An Improved Tight Closure Algorithm for Integer Octagonal Constraints*

Roberto Bagnara[1], Patricia M. Hill[2], and Enea Zaffanella[1]

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

Abstract. Integer octagonal constraints (a.k.a. *Unit Two Variables Per Inequality* or *UTVPI integer constraints*) constitute an interesting class of constraints for the representation and solution of integer problems in the fields of constraint programming and formal analysis and verification of software and hardware systems, since they couple algorithms having polynomial complexity with a relatively good expressive power. The main algorithms required for the manipulation of such constraints are the satisfiability check and the computation of the inferential closure of a set of constraints. The latter is called *tight* closure to mark the difference with the (incomplete) closure algorithm that does not exploit the integrality of the variables. In this paper we present and fully justify an $\mathrm{O}(n^3)$ algorithm to compute the tight closure of a set of UTVPI integer constraints.

1 Introduction

Integer octagonal constraints, also called *Unit Two Variables Per Inequality (UTVPI) integer constraints* —that is, constraints of the form $ax+by \leq d$ where $a, b \in \{-1, 0, +1\}$, $d \in \mathbb{Z}$ and the variables x and y range over the integers—, constitute an interesting subclass of linear integer constraints admitting polynomial solvability. The place these constraints occupy in the complexity/expressivity spectrum is in fact peculiar. Concerning complexity, relaxing the restriction imposing (at most) two variables per constraint, or relaxing the restriction on coefficients, or relaxing both restrictions make the satisfiability problem NP-complete [13, 14]. Concerning expressivity, integer octagonal constraints can be used for representing and solving many integer problems in the field of constraint programming, such as temporal reasoning and scheduling [13]. In the field of formal analysis and verification of software and hardware systems, these constraints have been successfully used in a number of applications [5, 6, 9, 19].

* This work has been partly supported by MURST project "AIDA — Abstract Interpretation: Design and Applications," and by a Royal Society (UK) International Joint Project (ESEP) award.

When (integer or rational) octagonal constraints are used to build abstract domains[3] —such as the *Octagon Abstract Domain* implemented in the library with the same name [20] or the domain of *octagonal shapes* defined in [2] and implemented in the *Parma Polyhedra Library* [4]— the most critical operation is not the satisfiability check (although very important in constraint programming) but *closure by entailment*. This is the procedure whereby a set of octagonal constraints is augmented with (a finite representation of) all the octagonal constraints that can be inferred from it. The closure algorithms for rational octagonal constraints are sound but not complete for integer octagonal constraints. The latter require so-called *tight* closure algorithms that fully exploit the integrality of the variables.

In 2005, Lahiri and Musuvathi proposed an $\mathrm{O}(n^3)$ algorithm for the satisfiability check of a (non trivially redundant) system of UTVPI integer constraints [15]. They also sketched (without formal definitions and proofs) a tight closure algorithm with the same worst-case complexity bound. Still in 2005, Miné proposed a modification of the strong (i.e., non-tight) closure algorithm for *rational* octagonal constraints and argued that this would provide a good and efficient approximation of tight closure [16]. In the same year we showed that the algorithm for computing the strong closure of rational octagonal constraints as described in [16] could be simplified with a consequential improvement in its efficiency [2, 3]. In this paper we show that our result can be extended so as to apply to integer octagonal constraints. This enables us to present and, for the first time, fully justify an $\mathrm{O}(n^3)$ algorithm to compute the tight closure of a set of UTVPI integer constraints.

In Section 2 we briefly introduce the terminology and notation adopted throughout the paper and we recall a few standard results on weighted graphs. In Section 3, we give the definition of rational-weighted octagonal graphs and recall some of the results that were established in [2, 3]. In Section 4, we extend these results to the case of integer-weighted octagonal graphs. Finally, in Section 5 we conclude and briefly discuss future work.

2 Preliminaries

Let $\mathbb{Q}_\infty := \mathbb{Q} \cup \{+\infty\}$ be totally ordered by the extension of '$<$' such that $d < +\infty$ for each $d \in \mathbb{Q}$. Let $\mathcal{N}$ be a finite set of *nodes*. A *rational-weighted directed graph* (graph, for short) G in $\mathcal{N}$ is a pair $(\mathcal{N}, w)$, where $w \colon \mathcal{N} \times \mathcal{N} \to \mathbb{Q}_\infty$ is the weight function for G.

Let $G = (\mathcal{N}, w)$ be a graph. A pair $(n_i, n_j) \in \mathcal{N} \times \mathcal{N}$ is an *arc* of G if $w(n_i, n_j) < +\infty$; the arc is *proper* if $n_i \neq n_j$. A *path* $\pi = n_0 \cdots n_p$ in G is a non-empty and finite sequence of nodes such that (n_{i-1}, n_i) is an arc of G, for all $i = 1, \ldots, p$. Each node n_i where $i = 0, \ldots, p$ and each arc (n_{i-1}, n_i) where

[3] In *abstract interpretation* theory [8], an *abstract domain* is an algebraic structure formalizing a set of approximate assertions endowed with an entailment (or approximation) relation, plus various operations that correctly approximate the operations of some *concrete domain*, i.e., the domain being abstracted/approximated.

$i = 1, \ldots, p$ is said to be *in* the path π. The *length* of the path π is the number p of occurrences of arcs in π and denoted by $\|\pi\|$; the *weight* of the path π is $\sum_{i=1}^{p} w(n_{i-1}, n_i)$ and denoted by $w(\pi)$. The path π is *simple* if each node occurs at most once in π. The path π is *proper* if all the arcs in it are proper. The path π is a *proper cycle* if it is a proper path, $n_0 = n_p$ and $p \geq 2$. If $\pi_1 = n_0 \cdots n_h$ and $\pi_2 = n_h \cdots n_p$ are paths, where $0 \leq h \leq p$, then the path concatenation $\pi = n_0 \cdots n_h \cdots n_p$ of π_1 and π_2 is denoted by $\pi_1 :: \pi_2$; if $\pi_1 = n_0 n_1$ (so that $h = 1$), then $\pi_1 :: \pi_2$ will also be denoted by $n_0 \cdot \pi_2$. Note that path concatenation is not the same as sequence concatenation. The path π is a *zero-cycle* if it is a proper cycle and $w(\pi) = 0$. A graph is *zero-cycle free* if all its proper cycles have strictly positive weights.

A graph $(\mathcal{N}, w)$ can be interpreted to represent the system of *potential constraints*

$$\mathcal{C} := \left\{ n_i - n_j \leq w(n_i, n_j) \;\middle|\; n_i, n_j \in \mathcal{N} \right\}.$$

Hence, the graph $(\mathcal{N}, w)$ is *consistent* if and only if the system of constraints it represents is satisfiable in $\mathbb{Q}$, i.e., there exists a rational valuation $\rho \colon \mathcal{N} \to \mathbb{Q}$ such that, for each constraint $(n_i - n_j \leq d) \in \mathcal{C}$, the relation $\rho(n_i) - \rho(n_j) \leq d$ holds. It is well-known that a graph is consistent if and only if it has no negative weight cycles (see [7, Section 25.5] and [23]).

The set of consistent graphs in $\mathcal{N}$ is denoted by $\mathbb{G}$. This set is partially ordered by the relation '$\trianglelefteq$' defined, for all $G_1 = (\mathcal{N}, w_1)$ and $G_2 = (\mathcal{N}, w_2)$, by

$$G_1 \trianglelefteq G_2 \quad \iff \quad \forall i, j \in \mathcal{N} : w_1(i, j) \leq w_2(i, j).$$

We write $G \triangleleft G'$ when $G \trianglelefteq G'$ and $G \neq G'$. When augmented with a bottom element $\bot$ representing inconsistency, this partially ordered set becomes a non-complete lattice $\mathbb{G}_\bot = \langle \mathbb{G} \cup \{\bot\}, \trianglelefteq, \sqcap, \sqcup \rangle$, where '$\sqcap$' and '$\sqcup$' denote the finitary greatest lower bound and least upper bound operators, respectively.

Definition 1. (Closed graph.) *A consistent graph $G = (\mathcal{N}, w)$ is* closed *if the following properties hold:*

$$\forall i \in \mathcal{N} : w(i, i) = 0; \tag{1}$$

$$\forall i, j, k \in \mathcal{N} : w(i, j) \leq w(i, k) + w(k, j). \tag{2}$$

The (shortest-path) closure *of a consistent graph G in $\mathcal{N}$ is*

$$\mathrm{closure}(G) := \bigsqcup \left\{ G' \in \mathbb{G} \;\middle|\; G' \trianglelefteq G \textit{ and } G' \textit{ is closed} \right\}.$$

When trivially extended so as to behave as the identity function on the bottom element $\bot$, shortest-path closure is a kernel operator (monotonic, idempotent and reductive) on the lattice $\mathbb{G}_\bot$, therefore providing a canonical form.

The following lemma recalls a well-known result for closed graphs (for a proof, see Lemma 5 in [3]).

Lemma 1. *Let $G = (\mathcal{N}, w) \in \mathbb{G}$ be a closed graph. Then, for any path $\pi = i \cdots j$ in G, it holds that $w(i, j) \leq w(\pi)$.*

3 Rational Octagonal Graphs

We assume in the following that there is a fixed set $\mathcal{V} = \{v_0, \dots, v_{n-1}\}$ of n variables. The octagon abstract domain allows for the manipulation of *octagonal constraints* of the form $av_i + bv_j \leq d$, where $a, b \in \{-1, 0, +1\}$, $a \neq 0$, $v_i, v_j \in \mathcal{V}$, $v_i \neq v_j$ and $d \in \mathbb{Q}$. Octagonal constraints can be encoded using potential constraints by splitting each variable v_i into two forms: a positive form v_i^+, interpreted as $+v_i$; and a negative form v_i^-, interpreted as $-v_i$. Then any octagonal constraint $av_i + bv_j \leq d$ can be written as a potential constraint $v - v' \leq d_0$ where $v, v' \in \{v_i^+, v_i^-, v_j^+, v_j^-\}$ and $d_0 \in \mathbb{Q}$. Namely, an octagonal constraint such as $v_i + v_j \leq d$ can be translated into the potential constraint $v_i^+ - v_j^- \leq d$; alternatively, the same octagonal constraint can be translated into $v_j^+ - v_i^- \leq d$. Furthermore, unary (octagonal) constraints such as $v_i \leq d$ and $-v_i \leq d$ can be encoded as $v_i^+ - v_i^- \leq 2d$ and $v_i^- - v_i^+ \leq 2d$, respectively.

From now on, we assume that the set of nodes is $\mathcal{N} := \{0, \dots, 2n-1\}$. These will denote the positive and negative forms of the variables in $\mathcal{V}$: for all $i \in \mathcal{N}$, if $i = 2k$, then i represents the positive form v_k^+ and, if $i = 2k+1$, then i represents the negative form v_k^- of the variable v_k. To simplify the presentation, for each $i \in \mathcal{N}$, we let $\bar{\imath}$ denote $i+1$, if i is even, and $i-1$, if i is odd, so that, for all $i \in \mathcal{N}$, we also have $\bar{\imath} \in \mathcal{N}$ and $\bar{\bar{\imath}} = i$. Then we can rewrite a potential constraint $v - v' \leq d$ where $v \in \{v_k^+, v_k^-\}$ and $v' \in \{v_l^+, v_l^-\}$ as the potential constraint $i - j \leq d$ in $\mathcal{N}$ where, if $v = v_k^+$, $i = 2k$ and, if $v = v_k^-$, $i = 2k+1$; similarly, if $v' = v_l^+$, $j = 2l$ and, if $v' = v_l^-$, $j = 2l+1$.

It follows from the above translations that any finite system of octagonal constraints, translated to a set of potential constraints in $\mathcal{N}$ as above, can be encoded by a graph G in $\mathcal{N}$. In particular, any finite *satisfiable* system of octagonal constraints can be encoded by a *consistent* graph in $\mathcal{N}$. However, the converse does not hold since in any valuation ρ of an encoding of a set of octagonal constraints we must also have $\rho(i) = -\rho(\bar{\imath})$, so that the arcs (i, j) and $(\bar{\jmath}, \bar{\imath})$ should have the same weight. Therefore, to encode rational octagonal constraints, we restrict attention to consistent graphs over $\mathcal{N}$ where the arcs in all such pairs are *coherent*.

Definition 2. (Octagonal graph.) *A (rational)* octagonal graph *is any consistent graph $G = (\mathcal{N}, w)$ that satisfies the coherence assumption:*

$$\forall i, j \in \mathcal{N} : w(i, j) = w(\bar{\jmath}, \bar{\imath}). \tag{3}$$

The set $\mathbb{O}$ of all octagonal graphs (with the usual addition of the bottom element, representing an unsatisfiable system of constraints) is a sub-lattice of $\mathbb{G}_\perp$, sharing the same least upper bound and greatest lower bound operators. Note that, at the implementation level, coherence can be automatically and efficiently enforced by letting arc (i, j) and arc $(\bar{\jmath}, \bar{\imath})$ share the same representation.

When dealing with octagonal graphs, one has to remember the relation linking the positive and negative forms of variables. A proper closure by entailment procedure should consider, besides transitivity, the following inference rule:

$$\frac{i - \bar{\imath} \le d_1 \qquad \bar{\jmath} - j \le d_2}{i - j \le \dfrac{d_1 + d_2}{2}} \tag{4}$$

Thus, the standard shortest-path closure algorithm is not enough to obtain a canonical form for octagonal graphs.

Definition 3. (Strongly closed graph.) *An octagonal graph* $G = (\mathcal{N}, w)$ *is* strongly closed *if it is closed and the following property holds:*

$$\forall i, j \in \mathcal{N} : 2w(i, j) \le w(i, \bar{\imath}) + w(\bar{\jmath}, j). \tag{5}$$

The strong closure *of an octagonal graph* G *in* $\mathcal{N}$ *is*

$$\text{S-closure}(G) := \bigsqcup \bigl\{\, G' \in \mathbb{O} \bigm| G' \trianglelefteq G \text{ and } G' \text{ is strongly closed} \,\bigr\}.$$

When trivially extended to the bottom element, strong closure is a kernel operator on the lattice of octagonal graphs.

A modified closure procedure is defined in [17], yielding strongly closed octagonal graphs. A significant efficiency improvement can be obtained thanks to the following theorem (for a proof, see Theorem 2 in [3]).

Theorem 1. *Let* $G = (\mathcal{N}, w)$ *be a closed octagonal graph. Consider the graph* $G_{\mathrm{S}} = (\mathcal{N}, w_{\mathrm{S}})$*, where* w_{S} *is defined, for each* $i, j \in \mathcal{N}$*, by*

$$w_{\mathrm{S}}(i, j) := \min\left\{ w(i, j), \frac{w(i, \bar{\imath})}{2} + \frac{w(\bar{\jmath}, j)}{2} \right\}.$$

Then $G_{\mathrm{S}} = \text{S-closure}(G)$*.*

Intuitively, the theorem states that strong closure can be obtained by application of any shortest-path closure algorithm followed by a *single* local propagation step using the constraint inference rule (4). In contrast, in the strong closure algorithm of [17], the outermost iterations of (a variant of) the Floyd-Warshall shortest-path algorithm are interleaved with n applications of the inference rule (4), leading to a more complex and less efficient implementation.

4 Integer Octagonal Graphs

We now consider the case of integer octagonal constraints, i.e., octagonal constraints where the bounds are all integral and the variables are only allowed to take integral values. These can be encoded by suitably restricting the codomain of the weight function of octagonal graphs.

Definition 4. (Integer octagonal graph.) *An* integer octagonal graph *is an octagonal graph* $G = (\mathcal{N}, w)$ *having an integral weight function:*

$$\forall i, j \in \mathcal{N} : w(i, j) \in \mathbb{Z} \cup \{+\infty\}.$$

As an integer octagonal graph is also a rational octagonal graph, the constraint system it encodes will be satisfiable when interpreted to take values in $\mathbb{Q}$. However, when interpreted to take values in $\mathbb{Z}$, this system may be unsatisfiable since the arcs encoding unary constraints can have an odd weight; we say that an octagonal graph is *$\mathbb{Z}$-consistent* if its encoded integer constraint system is satisfiable. For the same reason, the strong closure of an integer octagonal graph does not provide a canonical form for the integer constraint system it encodes and we need to consider the following *tightening* inference rule:

$$\frac{i - \bar{\imath} \leq d}{i - \bar{\imath} \leq 2\lfloor d/2 \rfloor}. \tag{6}$$

Definition 5. (Tightly closed graph.) *An octagonal graph $G = (\mathcal{N}, w)$ is* tightly closed *if it is a strongly closed integer octagonal graph and the following property holds:*

$$\forall i \in \mathcal{N} : w(i, \bar{\imath}) \text{ is even.} \tag{7}$$

The tight closure *of an octagonal graph G in $\mathcal{N}$ is*

$$\text{T-closure}(G) := \bigsqcup \bigl\{\, G' \in \mathbb{O} \bigm| G' \trianglelefteq G \text{ and } G' \text{ is tightly closed} \,\bigr\}.$$

By property (7), any tightly closed integer octagonal graph will encode a satisfiable integer constraint system and is therefore $\mathbb{Z}$-consistent. Moreover, since the encoding of any satisfiable integer constraint system will result in a $\mathbb{Z}$-consistent integer octagonal graph G that satisfies property (7), its tight closure T-closure(G) will also be $\mathbb{Z}$-consistent. This means that, if G is *not* $\mathbb{Z}$-consistent, then T-closure$(G) = \bigsqcup \varnothing = \bot$; that is, the tight closure operator computes either a tightly closed graph or the bottom element. Therefore, tight closure is a kernel operator on the lattice of octagonal graphs, as was the case for strong closure.

An incremental closure procedure for obtaining the tight closure of an octagonal graph was defined in [13] and improved in [12]. The algorithm, which is also presented and discussed in [19, Section 4.3.5], maintains the tight closure of a system of octagonal constraints by performing at most $\mathrm{O}(n^2)$ operations each time a new constraint is added: thus, for m constraints, the worst case complexity is $\mathrm{O}(mn^2)$. In particular, for the case of a dense system of octagonal constraints where $m \in \mathrm{O}(n^2)$, the worst case complexity is $\mathrm{O}(n^4)$.

The following theorem shows that a more efficient tight closure algorithm can be obtained by a simple modification to the improved strong closure algorithm of Theorem 1. Basically, inference rule (6) must be applied to ensure property (7) holds before applying inference rule (4).

Theorem 2. *Let $G = (\mathcal{N}, w)$ be a closed integer octagonal graph. Consider the graph $G_{\mathrm{T}} = (\mathcal{N}, w_{\mathrm{T}})$, where w_{T} is defined, for each $i, j \in \mathcal{N}$, by*

$$w_{\mathrm{T}}(i, j) := \min\left\{ w(i, j), \left\lfloor \frac{w(i, \bar{\imath})}{2} \right\rfloor + \left\lfloor \frac{w(\bar{\jmath}, j)}{2} \right\rfloor \right\}.$$

Then, if G_{T} is an octagonal graph, $G_{\mathrm{T}} =$ T-closure(G).

```
procedure tight_closure_if_consistent(var w [0..2n − 1] [0..2n − 1])
    { Classical Floyd-Warshall: O(n^3) }
    for k := 0 to 2n − 1 do
        for i := 0 to 2n − 1 do
            for j := 0 to 2n − 1 do
                w[i, j] := min(w[i, j], w[i, k] + w[k, j]);
    { Tight coherence: O(n^2) }
    for i := 0 to 2n − 1 do
        for j := 0 to 2n − 1 do
            w[i, j] := min(w[i, j], floor(w[i, ī]/2) + floor(w[j̄, j]/2));
```

Fig. 1. A $O(n^3)$ tight closure algorithm for $\mathbb{Z}$-consistent integer octagonal graphs

Figure 1 shows the pseudo-code for a $O(n^3)$ tight closure algorithm based on Theorem 2 and on the classical Floyd-Warshall shortest-path closure algorithm. Note that the pseudo-code in Figure 1 assumes that the data structure recording the weight function w, here denoted to be similar to a bidimensional array, automatically implements the coherence assumption for octagonal graphs (i.e., property (3) of Definition 2).

In the case of sparse graphs, a better complexity bound can be obtained by modifying the code in Figure 1 so as to compute the shortest path closure using Johnson's algorithm [7]: the worst case complexity of such an implementation will be $O(n^2 \log n + mn)$, which significantly improves upon the $O(mn^2)$ worst case complexity of [12, 13] when, e.g., $m \in \Theta(n)$. However, as observed elsewhere [19, 24], some of the targeted applications (e.g., static analysis) typically require the computation of graphs that are dense, so that the Floyd-Warshall algorithm is often a better choice from a practical perspective.

It is possible to define an incremental variant of the tight closure algorithm in Figure 1, which is simply based on the corresponding incremental version of the Floyd-Warshall shortest path closure algorithm. In such a case, we obtain the same worst case complexity of [12, 13].

The proof of Theorem 2 relies on a few auxiliary lemmas. The first two were also used in [3] for the formal proof of Theorem 1 above (for their detailed proofs, see Lemmas 9 and 10 in [3]).

Lemma 2. *Let $G = (\mathcal{N}, w)$ be an octagonal graph, $G^\star = (\mathcal{N}, w^\star) := \mathrm{closure}(G)$ and (z_1, z_2) be an arc in $G^\star$. Then there exists a simple path $\pi = z_1 \cdots z_2$ in G such that $w^\star(z_1, z_2) = w(\pi)$.*

Lemma 3. *Let $G = (\mathcal{N}, w)$ be a closed octagonal graph and $i, j \in \mathcal{N}$ be such that $i \neq \bar{\jmath}$ and $2w(i, j) \geq w(i, \bar{\imath}) + w(\bar{\jmath}, j)$. Let $G_\mathrm{s}^\star = (\mathcal{N}, w_\mathrm{s}^\star) := \mathrm{closure}(G_\mathrm{s})$*

where $G_{\mathrm{s}} := (\mathcal{N}, w_{\mathrm{s}})$ *and, for each* $h_1, h_2 \in \mathcal{N}$,

$$w_{\mathrm{s}}(h_1, h_2) := \begin{cases} \bigl(w(i, \bar{\imath}) + w(\bar{\jmath}, j)\bigr)/2, & \textit{if } (h_1, h_2) \in \bigl\{(i, j), (\bar{\jmath}, \bar{\imath})\bigr\}; \\ w(h_1, h_2), & \textit{otherwise.} \end{cases}$$

Let also $z_1, z_2 \in \mathcal{N}$. *Then one or both of the following hold:*

$$w_{\mathrm{s}}^{\star}(z_1, z_2) = w(z_1, z_2);$$
$$2w_{\mathrm{s}}^{\star}(z_1, z_2) \geq w(z_1, \bar{z}_1) + w(\bar{z}_2, z_2).$$

Informally, Lemma 3 states that if inference rule (4) is applied to a closed octagonal graph, then the resulting graph can be closed just by making further applications of inference rule (4). Note that, if G is an integer octagonal graph and property (7) holds, then the derived graph G_{s} will also be an integer octagonal graph. We now state a new lemma for integer octagonal graphs showing that when inference rule (6) is applied we obtain a similar conclusion to that for Lemma 3.

Lemma 4. *Let* $G = (\mathcal{N}, w)$ *be a closed integer octagonal graph and* $i \in \mathcal{N}$. *Let* $G_{\mathrm{t}}^{\star} := \mathrm{closure}(G_{\mathrm{t}})$ *where* $G_{\mathrm{t}} := (\mathcal{N}, w_{\mathrm{t}})$ *is an octagonal graph and, for each* $h_1, h_2 \in \mathcal{N}$,

$$w_{\mathrm{t}}(h_1, h_2) := \begin{cases} w(i, \bar{\imath}) - 1, & \textit{if } (h_1, h_2) = (i, \bar{\imath}); \\ w(h_1, h_2), & \textit{otherwise.} \end{cases} \tag{8}$$

Let $G_{\mathrm{t}}^{\star} = (\mathcal{N}, w_{\mathrm{t}}^{\star})$ *and* $z_1, z_2 \in \mathcal{N}$. *Then one or both of the following hold:*

$$w_{\mathrm{t}}^{\star}(z_1, z_2) = w(z_1, z_2), \tag{9}$$

$$w_{\mathrm{t}}^{\star}(z_1, z_2) \geq \left\lfloor \frac{w(z_1, \bar{z}_1)}{2} \right\rfloor + \left\lfloor \frac{w(\bar{z}_2, z_2)}{2} \right\rfloor. \tag{10}$$

Proof. By hypothesis and Definition 1, $G_{\mathrm{t}}^{\star} \trianglelefteq G_{\mathrm{t}} \trianglelefteq G$. If (z_1, z_2) is not an arc in $G_{\mathrm{t}}^{\star}$, then $w_{\mathrm{t}}^{\star}(z_1, z_2) = +\infty$; thus, as $G_{\mathrm{t}}^{\star} \trianglelefteq G$, we also have $w(z_1, z_2) = +\infty$ and hence property (9) holds. Suppose now that (z_1, z_2) is an arc in $G_{\mathrm{t}}^{\star}$. Then we can apply Lemma 2, so that there exists a simple path $\pi = z_1 \cdots z_2$ in G_{t} such that $w_{\mathrm{t}}^{\star}(z_1, z_2) = w_{\mathrm{t}}(\pi)$.

Suppose first that $w_{\mathrm{t}}(\pi) = w(\pi)$. Then, as G is closed, by Lemma 1 we obtain $w(\pi) \geq w(z_1, z_2)$ so that $w_{\mathrm{t}}^{\star}(z_1, z_2) \geq w(z_1, z_2)$. However $G_{\mathrm{t}}^{\star} \trianglelefteq G$ so that $w_{\mathrm{t}}^{\star}(z_1, z_2) \leq w(z_1, z_2)$ and therefore property (9) holds.

Secondly, suppose that $w_{\mathrm{t}}(\pi) \neq w(\pi)$. Then, by Equation (8), $(i, \bar{\imath})$ must be an arc in π, so that

$$\pi = \pi_1 :: (i\,\bar{\imath}) :: \pi_2, \tag{11}$$

where $\pi_1 = z_1 \cdots i$, $\pi_2 = j \cdots z_2$ are simple paths in G_{t} that do not contain the arc $(i, \bar{\imath})$. Therefore, by Equation (8), we have $w_{\mathrm{t}}(\pi_1) = w(\pi_1)$, $w_{\mathrm{t}}(\pi_2) = w(\pi_2)$.

Consider (11) and let[4]

$$\pi_1' = \pi_1 :: (i\,\bar{\imath}) :: \overline{\pi}_1, \qquad \pi_2' = \overline{\pi}_2 :: (i\,\bar{\imath}) :: \pi_2.$$

[4] If $\pi = j_0 \cdots j_p$ is a path in a graph in $\mathcal{N}$, then $\overline{\pi}$ denotes the path $\bar{\jmath}_p \cdots \bar{\jmath}_0$.

As G is an octagonal graph, we have $w(\pi_1) = w(\overline{\pi}_1)$ and $w(\pi_2) = w(\overline{\pi}_2)$ so that

$$w(\pi_1') = 2w(\pi_1) + w(i, \overline{\imath}), \qquad w(\pi_2') = 2w(\pi_2) + w(i, \overline{\imath}).$$

As G is closed, by Lemma 1,

$$w(\pi_1') \geq w(z_1, \overline{z}_1), \qquad w(\pi_2') \geq w(\overline{z}_2, z_2)$$

so that

$$w(\pi_1) + \frac{w(i, \overline{\imath})}{2} \geq \frac{w(z_1, \overline{z}_1)}{2}, \qquad w(\pi_2) + \frac{w(i, \overline{\imath})}{2} \geq \frac{w(\overline{z}_2, z_2)}{2}.$$

Therefore

$$\begin{aligned}
w_{\mathrm{t}}(\pi) &= w_{\mathrm{t}}(\pi_1) + w_{\mathrm{t}}(i, \overline{\imath}) + w_{\mathrm{t}}(\pi_2) \\
&= w(\pi_1) + \frac{w(i, \overline{\imath}) - 1}{2} + w(\pi_2) + \frac{w(i, \overline{\imath}) - 1}{2} \\
&\geq \frac{w(z_1, \overline{z}_1)}{2} - \frac{1}{2} + \frac{w(\overline{z}_2, z_2)}{2} - \frac{1}{2} \\
&\geq \Big\lfloor \frac{w(z_1, \overline{z}_1)}{2} \Big\rfloor + \Big\lfloor \frac{w(\overline{z}_2, z_2)}{2} \Big\rfloor.
\end{aligned}$$

Hence, as $w_{\mathrm{t}}^{\star}(z_1, z_2) = w_{\mathrm{t}}(\pi)$, we obtain property (10), as required. □

The next result, uses Lemmas 3 and 4 to derive a property relating the weight functions for a closed integer octagonal graph and its tight closure.

Lemma 5. *Let $G = (\mathcal{N}, w)$ be a closed integer octagonal graph such that $G^{\mathrm{T}} = (\mathcal{N}, w^{\mathrm{T}}) :=$ T-closure(G) is an octagonal graph and let $z_1, z_2 \in \mathcal{N}$. Then one or both of the following hold:*

$$w^{\mathrm{T}}(z_1, z_2) = w(z_1, z_2); \tag{12}$$

$$w^{\mathrm{T}}(z_1, z_2) = \Big\lfloor \frac{w(z_1, \overline{z}_1)}{2} \Big\rfloor + \Big\lfloor \frac{w(\overline{z}_2, z_2)}{2} \Big\rfloor. \tag{13}$$

Proof. The proof is by contraposition; thus we assume that neither (12) nor (13) hold. Without loss of generality, let the graph G be $\trianglelefteq$-minimal in the set of all closed integer octagonal graphs such that T-closure$(G) = G^{\mathrm{T}}$ and for which neither (12) nor (13) hold. Clearly the negation of (12) implies that $G \neq G^{\mathrm{T}}$, so that $G^{\mathrm{T}} \vartriangleleft G$.

As G is closed but not tightly closed, by Definitions 3 and 5, it follows that there exist $i, j \in \mathcal{N}$ such that either

(i) $i = \overline{\jmath}$ and $w(i, \overline{\imath})$ is odd; or
(ii) property (7) holds and $2w(i, j) > w(i, \overline{\imath}) + w(\overline{\jmath}, j)$.

Consider graph $G_1 = (\mathcal{N}, w_1)$ where the weight function w_1 is defined, for all $h_1, h_2 \in \mathcal{N}$, by

$$w_1(h_1, h_2) := \begin{cases} \left\lfloor \frac{w(i,\overline{\imath})}{2} \right\rfloor + \left\lfloor \frac{w(\overline{\jmath},j)}{2} \right\rfloor, & \text{if } (h_1, h_2) \in \bigl\{(i,j), (\overline{\jmath}, \overline{\imath})\bigr\}; \\ w(h_1, h_2), & \text{otherwise.} \end{cases}$$

Let $G_1^\star = \operatorname{closure}(G_1)$. By Definitions 1, 3 and 5,

$$G^{\mathrm{T}} \trianglelefteq G_1^\star \trianglelefteq G_1 \triangleleft G. \tag{14}$$

Thus $\text{T-closure}(G_1^\star) = G^{\mathrm{T}}$ so that, by the minimality assumption on G, one or both of the following hold:

$$w^{\mathrm{T}}(z_1, z_2) = w_1^\star(z_1, z_2); \tag{15}$$

$$w^{\mathrm{T}}(z_1, z_2) = \left\lfloor \frac{w_1^\star(z_1, \overline{z}_1)}{2} \right\rfloor + \left\lfloor \frac{w_1^\star(\overline{z}_2, z_2)}{2} \right\rfloor. \tag{16}$$

As $G^{\mathrm{T}} \neq \bot$, by (14), G_1 is consistent. Therefore, by construction, G_1 is an integer octagonal graph. If property (i) holds for i, j, then Lemma 4 can be applied and, if property (ii) holds for i, j, then Lemma 3 can be applied and also, since property (7) holds, both $w_1(z_1, \overline{z}_1)$ and $w(\overline{z}_2, z_2)$ are even. Hence, letting $G_1^\star := (\mathcal{N}, w_1^\star)$, one or both of the following hold:

$$w_1^\star(z_1, z_2) = w(z_1, z_2); \tag{17}$$

$$w_1^\star(z_1, z_2) \geq \left\lfloor \frac{w(z_1, \overline{z}_1)}{2} \right\rfloor + \left\lfloor \frac{w(\overline{z}_2, z_2)}{2} \right\rfloor. \tag{18}$$

Again by Lemmas 3 and 4,

$$w_1^\star(z_1, \overline{z}_1) \geq 2\left\lfloor \frac{w(z_1, \overline{z}_1)}{2} \right\rfloor,$$

$$w_1^\star(\overline{z}_2, z_2) \geq 2\left\lfloor \frac{w(\overline{z}_2, z_2)}{2} \right\rfloor;$$

since the lower bounds for $w_1^\star(z_1, \overline{z}_1)$ and $w_1^\star(\overline{z}_2, z_2)$ are even integers, we obtain

$$\left\lfloor \frac{w_1^\star(z_1, \overline{z}_1)}{2} \right\rfloor + \left\lfloor \frac{w_1^\star(\overline{z}_2, z_2)}{2} \right\rfloor \geq \left\lfloor \frac{w(z_1, \overline{z}_1)}{2} \right\rfloor + \left\lfloor \frac{w(\overline{z}_2, z_2)}{2} \right\rfloor. \tag{19}$$

Suppose first that (15) and (17) hold. Then by transitivity we obtain (12), contradicting the contrapositive assumption for G.

If (15) and (18) hold, then it follows

$$w^{\mathrm{T}}(z_1, z_2) \geq \left\lfloor \frac{w(z_1, \overline{z}_1)}{2} \right\rfloor + \left\lfloor \frac{w(\overline{z}_2, z_2)}{2} \right\rfloor. \tag{20}$$

On the other hand, if (16) holds, then, by (19), we obtain again property (20). However, by Definition 5 we also have

$$w^{\mathrm{T}}(z_1, z_2) \leq \left\lfloor \frac{w(z_1, \overline{z}_1)}{2} \right\rfloor + \left\lfloor \frac{w(\overline{z}_2, z_2)}{2} \right\rfloor.$$

By combining this inequality with (20) we obtain (13), contradicting the contrapositive assumption for G. $\square$

Proof (of Theorem 2). Let $G^{\mathrm{T}} := \text{T-closure}(G)$. By definition of G_{T}, $G_{\mathrm{T}} \trianglelefteq G$ so that $\text{T-closure}(G_{\mathrm{T}}) \trianglelefteq G^{\mathrm{T}}$. As G_{T} is an octagonal graph, G_{T} is consistent, and hence $G^{\mathrm{T}} \neq \bot$; let $G^{\mathrm{T}} = (\mathcal{N}, w^{\mathrm{T}})$. Letting $i, j \in \mathcal{N}$, to prove the result we need to show that $w^{\mathrm{T}}(i,j) = w_{\mathrm{T}}(i,j)$. Let $k_{ij} := \lfloor w(i,\bar{\imath})/2 \rfloor + \lfloor w(\bar{\jmath},j)/2 \rfloor$.

By Definitions 1, 3 and 5, it follows that both properties $w^{\mathrm{T}}(i,j) \leq w(i,j)$ and $w^{\mathrm{T}}(i,j) \leq k_{ij}$ hold so that, by definition of w_{T}, we have $w^{\mathrm{T}}(i,j) \leq w_{\mathrm{T}}(i,j)$. By Lemma 5, $w^{\mathrm{T}}(i,j) = w(i,j)$ and/or $w^{\mathrm{T}}(i,j) = k_{ij}$. Therefore since, by definition, $w_{\mathrm{T}}(i,j) = \min\{w(i,j), k_{ij}\}$, we obtain $w_{\mathrm{T}}(i,j) \leq w^{\mathrm{T}}(i,j)$. $\square$

It follows from the statement of Theorem 2 that an implementation based on it also needs to check the consistency of G_{T}. In principle, one could apply again a shortest-path closure procedure so as to check whether G_{T} contains some negative weight cycles. Fortunately, a much more efficient solution is obtained by the following result.

Theorem 3. *Let $G = (\mathcal{N}, w)$ be a closed integer octagonal graph. Consider the graphs $G_{\mathrm{t}} = (\mathcal{N}, w_{\mathrm{t}})$ and $G_{\mathrm{T}} = (\mathcal{N}, w_{\mathrm{T}})$ where, for each $i, j \in \mathcal{N}$,*

$$w_{\mathrm{t}}(i,j) := \begin{cases} 2\lfloor w(i,j)/2 \rfloor, & \text{if } j = \bar{\imath}; \\ w(i,j), & \text{otherwise}; \end{cases} \tag{21}$$

$$w_{\mathrm{T}}(i,j) := \min\left\{ w(i,j), \left\lfloor \frac{w(i,\bar{\imath})}{2} \right\rfloor + \left\lfloor \frac{w(\bar{\jmath},j)}{2} \right\rfloor \right\}. \tag{22}$$

Suppose that, for all $i \in \mathcal{N}$, $w_{\mathrm{t}}(i,\bar{\imath}) + w_{\mathrm{t}}(\bar{\imath},i) \geq 0$. Then G_{T} is an octagonal graph.

This result is a corollary of the following result proved in [15, Lemma 4].

Lemma 6. *Let $G = (\mathcal{N}, w)$ be an integer octagonal graph with no negative weight cycles and $G_{\mathrm{t}} = (\mathcal{N}, w_{\mathrm{t}})$, where w_{t} satisfies* (21)*, have a negative weight cycle. Then there exists $i, \bar{\imath} \in \mathcal{N}$ and a cycle $\pi = (i \cdot \pi_1 \cdot \bar{\imath}) :: (\bar{\imath} \cdot \pi_2 \cdot i)$ in G such that $w(\pi) = 0$ and the weight of the shortest path in G from i to $\bar{\imath}$ is odd.*

Proof (of Theorem 3). The proof is by contradiction; suppose G_{T} is not an octagonal graph; then by Definitions 1, 3 and 5, G_{T} is inconsistent. We show that G_{t} is also inconsistent. Again, we assume to the contrary that G_{t} is consistent and derive a contradiction. Let $i, j \in \mathcal{N}$. By (21), we have $w_{\mathrm{t}}(i,j) \leq w(i,j)$ and $w_{\mathrm{t}}(i,\bar{\imath})/2 + w_{\mathrm{t}}(\bar{\jmath},j)/2 = k_{ij}$, where $k_{ij} := \lfloor w(i,\bar{\imath})/2 \rfloor + \lfloor w(\bar{\jmath},j)/2 \rfloor$. Letting $\text{S-closure}(G_{\mathrm{t}}) = (\mathcal{N}, w_{\mathrm{t}}^{\mathrm{S}})$, we have, by Definition 3, $w_{\mathrm{t}}^{\mathrm{S}}(i,j) \leq w_{\mathrm{t}}(i,j)$ and $w_{\mathrm{t}}^{\mathrm{S}}(i,j) \leq w_{\mathrm{t}}(i,\bar{\imath})/2 + w_{\mathrm{t}}(\bar{\jmath},j)/2$. Thus $w_{\mathrm{t}}^{\mathrm{S}}(i,j) \leq \min\bigl(w(i,j), k_{ij}\bigr)$. As this holds for all $i, j \in \mathcal{N}$, by (22), $\text{S-closure}(G_{\mathrm{t}}) \trianglelefteq G_{\mathrm{T}}$, contradicting the assumption that G_{t} was consistent. Hence G_{t} is inconsistent and therefore contains a negative weight cycle.

By Lemma 6, there exists $i, \bar{\imath} \in \mathcal{N}$ and a cycle $\pi = (i \cdot \pi_1 \cdot \bar{\imath}) :: (\bar{\imath} \cdot \pi_2 \cdot i)$ in G such that $w(\pi) = 0$ and the weight of the shortest path in G from i to

$\bar{\imath}$ is odd. As G is closed, $w(i, \bar{\imath}) \le w(i \cdot \pi_1 \cdot \bar{\imath})$ and $w(\bar{\imath}, i) \le w(\bar{\imath} \cdot \pi_2 \cdot i)$. Thus $w(i, \bar{\imath}) + w(\bar{\imath}, i) \le w(\pi) = 0$. Moreover, $(i\bar{\imath})$ is a path and hence the shortest path from i to $\bar{\imath}$ so that $w(i\bar{\imath})$ is odd; hence, by (21), $w(i, \bar{\imath}) = w_{\mathrm{t}}(i, \bar{\imath}) + 1$ and $w(\bar{\imath}, i) \ge w_{\mathrm{t}}(\bar{\imath}, i)$. Therefore $w_{\mathrm{t}}(i, \bar{\imath}) + w_{\mathrm{t}}(\bar{\imath}, i) < 0$. □

```
function tight_closure(var w[0..2n − 1][0..2n − 1]) : bool
    { Initialization: O(n) }
    for i := 0 to 2n − 1 do w[i,i] := 0;
    { Classical Floyd-Warshall: O(n³) }
    for k := 0 to 2n − 1 do
        for i := 0 to 2n − 1 do
            for j := 0 to 2n − 1 do
                w[i,j] := min(w[i,j], w[i,k] + w[k,j]);
    { Check for ℚ-consistency: O(n) }
    for i := 0 to 2n − 2 step 2 do
        if w[i,i] < 0 return false;
    { Tightening: O(n) }
    for i := 0 to 2n − 1 do
        w[i,ī] := floor(w[i,ī]/2);
    { Check for ℤ-consistency: O(n) }
    for i := 0 to 2n − 2 step 2 do
        if w[i,ī] + w[ī,i] < 0 return false;
    { Strong coherence: O(n²) }
    for i := 0 to 2n − 1 do
        for j := 0 to 2n − 1 do
            w[i,j] := min(w[i,j], w[i,ī]/2 + w[j̄,j]/2);
    return true;
```

Fig. 2. A $\mathrm{O}(n^3)$ tight closure algorithm for integer coherent graphs

The combination of the results stated in Theorems 2 and 3 (together with the well known result for rational consistency) leads to an $\mathrm{O}(n^3)$ tight closure algorithm, such as that given by the pseudo-code in Figure 2, that computes the tight closure of any (possibly inconsistent) coherent integer-weighted graph returning the Boolean value 'true' if and only if the input graph is $\mathbb{Z}$-consistent.

5 Conclusion and Future Work

We have presented and fully justified an $\mathrm{O}(n^3)$ algorithm that computes the tight closure of a set of integer octagonal constraints. The algorithm —which is based on the extension to integer-weighted octagonal graphs of the one we proposed for rational-weighted octagonal graphs [2, 3]— and its proof of correctness means the issue about the possibility of computing the tight closure at a computational cost that is asymptotically not worse than the cost of computing all-pairs shortest paths is finally closed.

In the field of hardware and software verification, the integrality constraint that distinguishes integer-weighted from rational-weighted octagonal graphs can be seen as an abstraction of the more general imposition of a set of congruence relations. Such a set can be encoded by an element of a suitable abstract domain such as the non-relational congruence domain of [10] (that is, of the form $x = a \pmod{b}$), the weakly relational *zone-congruence* domain of [18] (that is, also allowing the form $x - y = a \pmod{b}$), the linear congruence domain of [11], and the more general fully relational *rational grids* domain developed in [1]. The combination of such domains with the abstract domain proposed in [2, 3] is likely to provide an interesting complexity-precision trade-off. Future work includes investigating such a combination, exploiting the ideas presented in this paper.

References

1. R. Bagnara, K. Dobson, P. M. Hill, M. Mundell, and E. Zaffanella. Grids: A domain for analyzing the distribution of numerical values. In G. Puebla, editor, *Logic-based Program Synthesis and Transformation, 16th International Symposium*, volume 4407 of *Lecture Notes in Computer Science*, pages 219–235, Venice, Italy, 2007. Springer-Verlag, Berlin.
2. R. Bagnara, P. M. Hill, E. Mazzi, and E. Zaffanella. Widening operators for weakly-relational numeric abstractions. In C. Hankin and I. Siveroni, editors, *Static Analysis: Proceedings of the 12th International Symposium*, volume 3672 of *Lecture Notes in Computer Science*, pages 3–18, London, UK, 2005. Springer-Verlag, Berlin.
3. R. Bagnara, P. M. Hill, E. Mazzi, and E. Zaffanella. Widening operators for weakly-relational numeric abstractions. Quaderno 399, Dipartimento di Matematica, Università di Parma, Italy, 2005. Available at `http://www.cs.unipr.it/Publications/`.
4. R. Bagnara, P. M. Hill, and E. Zaffanella. The Parma Polyhedra Library: Toward a complete set of numerical abstractions for the analysis and verification of hardware and software systems. Quaderno 457, Dipartimento di Matematica, Università di Parma, Italy, 2006. Available at `http://www.cs.unipr.it/Publications/`. Also published as `arXiv:cs.MS/0612085`, available from `http://arxiv.org/`.
5. V. Balasundaram and K. Kennedy. A technique for summarizing data access and its use in parallelism enhancing transformations. In B. Knobe, editor, *Proceedings of the ACM SIGPLAN'89 Conference on Programming Language Design and Implementation (PLDI)*, volume 24(7) of *ACM SIGPLAN Notices*, pages 41–53, Portland, Oregon, USA, 1989. ACM Press.

6. T. Ball, B. Cook, S. K. Lahiri, and L. Zhang. Zapato: Automatic theorem proving for predicate abstraction refinement. In R. Alur and D. Peled, editors, *Computer Aided Verification: Proceedings of the 16th International Conference*, volume 3114 of *Lecture Notes in Computer Science*, pages 457–461, Boston, MA, USA, 2004. Springer-Verlag, Berlin.
7. T. H. Cormen, T. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. The MIT Press, Cambridge, MA, 1990.
8. P. Cousot and R. Cousot. Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints. In *Proceedings of the Fourth Annual ACM Symposium on Principles of Programming Languages*, pages 238–252, New York, 1977. ACM Press.
9. P. Cousot, R. Cousot, J. Feret, L. Mauborgne, A. Miné, D. Monniaux, and X. Rival. The ASTRÉE analyzer. In M. Sagiv, editor, *Programming Languages and Systems, Proceedings of the 14th European Symposium on Programming*, volume 3444 of *Lecture Notes in Computer Science*, pages 21–30, Edinburgh, UK, 2005. Springer-Verlag, Berlin.
10. P. Granger. Static analysis of arithmetical congruences. *International Journal of Computer Mathematics*, 30:165–190, 1989.
11. P. Granger. Static analysis of linear congruence equalities among variables of a program. In S. Abramsky and T. S. E. Maibaum, editors, *TAPSOFT'91: Proceedings of the International Joint Conference on Theory and Practice of Software Development, Volume 1: Colloquium on Trees in Algebra and Programming (CAAP'91)*, volume 493 of *Lecture Notes in Computer Science*, pages 169–192, Brighton, UK, 1991. Springer-Verlag, Berlin.
12. W. Harvey and P. J. Stuckey. A unit two variable per inequality integer constraint solver for constraint logic programming. In M. Patel, editor, *ACSC'97: Proceedings of the 20th Australasian Computer Science Conference*, volume 19, pages 102–111. Australian Computer Science Communications, 1997.
13. J. Jaffar, M. J. Maher, P. J. Stuckey, and R. H. C. Yap. Beyond finite domains. In A. Borning, editor, *Principles and Practice of Constraint Programming: Proceedings of the Second International Workshop*, volume 874 of *Lecture Notes in Computer Science*, pages 86–94, Rosario, Orcas Island, Washington, USA, 1994. Springer-Verlag, Berlin.
14. J. C. Lagarias. The computational complexity of simultaneous Diophantine approximation problems. *SIAM Journal on Computing*, 14(1):196–209, 1985.
15. S. K. Lahiri and M. Musuvathi. An efficient decision procedure for UTVPI constraints. In B. Gramlich, editor, *Frontiers of Combining Systems: Proceedings of the 5th International Workshop, FroCoS 2005*, volume 3717 of *Lecture Notes in Artificial Intelligence*, pages 168–183, Vienna, Austria, 2005. Springer-Verlag, Berlin.
16. A. Miné. A new numerical abstract domain based on difference-bound matrices. In O. Danvy and A. Filinski, editors, *Proceedings of the 2nd Symposium on Programs as Data Objects (PADO 2001)*, volume 2053 of *Lecture Notes in Computer Science*, pages 155–172, Aarhus, Denmark, 2001. Springer-Verlag, Berlin.
17. A. Miné. The octagon abstract domain. In *Proceedings of the Eighth Working Conference on Reverse Engineering (WCRE'01)*, pages 310–319, Stuttgart, Germany, 2001. IEEE Computer Society Press.
18. A. Miné. A few graph-based relational numerical abstract domains. In M. V. Hermenegildo and G. Puebla, editors, *Static Analysis: Proceedings of the 9th International Symposium*, volume 2477 of *Lecture Notes in Computer Science*, pages 117–132, Madrid, Spain, 2002. Springer-Verlag, Berlin.

19. A. Miné. *Weakly Relational Numerical Abstract Domains.* PhD thesis, École Polytechnique, Paris, France, March 2005.
20. A. Miné. The octagon abstract domain. *Higher-Order and Symbolic Computation*, 19(1):31–100, 2006.
21. G. Nelson and D. C. Oppen. Fast decision algorithms based on Union and Find. In *Proceedings of the 18th Annual Symposium on Foundations of Computer Science (FOCS'77)*, pages 114–119, Providence, RI, USA, 1977. IEEE Computer Society Press. The journal version of this paper is [22].
22. G. Nelson and D. C. Oppen. Fast decision procedures based on congruence closure. *Journal of the ACM*, 27(2):356–364, 1980. An earlier version of this paper is [21].
23. V. R. Pratt. Two easy theories whose combination is hard. Memo sent to Nelson and Oppen concerning a preprint of their paper [21], September 1977.
24. A. Venet and G. Brat. Precise and efficient static array bound checking for large embedded C programs. In *Proceedings of the ACM SIGPLAN 2004 Conference on Programming Language Design and Implementation (PLDI'04)*, pages 231–242, Washington, DC, USA, 2004. ACM Press.