - G = \lambda_1 Q_1 + \cdots + \lambda_p Q_p$$

$\underset{I}{\leq}$ is noted $\leq$ if there is no risk of confusion.

Notice that if $p = 0$ then the relation $\underset{Q}{\leq}$ is equivalent to the equality. The figure V.5 illustrates

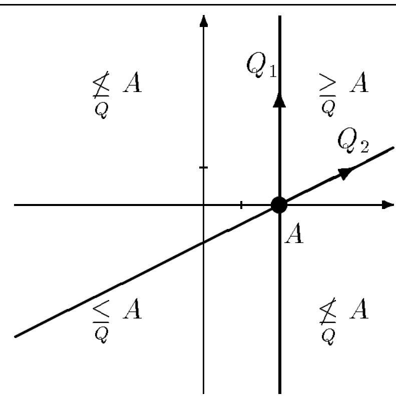


Figure V.5. Partition of $\mathbb{Q}^2$ by the point A and the order $\underset{Q}{\leq}$.

this definition in $\mathbb{Q}^2$ with the basis $Q = \left(\begin{smallmatrix} 0 & 2 \\ 3 & 1 \end{smallmatrix}\right)$ and the point $A = \left(\begin{smallmatrix} 2 \\ 0 \end{smallmatrix}\right)$.

3.1. Dual definitions. We are going to give two equivalent definitions of trapezoid congruences. These two definitions are both useful, the equational one for intuitive understanding about trapezoid congruences and the parametrical one for their machine representation. Later in this chapter, we will see that these representations are quite complementary so that some lattice operations or abstract operators have the use of both of them.

The notion of interval congruence is now generalized to $\mathbb{Q}^n$. Actually, only the interval congruences that are not a complementary of a finite rational interval are generalized.

DEFINITION 73 (RLICE). Let $[a,b]\langle q\rangle$ be an interval congruence and $(\lambda_1,\dots,\lambda_n)\in\mathbb{Q}^n$. The *Rational Linear Interval Congruence Equation* (RLICE)

$$\lambda_1 x_1+\lambda_2 x_2+\cdots+\lambda_n x_n \equiv [a,b]\langle q\rangle \tag{48}$$

is defined by the linear congruence equation system with rational unknowns

$$\bigvee_{a\le x_0\le b} \lambda_1 x_1+\lambda_2 x_2+\cdots+\lambda_n x_n \equiv x_0 \mod (q)$$

Geometrically, a RLICE corresponds to a set of "thick"[3] parallel hyperplanes regularly dispersed according to the modulo of the congruence equation. $\mathbb{Q}^n$ and the empty set are both representable using RLICEs ($[a,b]\langle q\rangle=\mathbb{Q}$ for the first and $a>b$ for the latter case). If q is zero, the equation (48) is possibly noted

$$a\le \quad \lambda_1 x_1+\cdots+\lambda_n x_n \quad \le\ b$$

and if moreover a or b is infinite, it is omitted, for example giving

$$a \quad \le \quad \lambda_1 x_1+\cdots+\lambda_n x_n$$

Let us now introduce a normalized form of a RLICE where its linear coefficients and modulo are prime.

DEFINITION 74 (PRIME RLICE). The RLICE

$$\lambda_1 x_1+\lambda_2 x_2+\cdots+\lambda_n x_n \quad\equiv\quad [a,b]\langle q\rangle$$

is said to be *prime* if $\gcd(\lambda_1,\lambda_2,\dots,\lambda_n,q)=1$

It is always possible to get an equivalent prime RLICE from any RLICE by dividing it by the greatest common divisor of its linear coefficients and its modulo. For example the RLICE $\frac{3}{4}x-2y+\frac{3}{2}z\equiv\left[\frac{1}{7},\frac{2}{7}\right]\left\langle\frac{27}{4}\right\rangle$ is transformed into $3x-8y+6z\equiv\left[\frac{4}{7},\frac{8}{7}\right]\langle 27\rangle$.

DEFINITION 75 (RLICE NEGATION). Let E be a RLICE. E' is its *negation* if and only if the system $E\wedge E'$ is equivalent to the disjunction of two rational linear congruence equations.

The negation of a RLICE always exists when its modulo is non zero or its representative upper bound is infinite. It is obtained by taking the complementation of the interval congruence used for the definition of the RLICE following the definition 30. For example the following left hand side systems are composed of two mutually negative RLICEs, because of their equivalence with the right hand side systems which are composed of two rational linear congruence equations

[3] The thickness comes from the possibly non null width of the representative $[a,b]$ in equation (48).

(possibly identical):

$$\left.\begin{array}{rcl} 2x+3y & \equiv & [4,6] \mod 32 \\ 2x+3y & \equiv & [6,36] \mod 32 \end{array}\right\} = \left\{\begin{array}{rcl} 2x+3y & \equiv & 4 \mod 32 \\ 2x+3y & \equiv & 6 \mod 32 \end{array}\right.$$

$$\left.\begin{array}{rcl} 4 & \leq & 2x+3y \\ -4 & \leq & -2x-3y \end{array}\right\} = \left\{\begin{array}{rcl} 2x+3y & = & 4 \\ 2x+3y & = & 4 \end{array}\right.$$

Here is the definition of our abstract model.

Definition 76 (Equational trapezoid congruence). The rational tuple sets corresponding to solutions of RLICE non-singular systems are *equational trapezoid congruences* of $\mathbb{Q}^n$.

An equational trapezoid congruence is said to be prime if all its constitutive RLICEs are prime. Here is the parametric equivalent definition.

Definition 77 (Parametric trapezoid congruences). Let

- p, r, s and t be non negative integers such that $0 \leq p+r+s+t \leq n$;
- $S = (S_1, \ldots, S_{p+r+s+t}) \in \mathbb{Q}^{n,p+r+s+t}$ be a collection of linearly independent vectors of $\mathbb{Q}^n$;
- $A, B \in \mathbb{Q}^n$ and $C \in \mathbb{Q}^{p+r+s+t}$ such that $B - A = SC$.

The *parametric trapezoid congruence* of $\mathbb{Q}^n$ with lower bound A, upper bound B, shape S of integer rank p, rational rank r, bounded rank s and unbounded rank t is the subset of $\mathbb{Q}^n$ noted $[A,B]\,\langle S\rangle_{(p,r,s,t)}$ defined by:

$$[A,B]\,\langle S\rangle_{(p,r,s,t)} \stackrel{\text{def}}{=} \left\{ X \in \mathbb{Q}^n / \begin{array}{c} X = A + S(\Gamma+\Phi), \\ \Phi \in \mathbb{Z}^p\mathbb{Q}^r\{0\}^s\mathbb{Q}_+^t, \\ O \leq \Gamma \leq C \end{array} \right\} \tag{49}$$

or equivalently in terms of rational linear cosets by:

$$[A,B]\,\langle S\rangle_{(p,r,s,t)} = \bigcup_{\substack{A \underset{S}{\leq} X \underset{S}{\leq} B \\ O \underset{S^{p+r+s+1,p+r+s+t}}{\leq} Y}} (X+Y)\,\langle S^{p+r}\rangle_{(p,r)} \tag{50}$$

where O is the null vector and $\leq$ the componentwise order.

The proof of the equivalence of the two defining expressions 49 and 50 is just a verification. Sometimes, the parentheses around S are omitted for a sake of clarity. The relation between C and the bounds of the trapezoid congruence is not recalled if there is no risk of confusion.

Geometrically speaking, the definition (49) corresponds to a set of rational tuples which are the sum of a point A, a trapezoid $\{S\Gamma \in \mathbb{Q}^n,\ 0 \leq \Gamma \leq C\}$ and a regular distribution pattern $\{S\Phi,\ \Phi \in \mathbb{Z}^p\mathbb{Q}^r\{0\}^s\mathbb{Q}_+^t\}$ (look at the figure V.7 to differentiate the four kinds of components

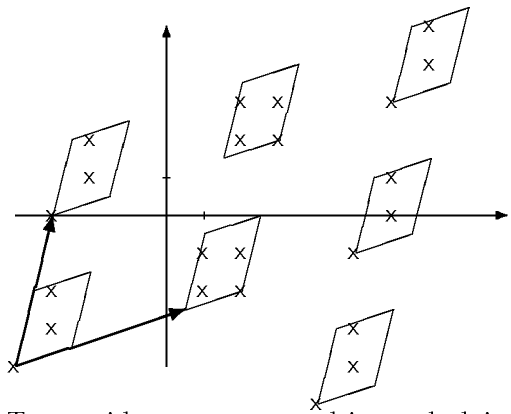


FIGURE V.6. Trapezoid congruence and its underlying relational coset congruence.

of this distribution pattern), while the definition (50) considers a set of rational linear cosets of common modulo the linear subgroup $\langle S^{p+r}\rangle_{(p,r)}$ and consecutive representatives bounded by A and B (for the order induced by S) and unbounded in the directions of the t last vectors of S. We call

$$\left\{X+Y \in \mathbb{Q}^n,\ A \underset{S}{\leq} X \underset{S}{\leq} B \text{ and } O \underset{S^{p+r+s+1,p+r+s+t}}{\leq} Y\right\}$$

the representative and the linear subgroup $\langle S^{p+r}\rangle_{(p,r)}$ the modulo of the trapezoid congruence $[A,B]\,\langle S\rangle_{(p,r,s,t)}$. Notice that the representative of a parametrical trapezoid congruence is not unique. $TC(\langle Q\rangle_{(p,r)})$ denotes the set of all trapezoid congruences of modulo $\langle Q\rangle_{(p,r)}$ and TC the set of all trapezoid congruences.

3.2. Examples. Examples will only be presented in the case of $\mathbb{Q}^2$, although it is often necessary to consider much higher dimension spaces. Let us see on an example what a parametrical trapezoid congruence looks like. Figure V.6 represents the parametrical trapezoid congruence

$$\left[\begin{pmatrix} \frac{3}{2} \\ \frac{3}{2} \end{pmatrix}, \begin{pmatrix} \frac{7}{2} \\ 4 \end{pmatrix}\right] \left\langle \begin{matrix} \frac{9}{2} & 1 \\ \frac{3}{2} & 4 \end{matrix} \right\rangle_{(2,0,0,0)}$$

and is the solution of the prime equational trapezoid congruence

$$\left\{ \begin{array}{rcl} x-3y & \equiv & [\frac{5}{2},8]\,\langle 11\rangle \\ 8x-2y & \equiv & [9,20]\,\langle 33\rangle \end{array} \right.$$

too. The two linearly independent vectors constituting the modulo have been represented with thick arrows. The drawn trapezoids with sides parallel to each vector of the modulo stand for the representatives of the given parametrical trapezoid congruence. More classical patterns of subscript set values like strips or blocks can be easily represented by parametrical trapezoid congruences.

The figure V.7 summarizes different kinds of shapes of parametrical trapezoid congruences of $\mathbb{Q}^2$. Example (1) is the rational linear coset

$$\begin{pmatrix} \frac{3}{2} \\ \frac{3}{2} \end{pmatrix} \left\langle \begin{array}{cc} \frac{9}{2} & \frac{1}{2} \\ \frac{3}{2} & 2 \end{array} \right\rangle_{(2,0)} = \left[\begin{pmatrix} \frac{3}{2} \\ \frac{3}{2} \end{pmatrix}, \begin{pmatrix} \frac{3}{2} \\ \frac{3}{2} \end{pmatrix} \right] \left\langle \begin{array}{cc} \frac{9}{2} & \frac{1}{2} \\ \frac{3}{2} & 2 \end{array} \right\rangle_{(2,0,0,0)}$$

Example (2) is the set of bounded parallelograms

$$\left[\begin{pmatrix} 2 \\ 2 \end{pmatrix}, \begin{pmatrix} 7 \\ 3 \end{pmatrix} \right] \left\langle \begin{array}{cc} 3 & 4 \\ 3 & 0 \end{array} \right\rangle_{(1,0,1,0)}$$

Then the bounded and unbounded ranks are exchanged providing the set of half strips case (3)

$$\left[\begin{pmatrix} 2 \\ 2 \end{pmatrix}, \begin{pmatrix} 7 \\ 3 \end{pmatrix} \right] \left\langle \begin{array}{cc} 3 & 4 \\ 3 & 0 \end{array} \right\rangle_{(1,0,0,1)}$$

The case (4) is a set of unbounded strips

$$\left[\begin{pmatrix} -2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right] \left\langle \begin{array}{cc} 6 & 1 \\ 2 & 2 \end{array} \right\rangle_{(1,1,0,0)}$$

it is equal to

$$\left[\begin{pmatrix} -2 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 3 \end{pmatrix} \right] \left\langle \begin{array}{cc} 6 & 1 \\ 2 & 2 \end{array} \right\rangle_{(2,0,0,0)}$$

too. The example (5) corresponds to

$$\left[\begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 6 \\ 2 \end{pmatrix} \right] \left\langle \begin{array}{cc} 3 & 1 \\ 3 & 0 \end{array} \right\rangle_{(0,0,2,0)}$$

The example (6) corresponds to one representative of example (3) and is the trapezoid congruence

$$\left[\begin{pmatrix} -1 \\ -1 \end{pmatrix}, \begin{pmatrix} 5 \\ 0 \end{pmatrix} \right] \left\langle \begin{array}{cc} 3 & 1 \\ 3 & 0 \end{array} \right\rangle_{(0,0,1,1)}$$

The example (7) to

$$\left[\begin{pmatrix} -1 \\ -1 \end{pmatrix}, \begin{pmatrix} -1 \\ -1 \end{pmatrix} \right] \left\langle \begin{array}{cc} 3 & 1 \\ 3 & 0 \end{array} \right\rangle_{(0,0,0,2)}$$

and finally the example (8) corresponds to a half plane

$$\left[\begin{pmatrix} 0 \\ -1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right] \left\langle \begin{array}{cc} 1 & 1 \\ 1 & 0 \end{array} \right\rangle_{(0,1,0,1)}$$

Hence the trapezoid congruence model contains the most usually encountered patterns in the field of matrix computation.

3.3. Equivalence of parametrical and equational trapezoid congruences. The proof of the equivalence of the two definitions of trapezoid congruences (given in appendix D) leads to an algorithm used implicitly in the following. To take a parametrical trapezoid congruence and to give the corresponding equational trapezoid congruence is no more difficult than solving a set of linear equations. The other way is a bit more complicated, since first the equations are solved and then their solution sets are intersected. In fact, equation solving and intersection of two solution sets are equivalent problems because solving an equation in the solution set of the other gives the intersection of the two solution sets. The solution of a RLICE in $\mathbb{Q}^n$ is a parametrical trapezoid congruence, hence a method to solve a RLICE in a parametrical trapezoid congruence is only needed.

The latter problem is easily reduced to the particular case in which the parametrical trapezoid congruence is of orthonormal shape (the collection of vectors constituting the shape is orthonormal). The resolution of a RLICE in a parametrical trapezoid congruence is used to define the abstract test with a RLICE condition.

The following theorem (proven in appendix D) holds:

THEOREM 78 (TRAPEZOID CONGRUENCE REPRESENTATIONS EQUIVALENCE). *The equational and parametric definitions of trapezoid congruences are equivalent.*

In the following, parametrical and equational trapezoid congruences are not differentiated, except if one formalism is explicitly requested. An example of the two equivalent representations of a trapezoid congruence is given at the beginning of the section 3.2.

3.4. Comparison. The partial order on TC is not expressible in terms of the order on rational linear cosets, but is reduced by the following theorem to the comparison on interval congruences.

THEOREM 79 (CHARACTERIZATION OF THE PARTIAL ORDER ON TC). *Given a trapezoid congruence in parametrical form $T_1 = [A, B]\langle S\rangle_{(p,r,s,t)}$ and another in equational form $T_2 = (\Lambda_i.X \equiv [a_i, b_i]\langle q_i\rangle)_{i\in[1,m]}$, $T_1 \subseteq T_2$ if and only if for all i in $[1, m]$:*

$$[g_i, d_i]\langle e_i\rangle \quad \subseteq_\sharp \quad [a_i, b_i]\langle q_i\rangle$$

where

$$\begin{aligned} [g_i, d_i] &= \Lambda_i.A + \sum_{j=1}^{p+r+s+t} \Lambda_i.S_j * [0, c_j] + \sum_{j=p+1}^{p+r} \Lambda_i.S_j * [-\infty, +\infty] + \sum_{j=p+r+s+1}^{p+r+s+t} \Lambda_i.S_j * [0, +\infty] \\ e_i &= \gcd(\Lambda_i.S_1, \Lambda_i.S_2, \ldots, \Lambda_i.S_p) \end{aligned}$$

PROOF. $T_1 \subseteq T_2$ if and only if

$$\Lambda_i.(A + S(\Gamma + \Phi)) \equiv [a_i, b_i]\langle q_i\rangle$$

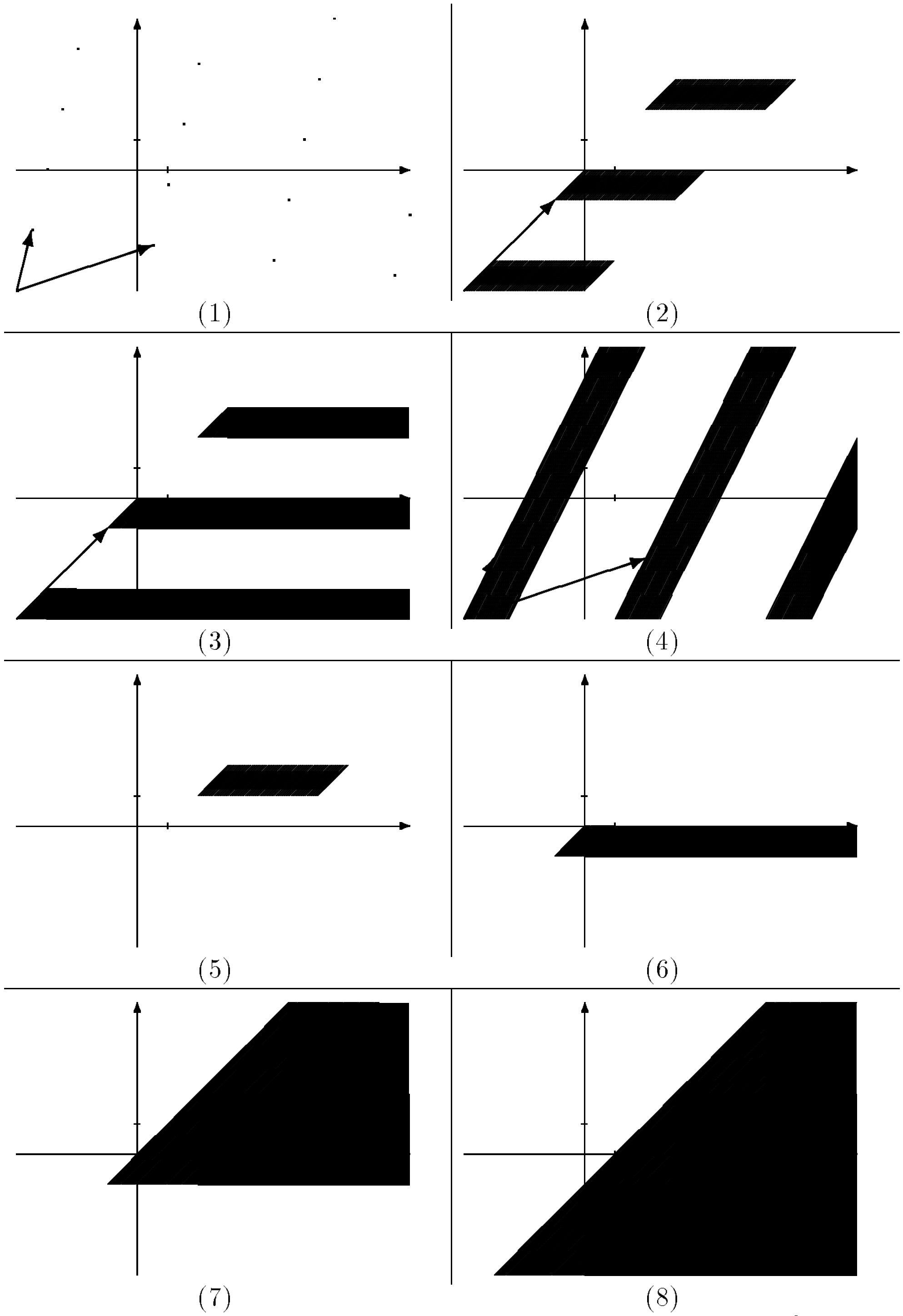


FIGURE V.7. Different kinds of trapezoid congruences of $\mathbb{Q}^2$.

for all i in $[1, m]$, $0 \leq \Gamma \leq C$ and Φ in $\mathbb{Z}^p\mathbb{Q}^r\{0\}^s\mathbb{Q}_+^t$, which is equivalent to the condition

$$f_i + (\Lambda_i.S_1)x_1 + (\Lambda_i.S_2)x_2 + \cdots + (\Lambda_i.S_p)x_p \equiv [a_i, b_i]\,\langle q_i \rangle$$

for all $f_i \in [g_i, d_i]$ and $x_i \in \mathbb{Z}$. Now noticing that

$$\langle \Lambda_i.S_1 \rangle + \langle \Lambda_i.S_2 \rangle + \cdots + \langle \Lambda_i.S_p \rangle = \langle e_i \rangle$$

we see that it is equivalent to the interval congruence inclusion figuring in the theorem. $\square$

The comparison algorithm follows directly from this theorem. For example the comparison

$$\left[\begin{pmatrix} -2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ \frac{3}{2} \end{pmatrix}\right] \left\langle \begin{matrix} \frac{11}{2} & 1 \\ 1 & \frac{3}{2} \end{matrix} \right\rangle_{(1,0,1,0)} \subseteq \left[\begin{pmatrix} -2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}\right] \left\langle \begin{matrix} 6 & 2 \\ 1 & 2 \end{matrix} \right\rangle_{(1,1,0,0)}$$

where the right operand is equationally represented by the RLICE $2x - y \equiv [-4, -1]\,\langle 10 \rangle$ is reduced to the comparison on IC

$$\left[-4, \frac{-3}{2}\right] \langle 10 \rangle \quad \subseteq_\sharp \quad [-4, -1]\,\langle 10 \rangle$$

Proposition 80 (Trapezoid congruences equal to $\mathbb{Q}^n$). *Let $T = [A, B]\,\langle S \rangle_{(p,r,s,t)}$ be a trapezoid congruence. T is equal to $\mathbb{Q}^n$ if and only if*

$$\begin{cases} p + r = n \\ A + S_1 + S_2 + \cdots + S_p \underset{S}{\leq} B \end{cases}$$

or equivalently, there exists an integer $i \leq p$ such that $c_i \geq 1$ (with $B - A = SC$).

Proof. Since S is a basis of $\mathbb{Q}^n$, we have $p + r + s + t = n$.
If $s + t > 0$ then T surely does not contain points P such that $P \underset{S^{\overline{p+r+1,n}}}{\leq} A$ hence $s + t = 0$.
The last point comes from the consideration of the definition (50) of parametrical trapezoid congruences. $\square$

Notice that the equational representation allows a simpler characterization of trapezoid congruences equal to $\mathbb{Q}^n$: the interval congruences of all the RLICEs of the system must be equal to $\mathbb{Z}$ (and the interval congruence bounds well ordered). The characterization is preferably done on the parametric representation in order to minimize the representation translations during the analysis (most of the operators on trapezoid congruences use the parametric representation).

Proposition 81 (Trapezoid congruences equal to $\emptyset$). *Let $T = [A, B]\,\langle S \rangle_{(p,r,s,t)}$ be a trapezoid congruence. T is equal to $\emptyset$ if and only if*

$$A \underset{S}{\not\leq} B$$

or equivalently, there exists an integer i such that $c_i < 0$ (with $B - A = SC$).

This is a direct consequence of the definition of the parametric trapezoid congruence. The equational way to say the same thing is to consider systems where at least one RLICE has its representative lower bound greater than its upper bound (recall that all equations are independent, hence no incompatibilities occur between RLICEs).

APPENDIX D

Representation translation algorithms

In order to prove the theorem 78, we are going to build two algorithms providing the translations between equational and parametric representations. These algorithms are extensions of Granger's algorithms providing the equivalence between equational and parametric representations of cosets of $\mathbb{Q}^n$. Six preliminary lemmas are necessary, the two first providing the solution set of a non zero and a zero modulo RLICE in a rational linear coset, the next two the solution set of a non zero and a zero modulo RLICE in an orthonormal trapezoid congruence[1]. The next lemma reduces the determination of the solution set of a RLICE in a trapezoid congruence to the determination of the solution set of another RLICE in an orthonormal trapezoid congruence. Finally the last lemma provides the translation from a parametrical representation to an equational one.

Considering a linear congruence equation with several consecutive possible representatives, it follows directly from proposition 12 that the intersection of $\mathbb{Z}^p\mathbb{Q}^r$ with the solution set of a non zero modulo RLICE is a parametric trapezoid congruence of integer rank $p+1$ and rational rank $r-1$.

LEMMA 82 (RLICE IN $\mathbb{Z}^p\mathbb{Q}^r$). *Let λ_{p+1} and q be non zero rational numbers, a and b be finite rational numbers. The solution set of the RLICE*

$$\lambda_1 x_1 + \lambda_2 x_2 + \ldots + \lambda_{p+1} x_{p+1} + \ldots + \lambda_{p+r} x_{p+r} \quad \equiv \quad [a, b]\,\langle q\rangle$$

in $[O, O]\,\langle I\rangle_{(p,r,0,0)}$ with $0 \leq p \leq p+r-1$ is the trapezoid congruence

$$\begin{aligned} T \;=\; & \left[\frac{a}{\lambda_{p+1}} I_{p+1}, \frac{b}{\lambda_{p+1}} I_{p+1}\right] \left\langle I_1 - \frac{\lambda_1}{\lambda_{p+1}} I_{p+1}, \ldots, I_p - \frac{\lambda_p}{\lambda_{p+1}} I_{p+1}, \frac{|q|}{\lambda_{p+1}} I_{p+1},\right. \\ & \qquad\qquad\qquad \left. I_{p+2} - \frac{\lambda_{p+2}}{\lambda_{p+1}} I_{p+1}, \ldots, I_{p+r} - \frac{\lambda_{p+r}}{\lambda_{p+1}} I_{p+1} \right\rangle_{(p+1,r-1,0,0)} \end{aligned}$$

The columns of the shape of T are linearly independent.

[1] An orthonormal element of TC has its lower bound equal to the null vector and an orthonormal shape.

This lemma is generalized to RLICEs with one non zero coefficient of rank greater than $p+1$ and less than $p+r$ by simply permuting the variables.

Now if the modulo of the RLICE is zero, the result is a trapezoid congruence of rational rank $r-1$ too, but of incremented bounded or unbounded rank (instead of integer rank as for the preceding lemma) depending on the finitness of the representative of the RLICE.

LEMMA 83 (DOUBLE LINEAR INEQUATION IN $\mathbb{Z}^p\mathbb{Q}^r$). *Let λ_{p+r} be a non zero rational number and a a finite rational number. The solution set of the RLICE*

$$a \leq \quad \lambda_1 x_1 + \lambda_2 x_2 + \ldots + \lambda_{p+r} x_{p+r} \quad \leq b$$

in $[O,O]\langle I\rangle_{(p,r,0,0)}$ with $0 \leq p \leq p+r-1$ is the trapezoid congruence

$$T \;=\; \left[\frac{a}{\lambda_{p+r}} I_{p+r}, \frac{a+(b-a)c}{\lambda_{p+r}} I_{p+r}\right] \left\langle I_1 - \frac{\lambda_1}{\lambda_{p+r}} I_{p+r}, \ldots, I_{p+r-1} - \frac{\lambda_{p+r-1}}{\lambda_{p+r}} I_{p+r}, \frac{1}{\lambda_{p+r}} I_{p+r} \right\rangle_{(p,r-1,c,1-c)}$$

where c is 1 when b is finite and 0 otherwise. The columns of the shape of T are linearly independent.

PROOF. The same verification as for Proposition 12 is necessary and is done by noticing that the difference there between the upper and lower bounds is $\frac{b-a}{|q|}\left(\frac{|q|}{\lambda_{p+1}} I_{p+1}\right)$. Hence the upper bound is greater than the lower bound for the partial order relative to the shape of T and the only points comprised between them are the ones corresponding to a solution of one congruence equation of representative between a and b following Proposition 12. □

Now a similar result is provided by the following lemma when the representative of the original trapezoid congruence is of non null sizes (its lower and upper bounds are distinct).

LEMMA 84 (NON ZERO MODULO RLICE IN A TRAPEZOID CONGRUENCE). *Let r be a positive integer, p, s, t three non negative integers such that $p+r+s+t=n$ and a and b finite rational numbers. The solution of the RLICE*

$$\lambda_1 x_1 + \lambda_2 x_2 + \ldots + \lambda_n x_n \quad \equiv \quad [a,b]\langle q\rangle \tag{51}$$

such that $q\lambda_{p+1} \neq 0$ and $a.b$ is finite, in the trapezoid congruence:

$$[O,P]\langle I\rangle_{(p,r,s,t)}$$

is equal to the trapezoid congruence:

$$\left[\frac{a}{|q|} S_{p+1} \;,\; \sum_{i=1}^{p} p_i S_i + \frac{b}{|q|} S_{p+1} + \sum_{i=p+r+1}^{n} S_i\right] \langle S\rangle_{(p+1,r-1,s,t)}$$

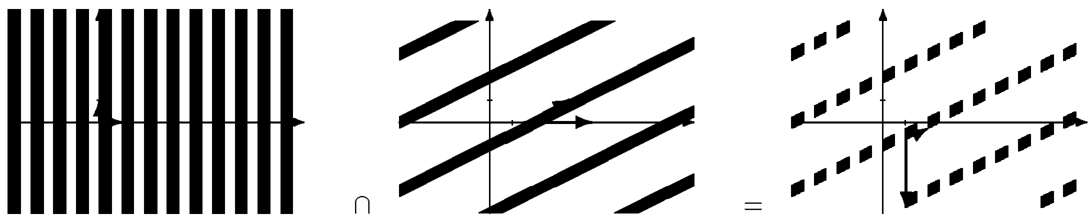


FIGURE D.8. Orthonormal trapezoid congruence and non zero modulo RLICE intersection.

where:

$$S_i \quad = \quad \begin{cases} \frac{|q|}{\lambda_{p+1}} I_{p+1} & \textit{if } i = p+1 \\ \\ I_i - \frac{\lambda_i}{\lambda_{p+1}} I_{p+1} & \textit{otherwise} \end{cases}$$

Moreover, $(\lambda_1, \lambda_2, \ldots, \lambda_n)$ *is orthogonal to the rational rank columns of* S*.*

PROOF. Using the expression (49) of the parametric trapezoid congruence definition, the trapezoid congruence provides

$$\Omega = [O,P]\,\langle I\rangle_{(p,r,s,t)} = \left\{ X \in \mathbb{Q}^n / \begin{array}{l} X = I(\Gamma+\Phi), \\ \Phi \in \mathbb{Z}^p\mathbb{Q}^r\{0\}^s\mathbb{Q}^t_+, \\ O \leq \Gamma \leq P \end{array} \right\}$$

and noticing that the order $\leq$ corresponds to the component wise order on vectors, Ω is equal to

$$(52) \qquad \bigcup_{0\leq \xi_i \leq p_i,\; i\in[1,p+r+s]} (\xi_1\,\langle 1\rangle) \times \ldots \times (\xi_p\,\langle 1\rangle) \times \mathbb{Q}^r \times \{\xi_{p+r+1}\} \times \ldots \times \{\xi_{p+r+s}\} \times \mathbb{Q}^t_+$$

Now, in the RLICE (51), we make the unknown change and constant instantiation

$$\begin{array}{lccl} \forall i \in [1,p] & x_i & = & y_i + \xi_i \\ \forall i \in [p+1,p+r] & x_i & = & y_i \\ \forall i \in [p+r+1,n] & x_i & = & \xi_i \end{array}$$

The ξ_i are the constants considered in expression (52) and are arbitrary non negative rational numbers when $i > p+r+s$. Hence we get

$$\begin{array}{l} (\lambda_1 y_1 + \lambda_1\xi_1 + \ldots + \lambda_p y_p + \lambda_p \xi_p) + (\lambda_{p+1} y_{p+1} \ldots + \lambda_{p+r} y_{p+r}) + \\ \qquad\qquad\qquad\qquad (\lambda_{p+r+1}\xi_{p+r+1} + \ldots + \lambda_n \xi_n) \;\equiv\; [a,b]\,\langle q\rangle \end{array}$$

and we are going to solve it parametrically with respect to its $s+t$ last unknowns. The problem is now transformed into solving the RLICE

$$\lambda_1 y_1 + \ldots + \lambda_p y_p + \lambda_{p+1} y_{p+1} + \ldots + \lambda_{p+r} y_{p+r} \quad \equiv \quad [a-\rho, b-\rho]\,\langle q\rangle$$

where $\rho = \sum_{i=1}^{p} \lambda_i \xi_i + \sum_{i=p+r+1}^{n} \lambda_i \xi_i$, in

$$\mathbb{Z}^p \times \mathbb{Q}^r \quad = \quad [O, O] \langle I \rangle_{(p,r,0,0)}$$

where $I = I(p+r)$ (by applying the same translation to expression (52)). Lemma 82 implies that each parametric equation has the solution

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{p+r} \end{pmatrix} \in \left[\frac{a-\rho}{\lambda_{p+1}} I_{p+1}, \frac{b-\rho}{\lambda_{p+1}} I_{p+1} \right] \Big\langle I_1 - \frac{\lambda_1}{\lambda_{p+1}} I_{p+1}, \ldots, I_p - \frac{\lambda_p}{\lambda_{p+1}} I_{p+1}, \frac{|q|}{\lambda_{p+1}} I_{p+1},$$
$$I_{p+2} - \frac{\lambda_{p+2}}{\lambda_{p+1}} I_{p+1}, \ldots, I_{p+r} - \frac{\lambda_{p+r}}{\lambda_{p+1}} I_{p+1} \Big\rangle_{(p+1, r-1, 0, 0)}$$

Let us note Q' the modulo of this trapezoid congruence solution. By applying the definition expression (50) of parametric trapezoid congruences and expressing I_{p+1} in terms of Q'_{p+1}, we get

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{p+r} \end{pmatrix} \in \bigcup_{\frac{a-\rho}{|q|} Q'_{p+1} \underset{Q'}{\leq} X \underset{Q'}{\leq} \frac{b-\rho}{|q|} Q'_{p+1}} X \langle Q' \rangle_{(p+1, r-1)}$$

Following the definition of the basis relative order, it is equivalent to

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{p+r} \end{pmatrix} \in \bigcup_{a \leq \sigma \leq b} \frac{\sigma - \rho}{|q|} Q'_{p+1} \langle Q' \rangle_{(p+1, r-1)}$$

Then, introducing the $s+t$ last parameters

$$\begin{pmatrix} y_1 \\ \vdots \\ y_{p+r} \\ \xi_{p+r+1} \\ \vdots \\ \xi_n \end{pmatrix} \in \bigcup_{a \leq \sigma \leq b} \left(\frac{\sigma - \rho}{|q|} S_{p+1} + \xi_{p+r+1} I_{p+r+1} + \ldots + \xi_n I_n \right) \langle S^{p+r} \rangle_{(p+1, r-1)}$$

where Q' is transformed by adding $s+t$ rows of zeros to get the $p+r$ first columns of S.

Then the definition of ρ provides for the tuples ${}^t(y_1, \ldots, y_{p+r}, \xi_{p+r+1}, \ldots, \xi_n)$ the expression

$$\bigcup_{a \le \sigma \le b} \left(\frac{\sigma - \sum_{i=1}^{p} \lambda_i \xi_i}{|q|} S_{p+1} + \sum_{i=p+r+1}^{n} \xi_i \left(I_i - \frac{\lambda_i}{\lambda_{p+1}} I_{p+1} \right) \right) \left\langle S^{p+r} \right\rangle_{(p+1, r-1)}$$

In terms of the initial variables ${}^t(x_1, \ldots, x_{p+r}, x_{p+r+1}, \ldots, x_n)$ the parameterized solution set is

$$\bigcup_{a \le \sigma \le b} \left(\sum_{i=1}^{p} \xi_i \left(I_i - \frac{\lambda_i}{\lambda_{p+1}} I_{p+1} \right) + \frac{\sigma}{|q|} S_{p+1} + \sum_{i=p+r+1}^{n} \xi_i S_i \right) \left\langle S^{p+r} \right\rangle_{(p+1, r-1)}$$

and the solution set of all parametric equations is

$$\bigcup_{\substack{a \le \sigma \le b \\ 0 \le \xi_i \le p_i, i \le p+r+s \\ 0 \le \xi_i, i > p+r+s}} \left(\sum_{i=1}^{p} \xi_i S_i + \frac{\sigma}{|q|} S_{p+1} + \sum_{i=p+r+1}^{n} \xi_i S_i \right) \left\langle S^{p+r} \right\rangle_{(p+1, r-1)}$$

which is

$$\bigcup_{\substack{\frac{a}{|q|} S_{p+1} \underset{S}{\le} X \underset{S}{\le} \sum_{i=1}^{p} p_i S_i + \frac{b}{|q|} S_{p+1} + \sum_{i=p+r+1}^{p+r+s} p_i S_i \\ 0 \le \xi_i, i > p+r+s}} \left(X + \sum_{i=p+r+s+1}^{n} \xi_i S_i \right) \left\langle S^{p+r} \right\rangle_{(p+1, r-1)}$$

and finally

$$\bigcup_{\substack{A \underset{S}{\le} X \underset{S}{\le} B \\ O \underset{S^{p+r+s+1, n}}{\le} Y}} (X + Y) \left\langle S^{p+r} \right\rangle_{(p+1, r-1)}$$

The nullity of the dot product $(\lambda_1, \lambda_2, \ldots, \lambda_n).S_i$ for $i \in [p+2, p+r]$ is straightforward. $\square$

Geometrically, this lemma gives a method to calculate the intersection of a trapezoid congruence (a set of regularly dispersed trapezoids with at least one unbounded side) with a set of regularly dispersed sets of consecutive parallel hyperplanes (the solutions of the RLICE); the result is a trapezoid congruence. The same generalization as for lemma 82 is possible. For example the solution set of the RLICE $x - 2y \equiv \left[\frac{3}{2}, \frac{5}{2}\right] \langle 6 \rangle$ in the parametric trapezoid congruence $\left[\left(\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right), \left(\begin{smallmatrix} \frac{1}{2} \\ 0 \end{smallmatrix}\right)\right] \left\langle \begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix} \right\rangle_{(1,1,0,0)}$ is the trapezoid congruence $\left[\left(\begin{smallmatrix} 0 \\ \frac{-3}{4} \end{smallmatrix}\right), \left(\begin{smallmatrix} \frac{1}{2} \\ -1 \end{smallmatrix}\right)\right] \left\langle \begin{smallmatrix} 1 & 0 \\ \frac{1}{2} & -3 \end{smallmatrix} \right\rangle_{(2,0,0,0)}$ as is illustrated on the figure D.8.

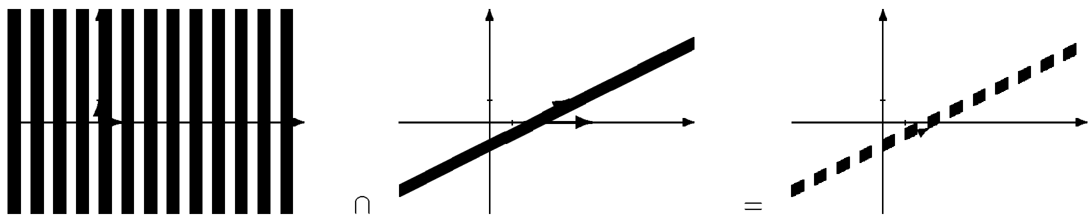


FIGURE D.9. Orthonormal trapezoid congruence and zero modulo RLICE intersection.

LEMMA 85 (ZERO MODULO RLICE IN A TRAPEZOID CONGRUENCE). *Let p, s, t be non negative integers, r a positive one such that $p + r + s + t = n$ and a a finite rational number. The solution of the RLICE*

$$a \leq \quad \lambda_1 x_1 + \lambda_2 x_2 + \ldots + \lambda_n x_n \quad \leq b \tag{53}$$

such that $\lambda_{p+r} \neq 0$ and a is finite in the trapezoid congruence:

$$[O, P] \langle I \rangle_{(p,r,s,t)}$$

is equal to the trapezoid congruence:

$$\left[aS_{p+r+s} \ , \ \sum_{i=1}^{p} p_i S_i + p_{p+r+s} S_{p+r} + \sum_{i=p+r+1}^{p+r+s-1} p_i S_i + (a + c(b-a)) S_{p+r+s} \right] \langle S \rangle_{(p,r-1,s+c,t+1-c)}$$

where:

$$S_i \quad = \quad \begin{cases} \frac{1}{\lambda_{p+r}} I_{p+r} & \text{if } i = p + r + s \\ \\ I_{p+r+s} - \frac{\lambda_{p+r+s}}{\lambda_{p+r}} I_{p+r} & \text{if } i = p + r \ \wedge \ s \neq 0 \\ \\ I_i - \frac{\lambda_i}{\lambda_{p+r}} I_{p+r} & \text{otherwise} \end{cases}$$

and c is 1 when b is finite and 0 otherwise. Moreover, $(\lambda_1, \lambda_2, \ldots, \lambda_n)$ is orthogonal to the rational rank columns of S.

PROOF. Following the same way as for proving the lemma 84, the problem is transformed into solving the RLICE

$$a - \rho \leq \quad \lambda_1 y_1 + \ldots + \lambda_p y_p + \lambda_{p+1} y_{p+1} + \ldots + \lambda_{p+r} y_{p+r} \quad \leq b - \rho \tag{54}$$

where $\rho = \sum_{i=1}^{p} \lambda_i \xi_i + \sum_{i=p+r+1}^{n} \lambda_i \xi_i$, in

$$\mathbb{Z}^p \times \mathbb{Q}^r \quad = \quad [O, O] \langle I \rangle_{(p,r,0,0)}$$

where $I = I(p+r)$. Lemma 83 implies that each parametric equation (54) has its solutions ${}^t(y_1, y_2, \ldots, y_{p+r})$ in

$$\left[\frac{a-\rho}{\lambda_{p+r}} I_{p+r}, \frac{a-\rho+(b-a)c}{\lambda_{p+r}} I_{p+r}\right] \left\langle I_1 - \frac{\lambda_1}{\lambda_{p+r}} I_{p+r}, \ldots, I_{p+r-1} - \frac{\lambda_{p+r-1}}{\lambda_{p+r}} I_{p+r}, \frac{1}{\lambda_{p+r}} I_{p+r} \right\rangle_{(p,r-1,c,1-c)}$$

Let us note Q' the modulo of the trapezoid congruence solution. By applying the definition of parametric trapezoid congruences and expressing I_{p+r} in terms of Q'_{p+r}, we get a first expression corresponding to the case where b is infinite

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{p+r} \end{pmatrix} \in \bigcup_{0 \leq \sigma} \left((a-\rho) Q'_{p+r} + \sigma Q'_{p+r}\right) \left\langle {Q'}^{p+r-1} \right\rangle_{(p,r-1)}$$

and a second expression corresponding to the case where b is finite

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{p+r} \end{pmatrix} \in \bigcup_{a \leq \sigma \leq b} (\sigma - \rho) Q'_{p+r} \left\langle {Q'}^{p+r-1} \right\rangle_{(p,r-1)}$$

Both cases are expressed in

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{p+r} \end{pmatrix} \in \bigcup_{a \leq \sigma \leq b} (\sigma - \rho) Q'_{p+r} \left\langle {Q'}^{p+r-1} \right\rangle_{(p,r-1)}$$

Then introducing the $s+t$ last parameters

$$\begin{pmatrix} y_1 \\ \vdots \\ y_{p+r} \\ \xi_{p+r+1} \\ \vdots \\ \xi_n \end{pmatrix} \in \bigcup_{a \leq \sigma \leq b} \left((\sigma - \rho) S_{p+r+s} + \xi_{p+r+1} I_{p+r+1} + \ldots + \xi_n I_n\right) \left\langle S^{p+r-1} \right\rangle_{(p,r-1)}$$

where Q' is transformed by adding $s+t$ rows of zeros getting the corresponding columns of S. Then the decomposition of ρ provides for the set of tuples ${}^t(y_1, \ldots, y_{p+r}, \xi_{p+r+1}, \ldots, \xi_n)$ the expression

$$\bigcup_{a \leq \sigma \leq b} \left(\left(\sigma - \sum_{i=1}^{p} \lambda_i \xi_i \right) S_{p+r+s} + \sum_{i=p+r+1}^{n} \xi_i \left(I_i - \frac{\lambda_i}{\lambda_{p+r}} I_{p+r} \right) \right) \left\langle S^{p+r-1} \right\rangle_{(p,r-1)}$$

In terms of the initial variables ${}^t(x_1, \dots, x_{p+r}, x_{p+r+1}, \dots, x_n)$ the parameterized solution set is

$$\bigcup_{a \le \sigma \le b} \left(\sum_{i=1}^{p} \xi_i \left(I_i - \frac{\lambda_i}{\lambda_{p+r}} I_{p+r} \right) + \xi_{p+r+s} S_{p+r} + \sum_{i=p+r+1}^{p+r+s-1} \xi_i S_i + \sigma S_{p+r+s} + \sum_{i=p+r+s+1}^{n} \xi_i S_i \right) \left\langle S^{p+r-1} \right\rangle_{(p,r-1)}$$

and the solution set of all parametric equations is

$$\bigcup_{\substack{a \le \sigma \le b \\ 0 \le \xi_i \le p_i, i \le p+r+s \\ 0 \le \xi_i, i > p+r+s}} \left(\sum_{i=1}^{p} \xi_i S_i + \xi_{p+r+s} S_{p+r} + \sum_{i=p+r+1}^{p+r+s-1} \xi_i S_i + \sigma S_{p+r+s} + \sum_{i=p+r+s+1}^{n} \xi_i S_i \right) \left\langle S^{p+r-1} \right\rangle_{(p,r-1)}$$

which is

$$\bigcup_{\substack{a S_{p+r+s} \underset{S}{\le} X \underset{S}{\le} a S_{p+r+s} + c(b-a) S_{p+r+s} + T \\ 0 \le \sigma, \quad 0 \le \xi_i, i > p+r+s}} \left(X + (1-c)\sigma S_{p+r+s} + \sum_{i=p+r+s+1}^{n} \xi_i S_i \right) \left\langle S^{p+r-1} \right\rangle_{(p,r-1)}$$

where $T = \sum_{i=1}^{p} p_i S_i + p_{p+r+s} S_{p+r} + \sum_{i=p+r+1}^{p+r+s-1} p_i S_i$, and finally

$$\bigcup_{\substack{A \underset{S}{\le} X \underset{S}{\le} B \\ O \underset{S^{p+r+s+c,\,n}}{\le} Y}} (X + Y) \left\langle S^{p+r-1} \right\rangle_{(p,r-1)}$$

The nullity of the dot product $(\lambda_1, \lambda_2, \dots, \lambda_n).S_i$ for $i \in [p+1, p+r-1]$ is straightforward. $\square$

Geometrically, this lemma has the same interpretation as the preceding one except that the set of regularly dispersed sets of consecutive parallel hyperplanes is changed into only one set of consecutive parallel hyperplanes (the solutions of the RLICE). The same generalization as for Lemma 83 is possible. For example the solution set of the zero modulo RLICE $x - 2y \equiv \left[\frac{3}{2}, \frac{5}{2}\right] \langle 0 \rangle$ in the parametric trapezoid congruence $\left[\left(\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right), \left(\begin{smallmatrix} \frac{1}{2} \\ 0 \end{smallmatrix}\right)\right] \left\langle \begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix} \right\rangle_{(1,1,0,0)}$ is the trapezoid congruence $\left[\left(\begin{smallmatrix} 0 \\ \frac{-3}{4} \end{smallmatrix}\right), \left(\begin{smallmatrix} \frac{1}{2} \\ -1 \end{smallmatrix}\right)\right] \left\langle \begin{smallmatrix} 1 & 0 \\ \frac{1}{2} & -3 \end{smallmatrix} \right\rangle_{(1,0,1,0)}$ as is illustrated on the figure D.9.

The problem of solving a RLICE in a trapezoid congruence is now reduced to the resolution of an equivalent equation in an orthonormal trapezoid congruence.

LEMMA 86 (RLICE IN A TRAPEZOID CONGRUENCE). *Let p, s, t be non negative integers, r a positive one, a a finite rational number and b a finite rational number if q is not zero. The solution of the RLICE*

$$\lambda_1 x_1 + \lambda_2 x_2 + \ldots + \lambda_n x_n \quad \equiv \quad [a, b] \langle q \rangle$$

in the parametric trapezoid congruence

$$[A, B] \langle S \rangle_{(p,r,s,t)}$$

such that there exists an integer $j \in [p+1, p+r]$ verifying $(\lambda_1, \lambda_2, \ldots, \lambda_n).S_j \neq 0$, is the trapezoid congruence

$$[A + SA', A + SB'] \langle SS' \rangle_{(p',r',s',t')}$$

where $[A', B'] \langle S' \rangle_{(p',r',s',t')}$ is the solution set of the RLICE

$$(\lambda_1, \lambda_2, \ldots, \lambda_n) S \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{p+r+s+t} \end{pmatrix} \equiv [a - (\lambda_1, \lambda_2, \ldots, \lambda_n) A, b - (\lambda_1, \lambda_2, \ldots, \lambda_n) A] \langle q \rangle$$

in the orthonormal trapezoid congruence:

$$[O, C] \langle I \rangle_{(p,r,s,t)}$$

with $B - A = SC$.
Moreover, $(\lambda_1, \lambda_2, \ldots, \lambda_n)$ is orthogonal to the rational rank columns of SS'.

PROOF. $X = {}^t(x_1, x_2, \ldots, x_n)$ is in S is equivalent to

$$\begin{cases} \lambda_1 x_1 + \lambda_2 x_2 + \ldots + \lambda_n x_n \equiv [a, b] \langle q \rangle \\ X = A + S(\Gamma + \Phi) \\ B - A = SC \\ \Phi \in \mathbb{Z}^p \times \mathbb{Q}^r \times \{0\}^s \times \mathbb{Q}_+^t \\ O \leq \Gamma \leq C \end{cases}$$

If we note $\Lambda = (\lambda_1, \lambda_2, \ldots, \lambda_n)$, we get a new equivalent system in terms of the unknown Y

$$\begin{cases} Y = \Gamma + \Phi \\ \Lambda SY \equiv [a - \Lambda A, b - \Lambda A] \langle q \rangle \\ X = A + SY \\ \Phi \in \mathbb{Z}^p \times \mathbb{Q}^r \times \{0\}^s \times \mathbb{Q}_+^t \\ B - A = SC \\ O \leq \Gamma \leq C \end{cases}$$

which is equivalent to

$$\left\{\begin{array}{rcl} Y & \in & [O,C]\,\langle I\rangle_{(p,r,s,t)} \\ SC & = & B-A \\ \Lambda SY & \equiv & [a-\Lambda A, b-\Lambda A]\,\langle q\rangle \\ X & = & A+SY \end{array}\right.$$

If $r > 0$ and at least one component of the vector ΛS of rank greater than r and smaller than $r+s$ is not null, then lemmas 84 and 85 provide the solution set $[A', B']\,\langle S'\rangle_{(p',r',s',t')}$ for Y and a new equivalent system is provided by

$$\left\{\begin{array}{l} X = A + SY \\ Y = A' + S'(\Gamma' + \Phi') \\ B' - A' = S'C' \\ \Phi' \in \mathbb{Z}^{p'} \times \mathbb{Q}^{r'} \times \{0\}^{s'} \times \mathbb{Q}^{t'} \\ O \leq \Gamma' \leq C' \end{array}\right.$$

Hence

$$\left\{\begin{array}{l} X = (A + SA') + SS'(\Gamma' + \Phi') \\ S(B' - A') = (SS')C' \\ \Phi' \in \mathbb{Z}^{p'} \times \mathbb{Q}^{r'} \times \{0\}^{s'} \times \mathbb{Q}^{t'} \\ O \leq \Gamma' \leq C' \end{array}\right.$$

which is the trapezoid congruence $[A + SA', A + SB']\,\langle SS'\rangle_{(p',r',s',t')}$.
The nullity of the dot product $(\lambda_1, \lambda_2, \ldots, \lambda_n).(SS')_i$ for a column of rational rank $(SS')_i$ is equivalent to the one of $((\lambda_1, \lambda_2, \ldots, \lambda_n)S).S'_i$ (because $p + r + s + t = p' + r' + s' + t'$) which is implied by lemmas 84 and 85. $\square$

Lemma 87 (Conversion to a RLICE system). *Let $T = [A, B]\,\langle S\rangle_{(p,r,s,t)}$ be a parametric trapezoid congruence, R a $(p + r + s + t, n)$ rational matrix such that $RS = I$. Then T is equal to the equational trapezoid congruence defined by the RLICE system with the unknowns $X = {}^t(x_1, x_2, \ldots, x_n)$*

$$\left\{\begin{array}{ll} ({}^tR)_i\,.X \equiv [({}^tR)_i\,.A, ({}^tR)_i\,.B]\,\langle 1\rangle & \textit{if} \quad i \in [1,p] \\[2ex] ({}^tR)_i\,.A \leq ({}^tR)_i\,.X \leq ({}^tR)_i\,.B & \textit{if} \quad i \in [p+r+1, p+r+s] \\[2ex] ({}^tR)_i\,.A \leq ({}^tR)_i\,.X & \textit{if} \quad i \in [p+r+s+1, p+r+s+t] \end{array}\right.$$

Proof. R exists since the elements of the collection $(S_i)_{i\in[1,p+r+s+t]}$ are linearly independent, thus the RLICE system is non singular.

The elements X of T are defined by the system

$$\begin{cases} X = A + S(\Gamma + \Phi) \\ B - A = SC \\ \Phi \in \mathbb{Z}^p \times \mathbb{Q}^r \times \{0\}^s \times \mathbb{Q}_+^t \\ O \leq \Gamma \leq C \end{cases}$$

which is equivalent to

$$\begin{cases} RX = RA + (\Gamma + \Phi) \\ R(B - A) = C \\ \Phi \in \mathbb{Z}^p \times \mathbb{Q}^r \times \{0\}^s \times \mathbb{Q}_+^t \\ O \leq \Gamma \leq C \end{cases}$$

The p first rows of $RX = RA + (\Gamma + \Phi)$ provide the RLICEs with modulo one, the r next rows are simply ignored because of their rational component of Φ, the next s rows provide the double inequations and finally the t last rows, the inequalities. $\square$

For example, the trapezoid congruence $\left[\left(\begin{smallmatrix} 0 \\ \frac{-3}{4} \end{smallmatrix}\right), \left(\begin{smallmatrix} \frac{1}{2} \\ -1 \end{smallmatrix}\right)\right] \left\langle \begin{smallmatrix} 0 & 1 \\ -3 & \frac{1}{2} \end{smallmatrix} \right\rangle_{(1,0,1,0)}$ is equivalent to the RLICE system

$$\begin{cases} \frac{1}{6}x - \frac{1}{3}y \equiv \left[\frac{1}{4}, \frac{5}{12}\right] \langle 1 \rangle \\ x \equiv \left[0, \frac{1}{2}\right] \langle 0 \rangle \end{cases}$$

Finally we are now able to prove the theorem 78.

Proof. [of the equivalence of parametric and equational representations] Lemma 87 provides the equational trapezoid congruence equal to a given parametric trapezoid congruence. For the other way, let us take a non singular RLICE system

$$\Sigma = \left(\Lambda_i . \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \equiv [a_i, b_i] \langle q_i \rangle \right)_{i \in [1,n]}$$

We suppose that all the lower bounds of the interval congruences of the system are finite and that if their modulo is non zero that their upper bounds are finite too. Indeed, if it is not the case, equivalent systems verifying these conditions are easily determined. The RLICEs with an interval congruence of infinite lower and upper bounds are just removed and those which have only one infinite bound are also removed if the corresponding modulo is non zero and the RLICEs are inversed otherwise. The parametric corresponding trapezoid congruence is obtained by an incremental resolution of Σ in $\mathbb{Q}^n$. Lemma 86 solves the first RLICE

$$\Sigma_1 = \Lambda_1 . \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \equiv [a_1, b_1] \langle q_1 \rangle$$

in $\mathbb{Q}^n = [O, O]\langle I\rangle_{(0,n,0,0)}$, giving the parametric trapezoid congruence T_1 whose rational rank is greater than $n-1$ and whose rational rank columns are orthogonal to Λ_1. Λ_1 and Λ_2 are linearly independent, hence Λ_2 is not orthogonal to the rational rank columns of T_1 and the RLICE Σ_2 is solvable in T_1 by lemma 86. After $n-1$ iterations of this process, the obtained parametric trapezoid congruence T_n is the parametric representation of the system Σ. The equivalence between parametric and equational trapezoid congruences is thus proved. $\square$

CHAPTER VI

ABSTRACT INTERPRETATION OF TRAPEZOID CONGRUENCES

This chapter is devoted to the design of some abstract interpretations using the two domains described in the chapter V. First the connection between these two domains is provided in section 1; its particular features are expressed in terms of the general abstract interpretation framework [CC92b]. Then the approximate operators on the abstract domain are determined together with the widening operator in the section 2. Finally, the section 3 provides the abstract statements and is ended with a complete analysis example.

1. Semantic operators

The concrete domain RCC and the abstract one TC (with two dual definitions) are designed in chapter V. We bind them now using a pair of abstraction and concretization functions in order to give the meaning of the abstract elements and to prove that their respective orders are coherent.

1.1. Soundness relation.

DEFINITION 88 (THE SOUNDNESS RELATION σ). The *soundness relation* σ on $\mathbb{P}(\mathbb{Z}^n)\times TC$ is defined by

$$\sigma \stackrel{\text{def}}{=} \{(P,T),\ P \subseteq T\}$$

1.2. Abstraction. To abstract a relational coset congruence is to find a rational superset of it. To be as accurate as possible the abstraction should not add any new integer solution to the original system.

DEFINITION 89 (ABSTRACTION $\alpha^{\bowtie}$). The *abstraction function* is defined by:

$$\begin{array}{rrcl} \alpha^{\bowtie} : & RCC & \rightarrow & TC \\ & (\Delta_i.X \equiv \theta_i.[l_i,u_i]\langle m_i\rangle)_{i\in[1,p]} & \mapsto & (\Delta_i.X \equiv \alpha(\theta_i.[l_i,u_i]\langle m_i\rangle))_{i\in[1,p]} \end{array}$$

where the abstraction function α over coset congruences is given in definition 41.

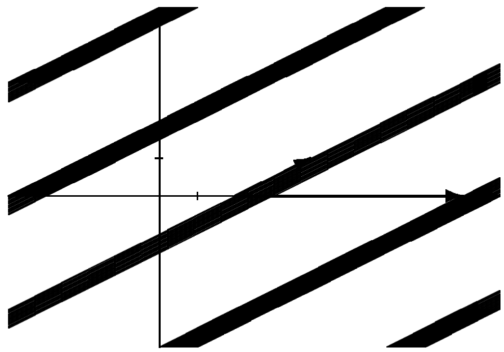


FIGURE VI.10. The abstraction of the relational coset congruence of the figure V.3.

Notice that an abstraction relation α only requiring that $\alpha\left(\theta_i.\left[l_i,u_i\right]\langle m_i\rangle, \left[a_i,b_i\right]\langle q_i\rangle\right) \Leftrightarrow \theta_i.\left[l_i,u_i\right]\langle m_i\rangle = \left[a_i,b_i\right]\langle q_i\rangle \cap \mathbb{Z}$ would have been sufficient but has not been chosen for the sake of simplicity.

For example, the abstraction $\alpha^{\bowtie}((x-2y \equiv 1.[2,3]\langle 6\rangle))$ of the relational coset congruence C_0 of the figure V.3 is the trapezoid congruence $(x-2y \equiv [2,3]\langle 6\rangle)$ which is represented on Figure VI.10.

1.3. Concretization. The concretization function $\gamma^{\bowtie}$ is first defined on a subset of TC that is the trapezoid congruences equationaly defined with integer coefficients. It is then implicitly extended to TC since every element of TC is equivalent to an element defined using integer coefficients. Such RLICEs defining equational trapezoid congruences are obtained by multiplying their coefficients with the least common multiple of their denominators. Hence the functional property feature of the concretization function is preserved by that preliminary multiplication.

DEFINITION 90 (CONCRETIZATION $\gamma^{\bowtie}$). The *concretization function* $\gamma^{\bowtie}$ associates to the trapezoid congruence $T = (\Delta_i.X \equiv [a_i,b_i]\langle q_i\rangle)_{i\in[1,p]}$ the relational coset congruence

$$\gamma^{\bowtie}(T) = \left(\frac{1}{g_i}\Delta_i.X \equiv \begin{cases} 1.[1,0]\langle 1\rangle & \text{if } \left\lceil \frac{l_i}{g_i}\right\rceil > \left\lfloor \frac{u_i}{g_i}\right\rfloor \\ \left\|\theta_i.\left[\left\lceil \frac{l_i}{g_i}\right\rceil, \left\lfloor \frac{u_i}{g_i}\right\rfloor\right]\left\langle \frac{m_i}{g_i}\right\rangle\right\| & \text{otherwise} \end{cases}\right)_{i\in[1,p]}$$

where $(\Delta_i)_{i\in[1,p]}$ is a collection of integer tuples of $\mathbb{Z}^n$, $g_i = \gcd(\Delta_i, m_i)$ and $\theta_i.[l_i,u_i]\langle m_i\rangle = \gamma\left([a_i,b_i]\langle q_i\rangle\right)$.

The preceding definition holds because the resulting system of congruence equations always is a RCC (the coset congruences of the LCCEs are normalized and the linear coefficients of each LCCE are prime with the corresponding coset congruence modulo). Indeed, the modulos of a coset congruence and of its normalization have different absolute values if and only if they are equal to the empty set or to $\mathbb{Z}$. It is easy to see that $\theta_i.\left[\left\lceil \frac{l_i}{g_i}\right\rceil, \left\lfloor \frac{u_i}{g_i}\right\rfloor\right]\left\langle \frac{m_i}{g_i}\right\rangle$ equals $\mathbb{Z}$ if and only if $\theta_i.[l_i,u_i]\langle m_i\rangle$ is equal to $\mathbb{Z}$ too[1] and, in this case, they both are $1.[0,0]\langle 1\rangle$. Finally notice that $\theta_i.\left[\left\lceil\frac{l_i}{g_i}\right\rceil,\left\lfloor\frac{u_i}{g_i}\right\rfloor\right]\left\langle\frac{m_i}{g_i}\right\rangle$ is never empty because $\left\lceil\frac{l_i}{g_i}\right\rceil\le\left\lfloor\frac{u_i}{g_i}\right\rfloor$; hence the gcd of the linear coefficients of the LCCE and of their modulo always is 1. The coset congruences of the LCCEs are normalized.

Theorem 91 (Correctness of $\gamma^{\bowtie}$). *The meaning $\gamma^{\bowtie}(T)$ of a trapezoid congruence T is its intersection with $\mathbb{Z}^n$.*

Proof. Each constitutive RLICE of the trapezoid congruence $(\Delta_i.X\equiv[a_i,b_i]\langle q_i\rangle)_{i\in[1,p]}$ corresponds to the system of linear congruence equations

$$\bigvee_{a_i\le x_0\le b_i}\Delta_i.X\equiv x_0\mod(q_i)$$

and, if $\theta_i.[l_i,u_i]\langle m_i\rangle=\gamma([a_i,b_i]\langle q_i\rangle)$, then the integer solution set of this system is equal to the one of the system with integer unknowns

$$\bigvee_{l_i\le k\le u_i}\Delta_i.X\equiv k\theta_i\mod(m_i)\tag{55}$$

because the linear coefficients are integers and $\left(\bigcup_{a_i\le x_0\le b_i}x_0\langle q_i\rangle\right)\cap\mathbb{Z}=\bigcup_{l_i\le k\le u_i}k\theta_i\langle m_i\rangle$. Moreover it is equal to the solution set of the system provided by only keeping in the disjunction system (55) the linear congruence equations with solutions. If $g_i=\gcd(\Delta_i,m_i)$, these ones are characterized by $k\theta_i\in\langle g_i\rangle$. The lemma 21 provides the result. $\square$

In addition to the preceding concretization algorithm, first if the coset congruence of an obtained LCCE is empty then the other LCCEs are removed and, finally, the LCCEs whose coset congruences are equal to $\mathbb{Z}$ are removed. Hence the resulting trapezoid congruence meaning has the property that, excepting the case where it is equal to one LCCE with an empty coset congruence, all its constitutive linear congruence equation solution sets are non empty.

For example the meaning of the trapezoid congruence

$$\left(6x+12y\equiv\left[\frac{1}{2},\frac{5}{6}\right]\left\langle\frac{12}{7}\right\rangle,3x-5y\equiv\left[\frac{1}{4},\frac{7}{4}\right]\left\langle\frac{9}{4}\right\rangle\right)$$

is empty since $\gamma\left(\left[\frac{1}{2},\frac{5}{6}\right]\left\langle\frac{12}{7}\right\rangle\right)=5.[7,8]\langle 12\rangle$, $\gcd(6,12,12)=6$ and $\left\lceil\frac{7}{6}\right\rceil>\left\lfloor\frac{8}{6}\right\rfloor$ while the meaning of the trapezoid congruence

$$\left(x-2y\equiv\left[\frac{7}{4},\frac{10}{3}\right]\langle 6\rangle,3x-9y\equiv\left[\frac{1}{2},\frac{7}{5}\right]\left\langle\frac{6}{5}\right\rangle\right)$$

is the relational coset congruence

$$(x-2y\equiv 1.[2,3]\langle 6\rangle)$$

[1] It is a direct consequence of the proof of the theorem on the correctness of $\gamma_{\bowtie}$.

Indeed, $\gamma\left(\left[\frac{1}{2},\frac{7}{5}\right]\left\langle\frac{6}{5}\right\rangle\right) = 1.\,[5,9]\,\langle 6\rangle$ and $\left\|1.\left[\left\lceil\frac{5}{3}\right\rceil,\left\lfloor\frac{9}{3}\right\rfloor\right]\left\langle\frac{6}{3}\right\rangle\right\| = 1.\,[0,0]\,\langle 1\rangle = \mathbb{Z}$; hence the RLICE $3x-9y \equiv \left[\frac{1}{2},\frac{7}{5}\right]\left\langle\frac{6}{5}\right\rangle$ is redundant. Finally the resulting relational coset congruence corresponds to the cosets $\binom{2}{0}\left\langle\begin{smallmatrix}2&0\\1&3\end{smallmatrix}\right\rangle_{(2,0)}$ and $\binom{3}{0}\left\langle\begin{smallmatrix}2&0\\1&3\end{smallmatrix}\right\rangle_{(2,0)}$ represented on Figure V.3.

1.4. Characteristics of the connection $(\alpha^{\bowtie},\gamma^{\bowtie})$. When the meaning of T is not empty, each equation of $\gamma^{\bowtie}(T)$ is a disjunction of $\left\lfloor\frac{u_i}{g_i}\right\rfloor - \left\lceil\frac{l_i}{g_i}\right\rceil$ (which is possibly infinite) rational linear congruence equations; hence $\gamma^{\bowtie}(T)$ is the disjunction of

$$\left(\left\lfloor\frac{u_1}{g_1}\right\rfloor - \left\lceil\frac{l_1}{g_1}\right\rceil\right)\left(\left\lfloor\frac{u_2}{g_2}\right\rfloor - \left\lceil\frac{l_2}{g_2}\right\rceil\right)\cdots\left(\left\lfloor\frac{u_p}{g_p}\right\rfloor - \left\lceil\frac{l_p}{g_p}\right\rceil\right)$$

rational linear congruence equation systems. Following [Gra91a], we see that all the above mentioned systems have the same kind of solution $A\,\langle M\rangle_{(p,0)}$ where $\langle M\rangle_{(p,0)}$ is the solution set of the congruence equation system

$$(\Delta_i.X \equiv 0 \mod (m_i))_{i\in[1,p]}$$

Hence S is a set of linear cosets of modulo $\langle M\rangle_{(p,0)}$.

Proposition 92 (Characterization of $\gamma^{-1}(\mathbb{Z}^n)$). *Let $(\Delta_i.X \equiv [a_i,b_i]\,\langle q_i\rangle)_{i\in[1,p]}$ be a trapezoid congruence; it contains $\mathbb{Z}^n$ if and only if for all $i \in [1,p]$*

$$0 < \left|\frac{m_i}{g_i}\right| \le \left\lfloor\frac{u_i}{g_i}\right\rfloor - \left\lceil\frac{l_i}{g_i}\right\rceil + 1$$

where $\theta_i.\,[l_i,u_i]\,\langle m_i\rangle = \gamma\left([a_i,b_i]\,\langle q_i\rangle\right)$

Proof. This is a direct consequence of the propositions 69 and 17 where the cases corresponding to a zero modulo do not have to be considered because of the special kind of interval congruences used in the definition 66 of trapezoid congruences. □

Proposition 93 (Structure of $(\alpha^{\bowtie},\gamma^{\bowtie})$). *The pair of maps $(\alpha^{\bowtie},\gamma^{\bowtie})$ is not a Galois connection.*

Proof. Since the relational abstraction $\alpha^{\bowtie}$ and the relational concretization $\gamma^{\bowtie}$ partially coincide respectively with the non relational abstraction α and concretization γ when they are considered in one dimension, the counter example justifying the proposition 47 is used to prove the above proposition. □

1.5. Normalization. Intuitively, the normalization process corresponds here, given a parametric trapezoid congruence T, to find a new trapezoid congruence T' with the same modulo and such that its representative is the smallest one preserving the meaning of T ($T \cap \mathbb{Z}^n = T' \cap \mathbb{Z}^n$). A simple idea consists in reducing as much as possible the representative width of each equational trapezoid congruence constitutive RLICE. This is done by following the property stating that

$$\lambda_1 x_1 + \lambda_2 x_2 + \cdots + \lambda_n x_n \equiv a \mod (q)$$

has integer solution if and only if $\gcd(\lambda_1, \lambda_2, \ldots, \lambda_n, q)$ divides a. This solves well our normalization problem when the equational trapezoid congruence consists in only one RLICE, but not more. Indeed, the integer solutions of one RLICE preventing the reduction of its representative may not be solutions of another RLICE of the considered trapezoid congruence. Hence the representative reduction can go further without changing the meaning of the initial trapezoid congruence. We are not able for the moment to provide a normalization algorithm satisfying our initial intuitive idea, but only a partial normalization involving the abstraction and concretization function. Such a construction which is in fact equivalent to the one described above (reducing the RLICE representatives) allows to take advantage of the possible improvements of the concretization process for special cases.

Definition 94 (Normalization $\eta^{\bowtie}$). The *normalization operator* $\eta^{\bowtie}$ on the set of trapezoid congruences of $\mathbb{Q}^n$ is defined by

$$\eta^{\bowtie} \stackrel{\text{def}}{=} \alpha^{\bowtie} \circ \gamma^{\bowtie}$$

As for the non relational normalization operator η on interval congruences, $\eta^{\bowtie}$ generally replaces a trapezoid congruence with a non comparable one. This is a consequence of the non reductive normalization $\| \ \|$ on CC involved in the concretization γ of interval congruences, itself involved in the concretization of trapezoid congruences $\gamma^{\bowtie}$. Hence a normalized trapezoid congruence is possibly smaller and has the same meaning as the initial one.

For example the trapezoid congruence

$$\left(x - 3y \equiv \left[\frac{3}{4}, \frac{5}{4}\right] \left\langle \frac{9}{7} \right\rangle, 8x - 2y \equiv \left[\frac{17}{2}, 20\right] \langle 33 \rangle\right)$$

is normalized into

$$\left(x - 3y \equiv \left[\frac{1}{2}, \frac{3}{2}\right] \left\langle \frac{9}{2} \right\rangle, 8x - 2y \equiv [9, 20] \langle 33 \rangle\right)$$

2. Abstract operators

The goal of this section is to deal with the operators on the abstract domain that are needed for the analysis. Exact meet and join algorithms are not definable since TC is not a complete lattice, hence only safe approximations of them are defined.

2.1. Conversion. As illustrated in the definition of the approximate join operator, the only really needed conversion consists in finding an approximation of the smallest trapezoid congruence of TC containing a given trapezoid congruence when the new modulo divides the one of the original trapezoid congruence.

LEMMA 95. *Let $\langle S\rangle_{(p,r,s,t)}$ be a trapezoid congruence and $\langle Q\rangle_{(p',r')}$ a divisor of $\langle S^{p+r}\rangle_{(p,r)}$. There exists a shape $\langle S'\rangle_{(p',r',s',t')}$ such that ${S'}^{p'+r'} = Q$ and $S = S'P$ where P has the pattern*

$$P = \begin{array}{c} \begin{array}{cccc} p & p+r & p+r+s & p+r+s+t \\ \downarrow & \downarrow & \downarrow & \downarrow \end{array} \\ \left(\begin{array}{cc|cc} E & 0 & & \\ \hline & & & 0 \\ 0 & & & \hline F \end{array}\right) \end{array} \begin{array}{l} \leftarrow p' \\ \leftarrow p'+r' \\ \leftarrow p'+r'+s' \\ \leftarrow p'+r'+s'+t' \end{array} \tag{56}$$

where E is a (p', p) block of integer coefficients and 0 denotes a block of zero coefficients.

PROOF. The $p + r$ first columns of P are just a consequence of the proposition 15. Then it is sufficient to complete the $s' + t'$ columns of S' by taking linearly independent vectors of $S^{p+r+1,p+r+s+t}$. It is possible to choose s' in order to maximize the height of the block of zero coefficients above F. $\square$

DEFINITION 96 (SHAPE CONVERSION Cast). Let $T = [A, B]\,\langle S\rangle_{(p,r,s,t)}$ be a trapezoid congruence and $\langle S'\rangle_{(p',r',s',t')}$ a shape such that $\left\langle {S'}^{p'+r'}\right\rangle_{(p',r')}$ is a divisor of $\langle S^{p+r}\rangle_{(p,r)}$ and $S = S'P$ where P has the pattern (56) and in addition the coefficients of the block F are positive. The *cast* of T to the shape $\langle S'\rangle_{(p',r',s',t')}$ is defined by

$$\mathrm{Cast}_{\langle S'\rangle_{(p',r',s',t')}}(T) \stackrel{\text{def}}{=} [A + S'G, A + S'D]\,\langle S'\rangle_{(p',r',s',t')}$$

where C is such that $SC = B - A$ and G and D are rational $(p' + r' + s' + t')$uples such that

$$[g_j, d_j] \stackrel{\text{def}}{=} \begin{cases} \displaystyle\sum_{i=1}^{p+r+s} p_{ji} * [0, c_i] + \sum_{i=p+r+s+1}^{p+r+s+t} p_{ji} * [0, +\infty] & \text{if } 1 \le j \le p' \\ [0, 0] & \text{if } 1 \le j - p' \le r' \\ \displaystyle\sum_{i=p+r+1}^{p+r+s} p_{ji} * [0, c_i] & \text{if } 1 \le j - p' - r' \le s' + t' \end{cases}$$

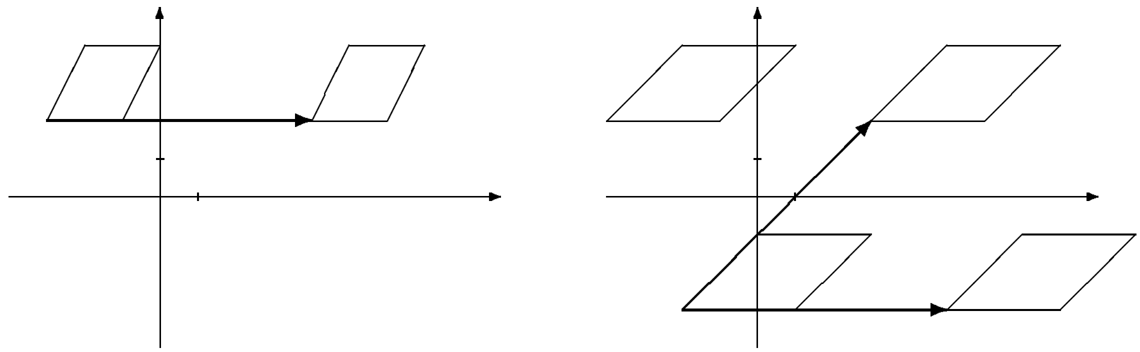


FIGURE VI.11. Trapezoid congruence conversion.

---

where $p * [0, +\infty] = [0, 1]$ by convention if $p \neq 0$.

PROPOSITION 97 (EXTENSIVITY OF Cast). *Let $T = [A, B]\langle S\rangle_{(p,r,s,t)}$ be a trapezoid congruence and $\langle S'\rangle_{(p',r',s',t')}$ a shape such that $\mathrm{Cast}_{\langle S'\rangle_{(p',r',s',t')}}(T)$ exists, then*

$$T \subseteq \mathrm{Cast}_{\langle S'\rangle_{(p',r',s',t')}}(T)$$

The proof is just a verification. The cast operation is illustrated on figure VI.11 where

$$\mathrm{Cast}_{\left\langle \begin{smallmatrix} 7 & 5 \\ 0 & 5 \end{smallmatrix}\right\rangle_{(2,0,0,0)}}\left([\left(\begin{smallmatrix} -3 \\ 2 \end{smallmatrix}\right), \left(\begin{smallmatrix} 0 \\ 4 \end{smallmatrix}\right)]\left\langle \begin{smallmatrix} 7 & 1 \\ 0 & 2 \end{smallmatrix}\right\rangle_{(1,0,1,0)}\right) \quad = \quad [\left(\begin{smallmatrix} -2 \\ -3 \end{smallmatrix}\right), \left(\begin{smallmatrix} 3 \\ -1 \end{smallmatrix}\right)]\left\langle \begin{smallmatrix} 7 & 5 \\ 0 & 5 \end{smallmatrix}\right\rangle_{(2,0,0,0)}$$

Unfortunately, the cast of T to a shape $\langle S'\rangle_{(p',r',s',t')}$ whose modulo divides the one of T is not always possible. The t last vectors of the shape of T have to be linear combinations of the columns of S but with positive coefficients relatively to the t' last columns of S'. If it is not the case, a new shape for which the shape cast is possible is easily provided by an extension of the linear part of the modulo of the shape $\langle S'\rangle_{(p',r',s',t')}$. Indeed, adding to the linear part of $\langle S'\rangle_{(p',r',s',t')}$ the vectors of its t' last columns corresponding to the rows of the block F containing negative coefficients provides a divisor of the modulo of $\langle S'\rangle_{(p',r',s',t')}$ and of the modulo of $\langle S\rangle_{(p,r,s,t)}$ too. Hence, given a divisor Q of the modulo of $\langle S\rangle_{(p,r,s,t)}$ it is always possible to find a divisor Q' of Q with the same non linear part dividing the modulo of $\langle S\rangle_{(p,r,s,t)}$ and allowing the shape cast of T to a shape of modulo Q'.

The following definition is a generalization of the greatest common divisor on linear subgroups to trapezoid congruence shapes.

Theorem & Definition 98 (Shape join $\bigwedge$). *Let $\langle S_1\rangle_{(p_1,r_1,s_1,t_1)}$ and $\langle S_2\rangle_{(p_2,r_2,s_2,t_2)}$ be two shapes. There exists a shape $\langle S\rangle_{(p,r,s,t)}$ such that its modulo $\langle S^{p+r}\rangle_{(p,r)}$ divides and has the same non linear part as the greatest common divisor of the modulos $\left\langle S_1^{p_1+r_1}\right\rangle_{(p_1,r_1)}$ and $\left\langle S_2^{p_2+r_2}\right\rangle_{(p_2,r_2)}$, and such that the casts of trapezoid congruences of shapes $\langle S_1\rangle_{(p_1,r_1,s_1,t_1)}$ and $\langle S_2\rangle_{(p_2,r_2,s_2,t_2)}$ to the shape $\langle S\rangle_{(p,r,s,t)}$ exist. $\langle S\rangle_{(p,r,s,t)}$ is noted*

$$\langle S_1\rangle_{(p_1,r_1,s_1,t_1)} \bigwedge \langle S_2\rangle_{(p_2,r_2,s_2,t_2)}$$

Proof. It is sufficient to build the linear part of the shape join such that it generates the t_1 last columns of the first shape and the t_2 last columns of the second. Then the shape cast is always possible. $\square$

The shape cast and shape join are the basic steps of the general algorithm taking two trapezoid congruences $T_1 = [A_1, B_1]\langle S_1\rangle_{(p_1,r_1,s_1,t_1)}$ and $T_2 = [A_2, B_2]\langle S_2\rangle_{(p_2,r_2,s_2,t_2)}$ and determining two new trapezoid congruences $T_1' = [A_1', B_1']\langle S\rangle_{(p,r,s,t)}$ and $T_2' = [A_2', B_2']\langle S\rangle_{(p,r,s,t)}$ of identical shape and respectively containing T_1 and T_2 where

$$\begin{aligned}
\langle S\rangle_{(p,r,s,t)} &= \langle S_1\rangle_{(p_1,r_1,s_1,t_1)} \bigwedge \langle S_2\rangle_{(p_2,r_2,s_2,t_2)}\\
T_1' &= \mathrm{Cast}_{\langle S\rangle_{(p,r,s,t)}}(T_1)\\
T_2' &= \mathrm{Cast}_{\langle S\rangle_{(p,r,s,t)}}(T_2)
\end{aligned}$$

2.2. Join. The approximate join operator over trapezoid congruences is based on the use of two elementary join operators which are homothetic and congruence-like join. These two basic operators both take trapezoid congruences with the same shape and different representatives. Hence a conversion of the two operands of a join operation to the same shape is necessary before running these operators. This conversion process is provided by the shape cast and shape join. The new common shape is based on the greatest common divisor of the two original trapezoid congruences modulos.

Homothetic join on TC

The next definition establishes how to join two trapezoid congruences with the same shape. More precisely, it gives one possible trapezoid congruence containing two trapezoid congruences of same modulo.

Recall that lemma 87 states that the conversion of two parametrical trapezoid congruences with the same shape to their equational representation are equational trapezoid congruences with identical associated homogeneous equation systems.

Definition 99 (Homothetic join $\sqcup_\Diamond$). Let $T_1 = (\Lambda_i.X \equiv [a_{1i}, b_{1i}]\langle q_i\rangle)_{i\in[1,n]}$ and $T_2 = (\Lambda_i.X \equiv [a_{2i}, b_{2i}]\langle q_i\rangle)_{i\in[1,n]}$ be two prime equational trapezoid congruences with the same associated parametric shape.

$$T_1 \sqcup_\Diamond T_2 \stackrel{\text{def}}{=} (\Lambda_i.X \equiv [a_{1i}, b_{1i}]\langle q_i\rangle \sqcup [a_{2i}, b_{2i}]\langle q_i\rangle)_{i\in[1,n]}$$

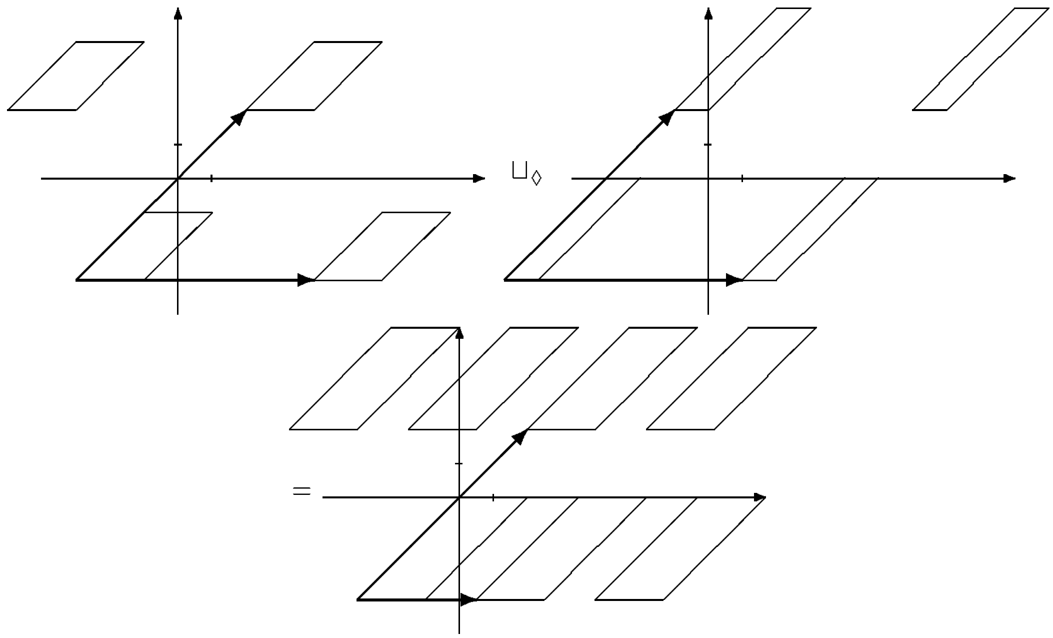


FIGURE VI.12. Homothetic join $\sqcup_{\Diamond}$.

PROPOSITION 100 ($\sqcup_{\Diamond}$ IS GREATER THAN $\cup$). *Let T_1 and T_2 be two trapezoid congruences*

$$T_1 \cup T_2 \quad \subseteq \quad T_1 \sqcup_{\Diamond} T_2$$

The proof is just a verification.

The basic idea resulting from the definition of that join operator is illustrated on figure VI.12 with an example where

$$\left[\begin{pmatrix} -3 \\ -3 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix}\right] \left\langle \begin{matrix} 7 & 5 \\ 0 & 5 \end{matrix} \right\rangle_{(2,0,0,0)} \quad \sqcup_{\Diamond} \quad \left[\begin{pmatrix} 1 \\ -3 \end{pmatrix}, \begin{pmatrix} 5 \\ 0 \end{pmatrix}\right] \left\langle \begin{matrix} 7 & 5 \\ 0 & 5 \end{matrix} \right\rangle_{(2,0,0,0)}$$

is equal to

$$\left[\begin{pmatrix} -3 \\ -3 \end{pmatrix}, \begin{pmatrix} 2 \\ 0 \end{pmatrix}\right] \left\langle \begin{matrix} \frac{7}{2} & 5 \\ 0 & 5 \end{matrix} \right\rangle_{(2,0,0,0)}$$

Congruence-like join on TC

An alternative to the homothetic join $\sqcup_{\Diamond}$ naturally defined for two trapezoid congruences of same modulo is the congruence join $\sqcup_{\nearrow}$ that first converts them to a divisor of their common shape following the definition 96 and then makes an homothetic join. The new modulo is chosen such that the converted representatives overlap.

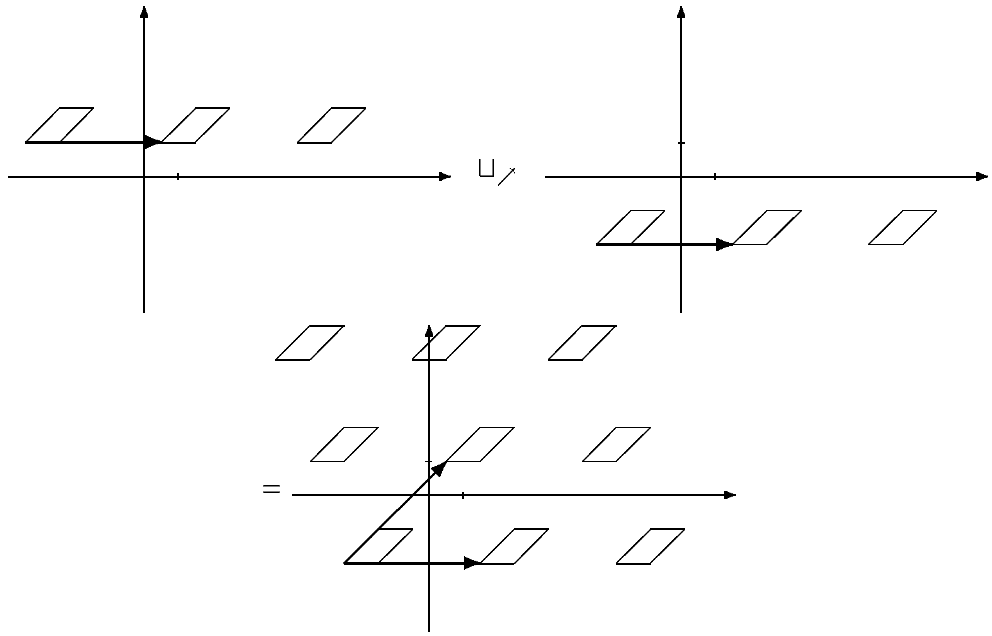


FIGURE VI.13. Congruence-like join $\sqcup_{\nearrow}$.

DEFINITION 101 (CONGRUENCE-LIKE JOIN $\sqcup_{\nearrow}$). Let $T_1 = [A_1, B_1] \langle S \rangle_{(p,r,s,t)}$ and $T_2 = [A_2, B_2] \langle S \rangle_{(p,r,s,t)}$ be two non comparable trapezoid congruences with the same shape. The *congruence-like join* $T_1 \sqcup_{\nearrow} T_2$ of T_1 and T_2 is defined by

$$T_1 \sqcup_{\nearrow} T_2 \quad \stackrel{\text{def}}{=} \quad \mathrm{Cast}_{\langle S' \rangle_{(p')} r' s' t'}(T_1) \sqcup_{\lozenge} \mathrm{Cast}_{\langle S' \rangle_{(p',r',s',t')}}(T_2)$$

where

$$\begin{aligned}
\Omega &= A_2 - A_1 + \frac{1}{2}\left(\sum_{i=1}^{p}(c_{2i} - c_{1i})S_i + \sum_{i=p+r+1}^{p+r+s}(c_{2i} - c_{1i})S_i\right) \\
\langle Q \rangle_{(u,v)} &\stackrel{\text{def}}{=} \gcd\left(\langle S^{p+r} \rangle_{(p,r)}, \langle \Omega \rangle_{(1,0)}\right) \\
\langle S' \rangle_{(p',r',s',t')} &= \langle Q \rangle_{(u,v,0,0)} \bigwedge \langle S \rangle_{(p,r,s,t)}
\end{aligned}$$

Notice that this definition implies that $\left\langle S'^{p'+r'} \right\rangle_{(p',r')}$ divides $\langle Q \rangle_{(u,v)}$ and the shape conversion of T_1 and T_2 to the shape $\left\langle S'^{p'+r'} \right\rangle_{(p')} r'$ exists.

PROPOSITION 102 ($\sqcup_{\nearrow}$ IS GREATER THAN $\cup$). *Let T_1 and T_2 be two trapezoid congruences*

$$T_1 \cup T_2 \quad \subseteq \quad T_1 \sqcup_{\nearrow} T_2$$

PROOF. It is a direct consequence of the extensivity of Cast and of the proposition 100. $\square$

The example of figure VI.13 illustrates the congruence-like join.

$$\left[\begin{pmatrix} \frac{1}{2} \\ 1 \end{pmatrix}, \begin{pmatrix} \frac{5}{2} \\ 2 \end{pmatrix}\right] \left\langle \begin{matrix} 4 & 2 \\ 0 & 2 \end{matrix} \right\rangle_{(1,0,1,0)} \quad \sqcup_{\nearrow} \quad \left[\begin{pmatrix} \frac{-5}{2} \\ -2 \end{pmatrix}, \begin{pmatrix} \frac{-1}{2} \\ -1 \end{pmatrix}\right] \left\langle \begin{matrix} 4 & 2 \\ 0 & 2 \end{matrix} \right\rangle_{(1,0,1,0)}$$

is equal to

$$\left[\begin{pmatrix} \frac{-5}{2} \\ -2 \end{pmatrix}, \begin{pmatrix} \frac{-1}{2} \\ -1 \end{pmatrix}\right] \left\langle \begin{matrix} 4 & 3 \\ 0 & 3 \end{matrix} \right\rangle_{(2,0,0,0)}$$

The problem raised with the congruence-like join is that if Ω is taken exactly as indicated in the definition, the resulting gcd will be a very large linear subgroup; the simple and effective solution consists in approximating Ω with a vector the projections of which on $S^p\mathbb{Q}^p$ have inverse integer coordinates with respect to S^p.

DEFINITION 103 (CHOICE $\downarrow^{\bowtie}$). Given two trapezoid congruences T_1 and T_2, the result $T_1 \downarrow^{\bowtie} T_2$ of the *choice* between T_1 and T_2 is the one having the smallest value by $\iota^{\bowtie} \circ \gamma^{\bowtie}$.

Approximate least upper bound

An approximation of the exact least upper bound operator is defined in terms of the homothetic join and of the congruence-like join.

DEFINITION 104 (APPROXIMATE JOIN $\sqcup^{\bowtie}$). Let $T_1 = [A_1, B_1] \langle S_1 \rangle_{(p_1,r_1,s_1,t_1)}$ and $T_2 = [A_2, B_2] \langle S_2 \rangle_{(p_2,r_2,s_2,t_2)}$ be to trapezoid congruences. Their *approximate join* $T_1 \sqcup^{\bowtie} T_2$ is equal to

$$\begin{cases} \quad T_1 & \text{if} \quad T_2 \subseteq T_1 \\ \text{else} \quad T_2 & \text{if} \quad T_1 \subseteq T_2 \\ \text{else} \quad (T_1' \sqcup_{\Diamond} T_2') \downarrow^{\bowtie} (T_1' \sqcup_{\nearrow} T_2') & \end{cases}$$

where $\langle S \rangle_{(p,r,s,t)} = \langle S_1 \rangle_{(p_1,r_1,s_1,t_1)} \bigwedge \langle S_2 \rangle_{(p_2,r_2,s_2,t_2)}$, $T_1' = \mathrm{Cast}_{\langle S \rangle_{(p,r,s,t)}}(T_1)$ and $T_2' = \mathrm{Cast}_{\langle S \rangle_{(p,r,s,t)}}(T_2)$.

2.3. Intersection. Since no shape meet algorithm is provided because only very approximate ones have been considered, only special cases of trapezoid congruences intersection are dealt with. Other cases are approximated either by one of their operands or by building an equational trapezoid congruence from the lists of RLICEs constituting both of the operand equational representations.

Homothetic meet on TC

The next definition provides an approximation of the intersection of two trapezoid congruences with the same shape.

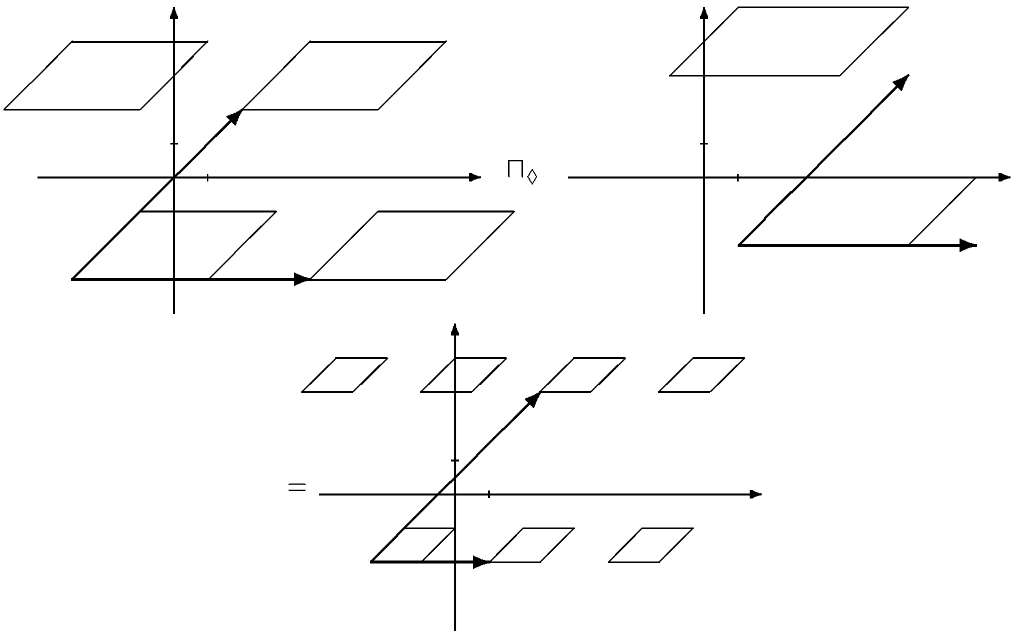


FIGURE VI.14. Homothetic meet $\sqcap_\Diamond$

DEFINITION 105 (HOMOTHETIC MEET $\sqcap_\Diamond$). Let $T_1 = (\Lambda_i.X \equiv [a_{1i}, b_{1i}]\,\langle q_i \rangle)_{i\in[1,n]}$ and $T_2 = (\Lambda_i.X \equiv [a_{2i}, b_{2i}]\,\langle q_i \rangle)_{i\in[1,n]}$ be two prime equational trapezoid congruences with the same associated parametric shape.

$$T_1 \sqcap_\Diamond T_2 \quad \stackrel{\text{def}}{=} \quad (\Lambda_i.X \equiv [a_{1i}, b_{1i}]\,\langle q_i \rangle \sqcap [a_{2i}, b_{2i}]\,\langle q_i \rangle)_{i\in[1,n]}$$

Moreover the cases where at least one of the intersections on interval congruences of the preceding definition provides the empty interval congruence derive from an empty exact homothetic meet of trapezoid congruences.

PROPOSITION 106 ($\sqcap_\Diamond$ SAFELY APPROXIMATES $\cap$). *Let T_1 and T_2 be two trapezoid congruences*

$$T_1 \cap T_2 \quad \subseteq \quad T_1 \sqcap_\Diamond T_2$$

The proof is exactly the same as for the proposition 100.

The example of the figure VI.14 corresponds to

$$\left[\begin{pmatrix} -3 \\ -3 \end{pmatrix}, \begin{pmatrix} 3 \\ -1 \end{pmatrix}\right] \left\langle \begin{matrix} 7 & 5 \\ 0 & 5 \end{matrix} \right\rangle_{(2,0,0,0)} \sqcap_{\Diamond} \left[\begin{pmatrix} 1 \\ -2 \end{pmatrix}, \begin{pmatrix} 8 \\ 0 \end{pmatrix}\right] \left\langle \begin{matrix} 7 & 5 \\ 0 & 5 \end{matrix} \right\rangle_{(2,0,0,0)}$$

which is equal to

$$\left[\begin{pmatrix} \frac{5}{2} \\ 3 \end{pmatrix}, \begin{pmatrix} 5 \\ 4 \end{pmatrix}\right] \left\langle \begin{matrix} \frac{7}{2} & 5 \\ 0 & 5 \end{matrix} \right\rangle_{(2,0,0,0)}$$

2.4. Widening. Two alternatives named congruence-like and interval-like widening are taken under consideration. They derive respectively of classical widenings on relational congruences and intervals. Let us explicate them separately on comparable trapezoid congruences $T_1 \subseteq T_2$ first before combining them in order to design a widening operator suitable for trapezoid congruences.

The transposition of Granger's widening on linear rational cosets to trapezoid congruences works as follows: take two comparable trapezoid congruences $T_1 = [A_1, B_1] \langle S_1 \rangle_{(p,r,0,0)}$ and $T_2 = [A_2, B_2] \langle S_2 \rangle_{(p,r,0,0)}$ having the same modulo linear part but possibly different modulo non-linear part, choose a direction vector E not generated by the linear common part of the modulo and find the smallest trapezoid congruence containing T_2 whose modulo linear part has been increased with E. Of course the choice of E is important and take T_1 into account in the sense that the density of points along the direction E must have increased between T_1 and T_2. In order to adapt this alternative to trapezoid congruences we can consider trapezoid congruences with identical modulos and simply take a vector of the modulo non-linear part and put it in the new modulo linear part. What is adopted is not so coarse but take the vectors of the modulo non-linear part along which the representative has increased from T_1 to T_2 and strictly increases the projection of the representative on them.

Now an adaptation of Cousot's widening on intervals is done by considering two comparable trapezoid congruences $T_1 = [A_1, B_1] \langle S \rangle_{(0,0,s,t)}$ and $T_2 = [A_2, B_2] \langle S \rangle_{(0,0,s,t)}$. The common vectors of the bounded part of the shape along which the representative has increased between T_1 and T_2 are placed in the unbounded part of the shape and the vectors of the unbounded part of the shape along which the representative has increased between T_1 and T_2 are placed in the linear part of the modulo.

Now we combine these two features simply using the widening operator on interval congruences.

Definition 107 (Equational widening ∇_1). Let $T_1 = (\Lambda_i.X \equiv [a_{1i}, b_{1i}] \langle q_i \rangle)_{i \in [1,n]}$ and $T_2 = (\Lambda_i.X \equiv [a_{2i}, b_{2i}] \langle q_i \rangle)_{i \in [1,n]}$ be two prime equational trapezoid congruences with the same associated parametric shape. Their *equational widening* $T_1 \nabla_1 T_2$ is defined by

$$T_1 \nabla_1 T_2 \stackrel{\text{def}}{=} (\Lambda_i.X \equiv ([a_{1i}, b_{1i}] \langle q_i \rangle \nabla [a_{2i}, b_{2i}] \langle q_i \rangle))_{i \in [1,n]}$$

where ∇ is the widening on interval congruences.

The result of the equational widening is an equational trapezoid congruence. Since the equational widening is parameterized by the widening on non relational interval congruence, other

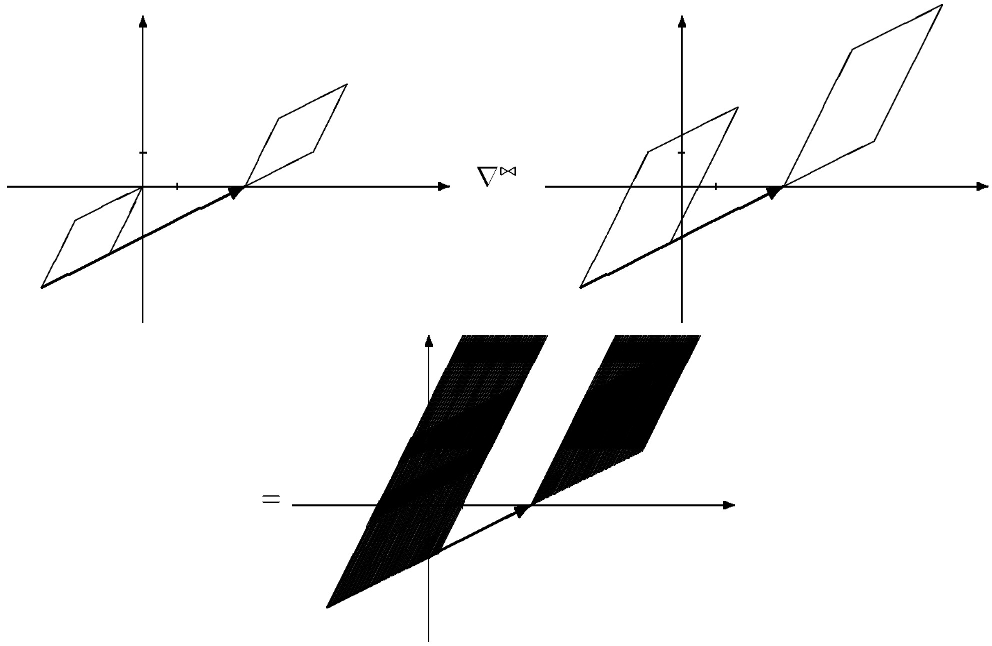


FIGURE VI.15. Relational widening $\nabla^{\bowtie}$.

operators are obtained by choosing any possible widening on IC, see the section 2.4 for such suggestions.

DEFINITION 108 (RELATIONAL WIDENING $\nabla^{\bowtie}$). Let $T_1 = [A_1, B_1] \langle S_1 \rangle_{(p_1,r_1,s_1,t_1)}$ and $T_2 = [A_2, B_2] \langle S_2 \rangle_{(p_2,r_2,s_2,t_2)}$ be two trapezoid congruences. Their *widening* $T_1 \nabla^{\bowtie} T_2$ is defined by

$$T_1 \nabla^{\bowtie} T_2 \quad \stackrel{\text{def}}{=} \quad \text{Cast}_{\langle S \rangle_{(p,r,s,t)}}(T_1) \nabla_1 \text{Cast}_{\langle S \rangle_{(p,r,s,t)}}(T_2)$$

where $\langle S \rangle_{(p,r,s,t)} = \langle S_1 \rangle_{(p_1,r_1,s_1,t_1)} \bigwedge \langle S_2 \rangle_{(p_2,r_2,s_2,t_2)}$.

The operator $\nabla^{\bowtie}$ is always defined. Indeed following lemma 87, two parametrical trapezoid congruences with the same shape are converted to equational forms of identical associated homogeneous systems. Those equational trapezoid congruences are then transformed into prime ones preserving the equality of their homogeneous part and the equational widening is possible.

The correctness of the relational widening operator definition results from the fact that

(1) $\nabla^{\bowtie}$ is greater than the join operator; it is a consequence of the extensivity of Cast and of ∇_1.
(2) the application of ∇_1 to a set of trapezoid congruences with the same shape is stationary after a finite number of steps (as a consequence of the similar property of ∇ on sets of interval congruences). The application of $\nabla^{\bowtie}$ to a set of trapezoid congruences

is equivalent to its application to an increasing chain because of the use of the Cast operator in $\nabla^{\bowtie}$. Finally the convergence property of ∇_1 leads to the convergence of $\nabla^{\bowtie}$.

An example of widening is provided on figure VI.15 where the widening

$$\left.\begin{array}{rcl} 2x-y & \equiv & [6,9]\,\langle 9\rangle \\ x-2y & \equiv & [0,3]\,\langle 0\rangle \end{array}\right\} \quad \nabla^{\bowtie} \quad \left\{\begin{array}{rcl} 2x-y & \equiv & [6,10]\,\langle 9\rangle \\ x-2y & \equiv & [-3,3]\,\langle 0\rangle \end{array}\right.$$

is in fact an equational widening and gives

$$\left\{\begin{array}{rcl} 2x-y & \equiv & [6,11]\,\langle 9\rangle \\ x-2y & \equiv & [-\infty,3]\,\langle 0\rangle \end{array}\right.$$

3. Abstract primitives

3.1. Affine assignment. An assignment of an affine expression to an integer variable is an affine transformation.

DEFINITION 109 (ABSTRACT AFFINE ASSIGNMENT Assign). Let F be an affine transformation on $\mathbb{Z}^n$ and u its linear part. The abstract application $\text{Assign}(F,T)$ of F to the trapezoid congruence $T = [A,B]\langle S\rangle_{(p,r,s,t)}$ is the trapezoid congruence defined by

$$\left[F(A) + \sum_{i=1}^{p+r+s+t} A_i, F(A) + \sum_{i=1}^{p+r+s+t} B_i\right] \langle S'\rangle_{(p',r',s',t')}$$

where

$$\langle S'\rangle_{(p',r',s',t')} = \bigwedge_{i=1}^{p+r+s+t} \langle u(S_i)\rangle_{(\epsilon_1,\epsilon_2,\epsilon_3,\epsilon_4)}$$

and

$$[A_i, B_i]\langle S'\rangle_{(p',r',s',t')} = \text{Cast}_{\langle S'\rangle_{(p',r',s',t')}}\Big([O, c_i u(S_i)]\langle u(S_i)\rangle_{(\epsilon_1,\epsilon_2,\epsilon_3,\epsilon_4)}\Big)$$

and $B - A = SC$; ϵ_1 is 1 if $1 \le i \le p$, 0 otherwise; ϵ_2 is 1 if $1 \le i-p \le r$, 0 otherwise; ϵ_3 is 1 if $1 \le i-p-r \le s$, 0 otherwise, ϵ_4 is 1 if $1 \le i-p-r-s \le t$, 0 otherwise.

This abstract affine assignment is not exact in general because the affine transformation of a trapezoid is not in general a trapezoid and hence has to be approximated with an embedding trapezoid. Intuitively, the abstract affine assignment proceeds as following: first an approximate shape $\langle S'\rangle_{(p',r',s',t')}$ of the result is determined and then the original trapezoid congruence is decomposed as the sum of trapezoid congruences with a one column shape $[O, c_i S_i]\langle S_i\rangle_{(\epsilon_1,\epsilon_2,\epsilon_3,\epsilon_4)}$ and abstract assigned separately giving $[O, c_i u(S_i)]\langle u(S_i)\rangle_{(\epsilon_1,\epsilon_2,\epsilon_3,\epsilon_4)}$; at the end the result is obtained by making the sum of all their conversions to the shape $\langle S'\rangle_{(p',r',s',t')}$.

PROOF. [of correctness] Recall that the abstract affine assignment is safe if

$$F(\gamma^{\bowtie}(T)) \quad \subseteq \quad \gamma^{\bowtie}(\text{Assign}(F,T))$$

Let us start by showing that $F(T) \subseteq \text{Assign}(F,T)$. The definition expression (49) of trapezoid congruences implies that there exists $O \le \Gamma \le C$ and $\Phi \in \mathbb{Z}^p\mathbb{Q}^r\{0\}^s\mathbb{Q}^t_+$ such that an element X of T is expressed

$$X = A + \sum_{i=1}^{p+r+s+t} (\gamma_i + \phi_i)S_i$$

an element X' of $F(T)$ is expressed

$$X' = F(A) + \sum_{i=1}^{p+r+s+t} (\gamma_i + \phi_i) u(S_i)$$

$F(A) \in [F(A), F(A)] \langle S' \rangle_{(p',r',s',t')}$ and for all $i \in [1, p+r+s+t]$ we have $(\gamma_i + \phi_i)u(S_i) \in [O, c_i u(S_i)] \langle u(S_i) \rangle_{(\epsilon_1,\epsilon_2,\epsilon_3,\epsilon_4)} \in [A_i, B_i] \langle S' \rangle_{(p',r',s',t')}$ and

$$\begin{aligned} X' &\in [F(A), F(A)] \langle S' \rangle_{(p',r',s',t')} + \sum_{i=1}^{p+r+s+t} [A_i, B_i] \langle S' \rangle_{(p',r',s',t')} \\ &\in \mathrm{Assign}(F, T) \end{aligned}$$

Now $F(T) \subseteq \mathrm{Assign}(F,T)$ and finally $F(\gamma^{\bowtie}(T)) = F(T \cap \mathbb{Z}^n) \subseteq F(T) \cap \mathbb{Z}^n \subseteq \mathrm{Assign}(F,T) \cap \mathbb{Z}^n = \gamma^{\bowtie}(\mathrm{Assign}(F,T))$. $\square$

Example
Suppose the assignment F

```
l:= 5*(i-1) + j
```

takes the entry context[2] T

$$\left[\begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 5 \\ 0 \\ 0 \end{pmatrix} \right] \left\langle \begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{matrix} \right\rangle_{(1,2,1,0)}$$

The exit context is given by the trapezoid congruence:

$$\left[\begin{pmatrix} 0 \\ 1 \\ 0 \\ -4 \end{pmatrix}, \begin{pmatrix} 0 \\ 5 \\ 0 \\ 0 \end{pmatrix} \right] \left\langle \begin{matrix} 1 & 0 & 0 \\ 0 & 0 & 4 \\ 0 & 1 & 0 \\ 5 & 0 & 4 \end{matrix} \right\rangle_{(1,1,1,0)}$$

3.2. Test with RLICE condition. The tests taken into account in our analysis correspond to conditions having an RLICE form. An advantage of the formalism of trapezoid congruences is that the negation of such conditions is straightforward (when the modulo of the RLICE is not null or at least one of its bounds is infinite).

DEFINITION 110 (ABSTRACT TEST Test). Let C be a RLICE, T an equational trapezoid congruence and E the set of equational trapezoid congruences consisting of RLICEs of T or of C such that their number is maximal. The *abstract test* $\mathrm{Test}(T, C)$ of condition C on context T is a minimal equational trapezoid congruence for the order $\iota^{\bowtie} \circ \gamma^{\bowtie}$ in E.

First remark that if the RLICE system obtained by adding the RLICE condition to the context is an equational trapezoid congruence then it is the result of the abstract test and

[2] The variables are in order i, j, k, l

in this case the abstract test is exact, hence optimal. The cases where the RLICE system, obtained by adding C to T, is singular are dealt with by removing one of the RLICE in order to get a trapezoid congruence.

The above definition only concerns the true branch of a test. The abstract test involved on the false branch is obtained by semantically negating the condition. The abstract test should have the condition $\alpha^{\bowtie}(N)$ with N the negation of the LCCE meaning of C. When the negation of $\gamma^{\bowtie}(C)$ is not a LCCE, but a conjunction of the LCCEs N_1 and N_2 (the case where the coset congruence of the LCCE is a finite integer interval), an approximation is obtained by taking the join of the abstract tests with $\alpha^{\bowtie}(N_1)$ and with $\alpha^{\bowtie}(N_1)$.

This operator is not comparable with the abstract test on rational relational cosets; it is in fact the only one not extending the corresponding operator on cosets and make the two analysis non comparable. This drawback is removable just by adding in the definition of the abstract test a special case corresponding to a rational linear congruence equation condition and a rational relational coset context, and considering their exact intersection.

Proof. [of correctness] Let us show that if S is the set of integer tuples solutions verifying the condition C we have

$$\gamma^{\bowtie}(T) \cap S \quad \subseteq \quad \gamma^{\bowtie}(\mathrm{Test}(T, C))$$

Every element of E by definition contains $T \cap C$, hence $T \cap C \subseteq \mathrm{Test}(T, C)$ and $(T \cap C) \cap \mathbb{Z}^n \subseteq \mathrm{Test}(T, C) \cap \mathbb{Z}^n$ and the result. $\square$

Example

Suppose we are analyzing the conditional :

```
if ((x + y) mod 100) = 2 then
    {S}
else
    {T}
```

with an entry context (before the if statement)

$$\left[\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \frac{3}{2} \\ 1 \\ 0 \end{pmatrix} \right] \left\langle \begin{matrix} 2 & 6 & 1 \\ 0 & -3 & 1 \\ -4 & 12 & 1 \end{matrix} \right\rangle_{(1,1,1,0)}$$

we get:

$$\left[\begin{pmatrix} 4 \\ -2 \\ 8 \end{pmatrix}, \begin{pmatrix} \frac{1}{2} \\ \frac{3}{2} \\ -2 \end{pmatrix} \right] \left\langle \begin{matrix} -2 & 200 & -3 \\ 2 & -100 & 3 \\ -12 & 400 & -7 \end{matrix} \right\rangle_{(2,0,1,0)}$$

At the entry of the "else" branch, adding to our original RLICEs system the complementary condition

$$x + y \equiv [-97, 1] \mod (100)$$

we get:

$$\left[\begin{pmatrix} -194 \\ 97 \\ -388 \end{pmatrix}, \begin{pmatrix} \frac{-3}{2} \\ \frac{5}{2} \\ -6 \end{pmatrix}\right] \left\langle \begin{matrix} -2 & 200 & -3 \\ 2 & -100 & 3 \\ -12 & 400 & -7 \end{matrix} \right\rangle_{(2,0,1,0)}$$

3.3. Projection. The abstract projection is useful to print the results of an analysis or to forget about some variables during an analysis, for example at the end of a procedure. The definition is very close to the one of abstract assignment, both being affine transformations.

Definition 111 (Abstract projection Proj). Let $T = [A, B] \langle S \rangle_{(p,r,s,t)}$ be a trapezoid congruence, V a set of variables of the program and A_V and S_V the projections of the lower bound and the matrix of the shape of T on V. The *abstract projection* $\mathrm{Proj}(T, V)$ of the context T on the variables of V is defined by

$$\left[A_V + \sum_{i=1}^{p+r+s+t} A_i, A_V + \sum_{i=1}^{p+r+s+t} B_i\right] \langle S' \rangle_{(p',r',s',t')}$$

where

$$\langle S' \rangle_{(p',r',s',t')} = \bigwedge_{i=1}^{p+r+s+t} \langle S_{V\,i} \rangle_{(\epsilon_1,\epsilon_2,\epsilon_3,\epsilon_4)}$$

and

$$[A_i, B_i] \langle S' \rangle_{(p',r',s',t')} = \mathrm{Cast}_{\langle S' \rangle_{(p',r',s',t')}} \Big([O, c_i S_{V\,i}] \langle S_{V\,i} \rangle_{(\epsilon_1,\epsilon_2,\epsilon_3,\epsilon_4)}\Big)$$

and $B - A = SC$, ϵ_1 is 1 if $1 \leq i \leq p$, 0 otherwise, ϵ_2 is 1 if $1 \leq i - p \leq r$, 0 otherwise, ϵ_3 is 1 if $1 \leq i - p - r \leq s$, 0 otherwise, ϵ_4 is 1 if $1 \leq i - p - r - s \leq t$, 0 otherwise.

The proof of the correctness of the abstract projection is quite close to the one of the abstract affine assignment. It is in fact the justification of the expression of the abstract projection.

Example
The exact projection of the trapezoid congruence

$$\begin{array}{rcl} i & \equiv & [0, 0] \langle 1 \rangle \\ -5i + k & \equiv & [1, 2] \langle 0 \rangle \\ j & \equiv & [1, 5] \langle 0 \rangle \\ -5i - j + l & \equiv & [-5, -5] \langle 0 \rangle \end{array}$$

on the subspace corresponding to the last two variables (the subset $V = \{k, l\}$) is

$$\begin{array}{rcl} k & \equiv & [1, 2] \langle 5 \rangle \\ -k + l & \equiv & [-6, -1] \langle 0 \rangle \end{array}$$

The opposite operator, that is embedding a trapezoid congruence of $\mathbb{Q}^n$ in $\mathbb{Q}^m$ with $m > n$, is even simpler than the projection because the embedding of the linearly independent vectors of the original trapezoid congruence shape are linearly independent.

3.4. Example. A prototype (about 6000 lines of Standard ML including the underlying coset lattices operators) is implemented according to the previous definitions in order to solve the approximate semantic equations automatically following the abstract interpretation framework of [CC77].

The example of Figure VI.16 corresponds to a backsubstitution on a matrix structured as a set of linear finite difference equations with boundary conditions imposed at endpoints. It solves `sx = b` for triangular matrices of shape:

$$\begin{pmatrix}
1 & & x & x & & & & & & & & & & & \\
 & 1 & x & x & & & & & & & & & & & \\
 & & 1 & & & & x & x & & & & & & & \\
 & & & 1 & & & x & x & & & & & & & \\
 & & & & 1 & & x & x & & & & & & & \\
 & & & & & 1 & x & x & & & & & & & \\
 & & & & & & 1 & & & & x & x & & & \\
 & & & & & & & 1 & & & x & x & & & \\
 & & & & & & & & 1 & & x & x & & & \\
 & & & & & & & & & 1 & x & x & & & \\
 & & & & & & & & & & \cdot & & & & \\
 & & & & & & & & & & & \cdot & & & \\
 & & & & & & & & & & & & \cdot & & \\
 & & & & & & & & & & & & & 1 & \\
 & & & & & & & & & & & & & & 1
\end{pmatrix}$$

where x represents possibly non null elements and empty spaces stand for zeros. The use of while loops instead of for loops is only for sake of clarity in the process of determining the set of semantic equations.

If α : denotes the program point just preceding the statement of the corresponding line on Figure VI.16 and T_α the system of RLICEs verified by the integer program variables at point α :, the system of approximate semantic equations associated with the backsubstitution procedure is the following:

$$\begin{array}{lrcl}
(57) & T_2 & = & \text{Test}(T_1, (i \equiv 0 \mod (1)))\nabla T_{11} \\
(58) & T_3 & = & \text{Assign}(T_2, j \leftarrow ne) \\
(59) & T_4 & = & \text{Test}\,((T_3 \nabla T_{10}), (j \equiv [1, ne] \mod (0))) \\
(60) & T_5 & = & \text{Assign}(T_4, l \leftarrow ne * i + j - ne) \\
(61) & T_6 & = & \text{Assign}(T_5, k \leftarrow ne * i + 1) \\
(62) & T_7 & = & \text{Test}\,((T_6 \nabla T_8), (k - ne * i \equiv [1, nb] \mod (0))) \\
(63) & T_8 & = & \text{Assign}(T_7, k \leftarrow k + 1) \\
(64) & T_9 & = & T_6 \nabla T_8 \\
(65) & T_{10} & = & \text{Assign}(T_9, j \leftarrow j - 1) \\
(66) & T_{11} & = & T_3 \nabla T_{10}
\end{array}$$

`ne` and `nb` are supposed to be some constants[3] declared in the calling procedures. The resolution process converges in two iterations and gives the following interesting result that

[3]In the following, $ne = 5$ and $nb = 2$; ne and nb respectively correspond to height and width of the rectangles of possibly non null coefficients of the matrix being backsubstituted.

```
PROCEDURE bksub(ne,nb,n:INTEGER;VAR x:glxarray;
                s:glsarray;b:glbarray);
VAR
  i,j,l,k:INTEGER;
BEGIN
    ...
1:  FOR i:=n-nb DOWNTO ne-nb+1 DO BEGIN
2:    j:=ne;
3:    WHILE (1<=j) AND (j<=ne) DO BEGIN
4:      l:=ne*(i-1) +j;
5:      x[l]:=b[l];
        k:=ne*i +1;
6:      WHILE ((ne*i +1)<=k)
              AND (k<=(ne*i +nb)) DO BEGIN
7:        x[l]:=x[l]-x[k]*s[l,k];
          k:=k+1
8:      END
9:      j:=j-1
10:   END;
11: END
    ...
END;
```

FIGURE VI.16. Backsubstitution following a Gaussian elimination.

---

at point 7: the accessors of the array s verify:

$$\begin{aligned} k &\equiv 1.\,[1,2]\,\langle 5\rangle \\ -k+l &\equiv 1.\,[-6,-1]\,\langle 0\rangle \end{aligned}$$

That is a good approximation of the effectively used part of matrix s. The origin of the inexactitude is that the representatives of trapezoid congruences are supposed to be parallel to the directions of the modulo and that it is not the case here since representatives are rectangles and the modulo is along the first bisector.

The compile time detection of such properties allows to use naive algorithms like the one given in Figure VI.16 without worrying about optimal storage problems on sparse matrices.

CHAPTER VII

APPLICATIONS

1. Representation of integer arrays

The following procedure is used in the process of data encryption coming from [PFTV86]. Although it is not the optimal coding method, it very well illustrates the possible use of the trapezoid congruence analysis for the purpose of representing integer arrays.

```
procedure ks(key: gl64array; n: integer; var kn: gl48array);
var
   j,it,id,ic,i: integer;
begin
{1:}    if n = 1 then begin
{2:}       for j := 1 to 56 do begin
{3:}          glicd[j] := key[ipc1[j]] end end;
{4:}    it := 2;
{5:}    if (n=1) or (n=2) or (n=9) or (n=16) then it := 1
{6:}    for i := 1 to it do begin
{7:}       ic := glicd[1]; id := glicd[29];
{8:}       for j := 1 to 27 do begin
{9:}          glicd[j] := glicd[j+1]; glicd[j+28] := glicd[j+29] end;
{10:}      glicd[28] := ic; glicd[56] := id end;
{11:}   for j := 1 to 48 do
{12:}      kn[j] := glicd[ipc2[j]]
end;
```

where `glicd` and `ipc1` are global arrays of 56 integers and `ipc2` a global array of 48 integers. This procedure is called several times to make 16 sub-keys from the initial one `key`. Remark that the abstract version of the conditional expression of the line `5:` is $n \equiv [1,2]\langle\frac{15}{2}\rangle$. The integer arrays `ipc1` and `ipc2` are constants of the program. The relation between their indices and their values can be abstracted by a trapezoid congruence. Let us call index_1 the abstract index of the constant array `ipc1` and ipc1 the corresponding value. In our example, `ipc1` is instanciated to:

```
60 52 44 36 28 20 12 4 59 51 43 35 27 19 11 3 58 50 42 34 26 18 10 2
57 49 41 33 25 17 9 1 64 56 48 40 32 24 16 8 63 55 47 39 31 23 15 7
62 54 46 38 30 22 14 6
```

and is safely represented by the trapezoid congruence:

$$\begin{aligned} \text{ipc1} &\equiv [1,64]\langle 0\rangle \\ 8 * \text{index}_1 + \text{ipc1} &\equiv [-3,5]\langle 63\rangle \end{aligned}$$

Since it is very simple to determine that for two different values of the index index_1 (element of $[1,56]$) the corresponding values of ipc1 in the trapezoid congruence abstraction of `ipc1` are different, we know that all the references to the elements of the array `key` at program point `3:` are distinct. Such a conclusion allows the loop parallelization when the mentioned references are on the left hand side of the assignment. The abstract relation between the value ipc1 and its index index_1 is a safe relation between the index of an element of the array `key` and its position in the array `glicd`: if the element e of index ind in the array `key` has been assigned to array `glicd` with index ind' then the relation

$$\begin{aligned} \text{ind'} &\equiv [1,64]\langle 0\rangle \\ 8 * \text{ind} + \text{ind'} &\equiv [-3,5]\langle 63\rangle \end{aligned}$$

holds.

If we rewrite the loop of program point `8:`

```
{1: } for j := 1 to 27 do begin
{2: }     k := j; l := j+1;
{3: }     glicd[k] := glicd[l];
{4: }     k := j+28; l := j+29;
{5: }     glicd[k] := glicd[l] end;
```

the trapezoid congruence analysis determines the projection of the approximation of the invariant on the variables `k` and `l`

$$\begin{aligned} l - k &\equiv [1,1]\langle 0\rangle \\ k &\equiv [1,27]\langle 0\rangle \end{aligned}$$

at program point `3:` and

$$\begin{aligned} l - k &\equiv [1,1]\langle 0\rangle \\ k &\equiv [29,55]\langle 0\rangle \end{aligned}$$

at program point `5:`. Hence making the join

$$\begin{aligned} l - k &\equiv [1,1]\langle 0\rangle \\ k &\equiv [1,27]\langle 28\rangle \end{aligned}$$

of these two invariant provides an approximation between the index `k` of an element of the array `glicd` after the loop and its position `l` in the array `glicd` before the execution of the loop. The combination of such information provides safe relations between the index of the input array `key` and the output array `kn`.

1.1. Related work. Several methods exist for summarizing array accesses, including those based on simple sections [BK89] that are a special kind of trapezoid where the linear coefficients figuring in the equational representation are in $\{-1, 0, 1\}$, or on regular sections [HK91] that corresponds to the combination of the two existing non relational analyses of intervals [CC76] and cosets [Gra89], even on convex hulls based on [CH78] that figures in [Tri85]. [May92] proposes an approach quite similar to a simple classical non relational interval analysis.

2. Dependence analysis

The use of a relational integer abstract interpretation for solving data dependence problems is described in [Mas91]. The use of the trapezoid congruence analysis in this framework is of course very interesting because of the use that it makes of the models of multidimensional rectangles and linear cosets. The very original contribution of the trapezoid congruence analysis for testing data dependence comes from its possibility to give a very accurate representation of indirection arrays. For example, very frequently, indirection arrays implement permutations that are represented using a trapezoid congruence (like in the preceding section), and if the trapezoid congruence approximation is accurate enough, a loop such as

```
for i := 1 to n do
   A[a[i]] := B[i]
```

is possibly parallelized by taking into account the permutation feature of the indirection array `a`.

On the next program example

```
      for i := 3 to 100 do begin
{S}      A[2*i] := B[i]+2
         if even(i) then
{T}         C[i] := D[i]+A[2*i+1]+A[2*i-4]+A[i] end
```

the non relational analysis detects that the variables `A[2*i]` of statement {`S`} and `A[2*i+1]` of statement {`T`} are independent. The relational analysis determines that the variables `A[2*i]` of {`S`} and `A[2*i-4]` of {`T`} are dependent and the corresponding distance vector is (2). It determines also that the variables `A[2*i]` of {`S`} and `A[i]` of {`T`} are dependent and the corresponding distance vector is (i).

The analysis of the program

```
      for i := 0 to 20 do
         for j := 0 to 20 do begin
            for k := 0 to 20 do begin
               F1 := j-1; F2 := 3*i+2; F3 := 3*k-7;
{S}            A[F1,F2,F3] := C[i+j*k] end
            for k := 0 to 20 do begin
               G1 := 3*i+4; G2 := 5*j-2; G3 := -2*k+4;
{T}            B[i*j*k] := A[G1,G2,G3] end end
```

using the linear congruence analysis of [Gra91a] implemented in our prototype determines that the statement {`T`} may depend on {`S`} for the elements `A[A1,A2,A3]` of the array `A` characterized by the relation

$$\begin{array}{rcl} A1 & \equiv & 10 \mod (15) \\ A3 & \equiv & 2 \mod (6) \\ -A1 + A2 & = & -2 \end{array}$$

3. Other derived analyses

Because the array indexes are essentially in a constant integer interval (multidimensionnal integer rectangle for multidimensionnal arrays) the combination of the trapezoid congruence analysis with a classical interval analysis should improve the accuracy of the results. Indeed, the choice made in the join operator between the different join strategies is more precise if the information resulting from an interval analysis is taken into account. For example choosing between $[3,5]\langle 7\rangle$ and $[4,6]\langle 7\rangle$ under the constraint $[0,5]$ should lead to $[4,6]\langle 7\rangle$ since $[4,6]\langle 7\rangle \cap [0,5] \subset [3,5]\langle 7\rangle \cap [0,5]$.

Several analyses are easily derived from the trapezoid congruence analysis, either in a non relationnal way or in a relationnal way. It is the case when the modulo of the trapezoid congruence is fixed during the analysis (its value depends on syntactic features of the program for example), a special case of which considers always null modulo elements (they are in fact a special case of linear inequalities).

Another special kind of trapezoid congruences is possibly considered where all the linear coefficients of the equational representation are in $\{-1, 0, 1\}$.

CONCLUSION

The presented work, in addition of completing the existing analyses on integer numbers, provides a method for combining two analyses. First, two well known abstract domains are considered and a more general than these two basic is built. Instead of the usual combination of the two basic analyses, which runs in parallel the two analyses and makes them interact at every step of the analysis, our combination runs only one analysis that heuristically determines at each step which one of the two basic analyses is the most informative. This is enabled by the generallity of our model.

A very interesting future work using the trapezoid congruence analysis is to design an abstract domain dealing with integer arrays by representing them by trapezoid congruence relations, that was our initial goal. It has been shown in this work that for example integer arrays implementing permutations are very well abstracted by trapezoid congruences, even when they are not abstracted by linear constraints or by linear congruence equations.

On an other hand, our analysis is extensible to an analysis of rational variables, by simply suppressing a number of links between the two abstraction levels, hence giving very close algorithms. This new analysis is then used to represent general arrays of rational numbers.

List of Definitions

List of Theorems

List of Figures

Contents

Bibliography

[AHI90] Z. Ammarguellat and W.L. Harrison III. Automatic recognition of induction variables and recurrence relations by abstract interpretation. In *Conference on Programming Language Design and Implementation*, pages 283–295, 1990.

[AK84] J.R. Allen and K. Kennedy. Automatic loop interchange. In *ACM Symposium on Compiler Construction*, volume 19 of *SIGPLAN notices*, pages 233–246, June 1984.

[AK87] R. Allen and K. Kennedy. Automatic translation of FORTRAN programs to vector form. *ACM Trans. Programming Languages and Systems*, 9(4):491–542, 1987.

[AN88] A. Aiken and A. Nicolau. Optimal loop parallelization. In *Conference on Programming Language Design and Implementation*, pages 308–317, 1988.

[Bir67] G. Birkhoff. *Lattice Theory*. Am. Math. Soc. Colloquium Publication, 1967.

[BK89] V. Balasundaram and K. Kennedy. A technique for summarizing data access and its use in parallelism enhancing transformations. In *Conference on Programming Language Design and Implementation*, SIGPLAN Notices, pages 41–53, Jun. 1989.

[BK93] S.B. Baden and S.R. Kohn. Lattice parallelism: A parallel programming model for manipulating non-uniform structured scientific data structures. In *Workshop on Languages, Compilers, and Run-Time Environments for Distributed Memory Multiprocessors*, volume 28 of *ACM SIGPLAN Notices*, pages 24–27, Boulder, Colorado, January 1993. ACM Press.

[CC76] P. Cousot and R. Cousot. Static determination of dynamic properties of programs. In Paris Dunod, editor, *Proc. of the second International Symposium on programming*, pages 106–130, 1976.

[CC77] P. Cousot and R. Cousot. Abstract interpretation : a unified lattice model for static analysis of programs by construction of approximation of fixpoints. In *4th Annual ACM Symposium on Principles of Programming Languages*, pages 238–252, Los Angeles, January 1977.

[CC79] P. Cousot and R. Cousot. Systematic design of program analysis frameworks. In *6th Annual ACM Symposium on Principles of Programming Languages*, pages 269–282, 1979.

[CC92a] P. Cousot and R. Cousot. Abstract interpretation and application to logic programs. *Journal of Logic Programming*, 13(2–3), 1992.

[CC92b] P. Cousot and R. Cousot. Abstract interpretation frameworks. *Journal of Logic and Computation*, 1992.

[CCK90] D. Callahan, S. Carr, and K. Kennedy. Improving register allocation for subscripted variables. In *Conference on Programming Language Design and Implementation*, pages 53–65, June 1990.

[CH78] P. Cousot and N. Halbwachs. Automatic discovery of linear restraints among variables of a program. In *Fifth Annual ACM Symposium on Principles of Programming Languages*, pages 84–97, Tucson, Ariz., Jan. 1978.

[Cou81] P. Cousot. *Program Flow Analysis: Theory and Applications*, chapter Semantic foundations of program analysis, pages 303–342. Prentice-Hall, 1981.

[Cox88] M.G. Cox. *Scientific Software Systems*, chapter Linear Algebra Support Modules for Approximation and other Software, pages 25–46. Chapman and Hall, 1988.

[Deu92] A. Deutsch. A storeless model of aliasing and its abstraction using finite representation of right-regular equivalence relations. In *International Conference on Computing Languages*, 1992.

[D'H89] E.H. D'Hollander. Partitionning and labeling of index sets in do loops with constant dependence vectors. In *International Conference on Parallel Processing*, pages 139–144, 1989.

[Dow90] M.L. Dowling. Optimal code parallelization using unimodular transformations. *Parallel computing*, 16:157–171, 1990.

[EHLP91] R. Eigenmann, J. Hoeflinger, Z. Li, and D. Padua. Experience in the automatic parallelization of four perfect-benchmark programs. In *Proceedings of the Fourth International Workshop on Languages and Compilers for Parallel Computing*, volume 589 of *Lecture Notes on Computer Science*, pages 65–82, 1991.

[Fea88a] P. Feautrier. Array expansion. In *ACM International Conference on Supercomputing*, pages 429–441, Saint-Malo, 1988.

[Fea88b] P. Feautrier. Semantical analysis and mathematical programming; application to parallelization and vectorization. In M. Cosnard et al., editor, *Workshop on Parallel and distributed algorithms*, pages 309–320. Elsevier Science Publisher B.V. (North Holland), 1989, October 1988.

[FW91] P. Feautrier and M.R. Werth. Systematic construction of programs for distributed memory computers. Technical Report MASI 91.36, Laboratoire de Méthodologie et Architecture des Systèmes Informatiques, 4, Place Jussieu, 75252 Paris cedex 05, 1991.

[Ger89] M. Gerndt. Array distribution in SUPERB. In *third International Conference on Supercomputing*, pages 164–174, 1989.

[GJG87] D. Gannon, W. Jalby, and K. Gallivan. Strategies for cache and local memory management by global program transformation. In *International Conference on Supercomputing*, pages 229–254, 1987.

[Gra89] P. Granger. Static analysis of arithmetical congruences. *Intern. J. Computer Math.*, 30:165–190, 1989.

[Gra90] P. Granger. Static analysis of linear congruence equalities among variables of a program. Research Report LIX/RR/90/10, Ecole Polytechnique, 91128 Palaiseau, France, Nov. 1990.

[Gra91a] P. Granger. *Analyses sémantiques de congruence*. PhD thesis, École Polytechnique, Palaiseau, July 1991.

[Gra91b] P. Granger. Static analysis of linear congruence equalities among variables of a program. In *International Joint Conference on Theory and Practice of Software Development*, volume 493 of *Lecture Notes on Computer Science*, pages 169–192. Springer Verlag, 1991.

[GS90] T. Gross and P. Steenkiste. Structured dataflow analysis for arrays and its use in optimizing compiler. *Software — Practice and Experience*, 20(2):133–155, February 1990.

[Gup90] R. Gupta. A fresh look at optimizing array bound checking. In *Conference on Programming Language Design and Implementation*, pages 272–282, 1990.

[HK91] P. Havlak and K. Kennedy. An implementation of interprocedural bounded regular section analysis. *IEEE Trans. on Parallel and Distributed Systems*, 2(3):350–360, 1991.

[HKT92] S. Hiranandani, K. Kennedy, and C.-W. Tseng. Compiler optimizations for Fortran D on MIMD distributed-memory machines. 1992.

[JD89] P. Jouvelot and B. Dehbonei. A unified semantic approach for the vectorization and parallelization of generalized reductions. In *ACM International Conference on Supercomputing*, pages 186–194, 1989.

[Jou87] P. Jouvelot. Semantic parallelization: A practical exercise in abstract interpretation. In *Annual ACM Symposium on Principles of Programming Languages*, pages 39–48, Jan. 1987.

[Kar76] M. Karr. Affine relationships among variables of a program. *Acta Informatica*, 6:133–151, 1976.

[Kil73] G.A. Kildall. A unified approach to global program optimization. In *Annual ACM Symposium on Principles of Programming Languages*, pages 194–206, 1973.

[KLS90] K. Knobe, J.D. Lukas, and Jr. Steele, G.L. Data optimization : Allocation of arrays to reduce communication on SIMD machines. *Journal of Parallel and Distributed Computing*, 8:102–118, 1990.

[LKK85] G. Lee, C.P. Kruskal, and D.J. Kuck. An empirical study of automatic restructuring of nonnumerical programs for parallel processors. *IEEE Transactions on Computers*, C-34(10):927–933, Oct. 1985.

[Mas91] F. Masdupuy. Using abstract interpretation to detect array data dependencies. In *International Symposium on Supercomputing, Fukuoka*, pages 19–27. Kyushu University press, 1991.

[Mas92] F. Masdupuy. Array operations abstraction using semantic analysis of trapezoid congruences. In *International Conference on Supercomputing*, Washigton D.C., July 1992.

[Mas93] F. Masdupuy. Semantic analysis of interval congruences. In *Proc. of the International Conference on Formal Methods in Programming and their Applications*, volume 735 of *LNCS*, pages 142–155, July 1993.

[May92] D.E. Maydan. *Accurate Analysis of Array References*. PhD thesis, Stanford University, September 1992.

[MHL91] D.E. Maydan, J.L. Hennessy, and M.S. Lam. Efficient and exact data dependence analysis. In *Conference on Programming Language Design and Implementation*, pages 1–14, June 1991.

[Mon92] B. Monsuez. Polymorphic typing by abstract interpretation. In *Foundations of Software Technology and Theoretical Computer Science*, pages 217–228, dec. 1992.

[MR88] T.J. Marlowe and B.G. Ryder. Properties of data flow frameworks: A unified model. Technical Report LCSR-TR-103, Rutgers University, Department of Computer Science, Hill Center for the Mathematical Sciences Bush Campus, Rutgers University, New Brunswick, New Jersey 08903, Apr. 1988.

[O.44] Ore O. Galois connections. *Transactions Amer. Math. Soc.*, 55:493–515, 1944.

[PFTV86] H.P. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, 1986.

[PP91] S.S. Pinter and R.Y. Pinter. Program optimization and parallelization using idioms. In

Annual ACM Symposium on Principles of Programming Languages, pages 79–92, Jan. 1991.

[SLY89] Z. Shen, Z. Li, and P. Yew. An empirical study on array subscript and data dependencies. In *International Conference on Parallel Processing*, pages 145–152, 1989.

[TP93] P. Tu and D. Padua. Array privatization for shared and ditributed memory machines. In *Workshop on Languages, Compilers, and Run-Time Environments for Distributed Memory Multiprocessors*, volume 28 of *ACM SIGPLAN Notices*, pages 64–67, Boulder, Colorado, January 1993. ACM Press.

[Tri85] R. Triolet. Interprocedural analysis for program restructuring with PARAFRASE. Technical Report 538, University of Illinois Urbana-Champaign, Dec. 1985.

[Wal88] D.R. Wallace. Dependence of multi-dimensional array references. In *International Conference on Supercomputing*, pages 418–428, 1988.

[WL91a] M.E. Wolf and M.S. Lam. A data locality optimizing algorithm. In *Conference on Programming Language Design and Implementation*, pages 30–44, Toronto, Ontario, Canada, June 1991.

[WL91b] M.E. Wolf and M.S. Lam. A loop transformation theory, and an algorithm to maximize parallelism. *IEEE Transactions on parallel and distributed systems*, 2(4):165–190, Oct. 1991.