Precise Widening Operators for Convex Polyhedra ⋆

Roberto Bagnara [a], Patricia M. Hill [b], Elisa Ricci [a], Enea Zaffanella [a]

[a] *Department of Mathematics, University of Parma, Italy*

[b] *School of Computing, University of Leeds, UK*

Abstract

In the context of static analysis via abstract interpretation, convex polyhedra constitute the most used abstract domain among those capturing numerical relational information. Since the domain of convex polyhedra admits infinite ascending chains, it has to be used in conjunction with appropriate mechanisms for enforcing and accelerating the convergence of fixpoint computations. Widening operators provide a simple and general characterization for such mechanisms. For the domain of convex polyhedra, the original widening operator proposed by Cousot and Halbwachs amply deserves the name of *standard widening* since most analysis and verification tools that employ convex polyhedra also employ that operator. Nonetheless, there is an unfulfilled demand for more precise widening operators. In this paper, after a formal introduction to the standard widening where we clarify some aspects that are often overlooked, we embark on the challenging task of improving on it. We present a framework for the systematic definition of new widening operators that are never less precise than a given widening. The framework is then instantiated on the domain of convex polyhedra so as to obtain a new widening operator that improves on the standard widening by combining several heuristics. A preliminary experimental evaluation has yielded promising results. We also suggest an improvement to the well-known widening delay technique that allows to gain precision while preserving its overall simplicity.

Key words: abstract interpretation, widening operators, convex polyhedra.

⋆ This work has been partly supported by MURST projects "Aggregate- and Number-Reasoning for Computing: from Decision Algorithms to Constraint Programming with Multisets, Sets, and Maps" and "Constraint Based Verification of Reactive Systems."

Email addresses: `bagnara@cs.unipr.it` (Roberto Bagnara), `hill@comp.leeds.ac.uk` (Patricia M. Hill), `ericci@cs.unipr.it` (Elisa Ricci),

1 Introduction

An ability to reason about numerical quantities is crucial for increasing numbers of applications in the field of automated analysis and verification of complex systems. Of particular interest are representations that capture *relational information*, that is, information relating different quantities such as, for example, the length of a buffer and the contents of a program variable, or the number of agents in different states in the modeling of a distributed protocol.

Convex polyhedra, since the work of Cousot and Halbwachs [1], constitute the most used abstract domain among those capturing numerical, relational information. They have been used to solve, by abstract interpretation [2], several important data-flow analysis problems such as array bound checking, compile-time overflow detection, loop invariant computations and loop induction variables. Convex polyhedra are also used, among many other applications, for the analysis and verification of synchronous languages [3,4] and of linear hybrid automata (an extension of finite-state machines that models time requirements) [5,6], for the computer-aided formal verification of concurrent and reactive systems based on temporal specifications [7], for inferring argument size relationships in logic languages [8,9], for the automatic parallelization of imperative programs [10], for detecting buffer overflows in C [11], and for the automatic generation of the ranking functions needed to prove progress properties [12].

Since the domain of convex polyhedra admits infinite ascending chains, it has to be used in conjunction with appropriate mechanisms for enforcing and accelerating the convergence of fixpoint computations. *Widening operators* [2,13–15] provide a simple and general characterization for such mechanisms. In its simplest form, a widening operator on a poset $\langle P, \sqsubseteq \rangle$ is defined as a partial function $\nabla \colon P \times P \rightarrowtail P$ satisfying:

(1) for all $x, y \in P$, if $x \mathbin{\nabla} y$ is defined then $x \sqsubseteq x \mathbin{\nabla} y$ and $y \sqsubseteq x \mathbin{\nabla} y$;
(2) for all increasing chains $y_0 \sqsubseteq y_1 \sqsubseteq \cdots$, if the increasing chain $x_0 := y_0$ and $x_{i+1} := x_i \mathbin{\nabla} y_{i+1}$ is defined for all $i \in \mathbb{N}$, then it is not strictly increasing.

It must be observed that a widening operator may serve different purposes, besides forcing the stabilization of approximated iteration sequences after a finite number of iterations: it may be used to speed up the convergence of iteration sequences and to select among a (possibly infinite) set of approximations of concrete elements when considering abstract domains that are algebraically weak [14]. Thus a widening does not need to be a total function, the only requirement is that its domain of definition be compatible with the intended application. The application will also affect the required trade-off between pre-

`zaffanella@cs.unipr.it` (Enea Zaffanella).

cision and efficiency: when speeding up convergence of an (perhaps intrinsically finite) iteration sequence, precision is more willingly given away; in other cases, the objective is to ensure termination without compromising precision too much. As a consequence, it is meaningful to have two or more widening operators, each one tuned with a different compromise between precision and efficiency. The different widenings can be used in different applications or even in the same application, with the system dynamically switching from one to another [16].

For the domain of convex polyhedra, the first widening operator was proposed by Cousot and Halbwachs in [1] and further refined in [17]. It amply deserves the name of *standard widening* since most analysis and verification tools that employ convex polyhedra also employ that operator.

There are a number of applications of convex polyhedra in the field of systems analysis and verification that are particularly sensitive to the precision of the deduced numerical information. The importance of precision in the field of automated verification has led to the use of *extrapolation operators*, that is, binary operators satisfying condition 1 in the definition of widening but not condition 2 (i.e., without convergence guarantees). For instance, in [18], Henzinger and Ho propose a new extrapolation operator for use in the HyTech model checker since "Halbwachs's widening operator [...] is sometimes too coarse for [their] purposes" (symbolic model checking of linear hybrid systems). An even more precise extrapolation operator, also used in the HyTech system, is presented in [19]: "This operator is tighter than (and therefore less aggressive than) both the widening operator of [4] and the extrapolation operator of [18], which is not monotone in its second argument." Other extrapolation operators based on similar approaches have been sketched in [3]. Still in the field of automatic verification, the need for more precision than warranted by the standard widening is remarked in both [20] and [21]; and a new extrapolation operator on sets of convex polyhedra is defined in each of these papers.

If giving up convergence guarantees is acceptable (though not desirable) for semi-automatic, human-operated verifiers, this is certainly not the case for fully-automatic program analyzers. In this field, the request for more precision has been partly satisfied by delaying the application of the widening operator k times for some fixed parameter $k \in \mathbb{N}$ [16]. A study of the effect of alternative values for k in the automatic determination of linear size relations between the arguments of logic programs has been conducted in [8,9]. One application of this idea is in termination inference [22]. In order to achieve reasonable precision, the cTI analyzer runs with $k = 3$ as a default, but there are simple programs (such as *mergesort*) whose termination can only be established with $k > 3$. On the other hand, setting $k = 4$ as the default can have a sensible impact on performance of cTI [F. Mesnard, personal communication, 2003]. Another technique to improve upon the results of a widening, while still ensuring termination, is described in [4,23] and named 'widening up to'. This is meant to recover from those extrapolations that go beyond the limits specified by a *fixed* set of constraints, which are specific of the application domain under consideration or have been obtained by a previous static analysis step.

In this paper, after a formal introduction to the standard widening where we clarify some important aspects that are often overlooked, we embark on the challenging task of improving on it. Elaborating on an idea originally proposed in [3], we present a domain-independent framework for the systematic definition of new widenings that are never less precise than a given widening operator. Their specification is based on the definition of a computable preorder relation which satisfies the ascending chain condition on the considered abstract domain and is compatible with the widening we are improving upon. The framework makes it particularly easy to combine several heuristics and prove that the resulting operator is indeed a widening at least as precise as the original widening. Here we instantiate it on the domain of convex polyhedra so as to obtain a widening operator improving on the standard widening. In particular, we consider a selection of extrapolation operators, some of which embody improvements of heuristics already proposed in the literature. An experimental evaluation of the new widening shows that, for the analysis problem considered, it captures common growth patterns and obtains precision improvements in as many as 33% of the benchmarks.

The paper is structured as follows: Section 2 recalls the required concepts and notations; Section 3 introduces the standard widening for the domain of convex polyhedra, highlighting a few important aspects of its formal definition that are often overlooked; Section 4 presents a domain-independent framework for the systematic definition of new widening operators improving upon any existing widening; Section 5 instantiates this framework to the domain of convex polyhedra by considering several variants of extrapolations techniques proposed in the literature, as well as one that is new to this paper; Section 6 summarizes the results of our experimental evaluation of the new widening; Section 7 proposes an improvement of the well-known widening delay technique. Section 8 concludes. This paper is a revised and extended version of [24].

2 Preliminaries

A *preorder* '$\preceq$' over a set S is a binary relation that is reflexive and transitive. A preorder is an *equivalence relation* (resp., a *partial order*) if it is also symmetric (resp., antisymmetric). A preorder '$\preceq$' induces an equivalence relation '$\equiv$' on S such that, for each $x, y \in S$, $x \equiv y$ if and only if both $x \preceq y$ and $y \preceq x$. The *strict* version '$\prec$' of a preorder '$\preceq$' is the relation such that, for each $x, y \in S$,

$x \prec y$ if and only if $x \preceq y$ and $x \not\equiv y$.

A *poset*, denoted by $\langle P, \sqsubseteq \rangle$, is a set P equipped with a partial order '$\sqsubseteq$'. A *chain* over the poset $\langle P, \sqsubseteq \rangle$ is a subset $C \subseteq P$ such that '$\sqsubseteq$' is a *total order* on C, i.e., for each $x, y \in C$ such that $x \neq y$, either $x \sqsubset y$ or $y \sqsubset x$. A poset satisfies the *ascending chain condition* if all its strictly increasing chains are finite. A preorder '$\preceq$' on a set S induces the poset $\langle S/\equiv, \sqsubseteq \rangle$: the set $S/\equiv$ is the quotient of S with respect to the equivalence relation '$\equiv$' induced by '$\preceq$'; and the partial order '$\sqsubseteq$' is such that $[x] \sqsubseteq [y]$ if and only if $x \preceq y$, for all equivalence classes $[x], [y] \in S/\equiv$. In the following, with a minor abuse of terminology and notation, we will sometimes define preorders on sets and later state properties that actually hold for the implicitly induced posets. For instance, a preorder '$\preceq$' will be said to satisfy the ascending chain condition on a set S to mean that the induced poset $\langle S/\equiv, \sqsubseteq \rangle$ satisfies the ascending chain condition.

The *lexicographic product of the preorders* '$\preceq_a$' *and* '$\preceq_b$' is the preorder '$\preceq_{ab}$' on S such that, for all $x, y \in S$,

$$x \preceq_{ab} y \quad \overset{\text{def}}{\iff} \quad (x \prec_a y) \vee (x \equiv_a y \wedge x \preceq_b y).$$

If both '$\preceq_a$' and '$\preceq_b$' satisfy the ascending chain condition on S, then '$\preceq_{ab}$' satisfies the ascending chain condition too. If '$\preceq$' is a preorder on S and $\bot \notin S$, then the $\bot$-*lifting* of '$\preceq$' is obtained by defining $\bot \prec x$ for all $x \in S$. If '$\preceq$' satisfies the ascending chain condition on S, then its $\bot$-lifting satisfies the ascending chain condition on $\{\bot\} \cup S$.

Let $\mathcal{U}$ be a set and $S \subseteq \mathcal{U}$. If $s, t \in \mathcal{U}$ and $s \in S$, then we write $S[t/s]$ to denote the set $\bigl(S \setminus \{s\}\bigr) \cup \{t\}$. The cardinality of S is denoted by $\# S$. If M and N are finite multisets over $\mathbb{N}$, $\#(n, M)$ denotes the number of occurrences of $n \in \mathbb{N}$ in M and $M \sqsubseteq_{\mathrm{ms}} N$ means that either $M = N$ or there exists $j \in \mathbb{N}$ such that $\#(j, M) > \#(j, N)$ and, for each $k \in \mathbb{N}$ with $k > j$, $\#(k, M) = \#(k, N)$. The relation '$\sqsubseteq_{\mathrm{ms}}$' is a partial order satisfying the ascending chain condition [25].

The set of non-negative reals is denoted by $\mathbb{R}_+$. Any vector $\vec{v} \in \mathbb{R}^n$ is also a matrix in $\mathbb{R}^{n \times 1}$ so that it can be manipulated with the usual matrix operations of addition and multiplication, both by a scalar and by another matrix. For each $i = 1, \ldots\ n$, the i-th component of a vector $\vec{v} \in \mathbb{R}^n$ is denoted by v_i. The transposition of a matrix M is denoted by M^{T}; thus, for all $\vec{v} \in \mathbb{R}^n$, we have $\vec{v} = (v_1, \ldots, v_n)^{\mathrm{T}}$. The *scalar product* of $\vec{v}, \vec{w} \in \mathbb{R}^n$ is $\langle \vec{v}, \vec{w} \rangle := \sum_{i=1}^{n} v_i w_i$. The vector $\vec{0} \in \mathbb{R}^n$ has all components equal to zero. We write $\vec{v} = \vec{w}$ and $\vec{v} \neq \vec{w}$ to denote the propositions $\bigwedge_{i=1}^{n} (v_i = w_i)$ and $\neg(\vec{v} = \vec{w})$, respectively.

Let $V = \{\vec{v}_1, \ldots, \vec{v}_k\} \subseteq \mathbb{R}^n$ be a finite set of real vectors. For all scalar constants $\lambda_1, \ldots, \lambda_k \in \mathbb{R}$, the vector $\vec{v} = \sum_{i=1}^{k} \lambda_i \vec{v}_i$ is said to be a *linear combination* of the vectors in V. Such a combination is said to be

- a *positive* (or *conic*) combination, if $\lambda_i \in \mathbb{R}_+$ for $i = 1, \ldots, k$;
- an *affine* combination, if $\sum_{i=1}^{k} \lambda_i = 1$;
- a *convex* combination, if it is both positive and affine.

The vectors in V are said to be *linearly independent* if the only solution of the equation

$$\sum_{i=1}^{k} \lambda_i \vec{v}_i = \vec{0}$$

is $\lambda_i = 0$, for each $i = 1, \ldots, k$; they are said to be *affinely independent* if the only solution of the system of equations

$$\begin{cases} \sum_{i=1}^{k} \lambda_i \vec{v}_i = \vec{0}, \\ \quad \sum_{i=1}^{k} \lambda_i = 0 \end{cases}$$

is $\lambda_i = 0$, for each $i = 1, \ldots, k$.

Let $V \subseteq \mathbb{R}^n$. The subspace of $\mathbb{R}^n$ defined by the set of all affine combinations of finite subsets of V is called the *affine hull* of V and denoted by $\mathrm{aff.hull}(V)$; the *orthogonal* of V and the *opposite* of V are given, respectively, by

$$V^{\perp} := \left\{ \vec{w} \in \mathbb{R}^n \;\middle|\; \forall \vec{v} \in V : \langle \vec{v}, \vec{w} \rangle = 0 \right\},$$
$$-V := \left\{ -\vec{v} \in \mathbb{R}^n \;\middle|\; \vec{v} \in V \right\}.$$

For each vector $\vec{a} \in \mathbb{R}^n$ and scalar $b \in \mathbb{R}$, where $\vec{a} \neq \vec{0}$, the linear inequality constraint $\langle \vec{a}, \vec{x} \rangle \geq b$ defines a topologically closed affine half-space of $\mathbb{R}^n$. We do not distinguish between syntactically different constraints defining the same affine half-space so that, for example, $x \geq 2$ and $2x \geq 4$ are the same constraint. The set $\mathcal{P} \subseteq \mathbb{R}^n$ is a (*closed* and *convex*) *polyhedron* if and only if either $\mathcal{P}$ can be expressed as the intersection of a finite number of closed affine half-spaces of $\mathbb{R}^n$, or $n = 0$ and $\mathcal{P} = \varnothing$. The set of all closed polyhedra on $\mathbb{R}^n$ is denoted by $\mathbb{CP}_n$. In this paper, we only consider polyhedra in $\mathbb{CP}_n$ when $n > 0$. The set $\mathbb{CP}_n$, when partially ordered by subset inclusion, is a lattice where the binary meet operation is set-intersection; the binary join operation, denoted '$\uplus$', is called *convex polyhedral hull*, *poly-hull* for short.

If $k \leq n + 1$ is the maximum number of affinely independent points of a polyhedron $\mathcal{P} \in \mathbb{CP}_n$, then the *dimension of* $\mathcal{P}$, denoted as $\dim(\mathcal{P})$, is $k - 1$. If $\mathcal{P} \neq \varnothing$, the *characteristic cone* of $\mathcal{P}$ is defined as

$$\mathrm{char.cone}(\mathcal{P}) := \{ \vec{w} \in \mathbb{R}^n \mid \forall \vec{v} \in \mathcal{P} : \vec{v} + \vec{w} \in \mathcal{P} \},$$

whereas the *lineality space* of $\mathcal{P}$ is

$$\text{lin.space}(\mathcal{P}) := \text{char.cone}(\mathcal{P}) \cap -\text{char.cone}(\mathcal{P}).$$

The linear equality constraint $\langle \vec{a}, \vec{x} \rangle = b$ defines an affine hyperplane of $\mathbb{R}^n$, i.e., the intersection of the affine half-spaces $\langle \vec{a}, \vec{x} \rangle \geq b$ and $\langle -\vec{a}, \vec{x} \rangle \geq -b$. Each polyhedron $\mathcal{P} \in \mathbb{CP}_n$ can therefore be represented by a finite set of linear equality and inequality constraints $\mathcal{C}$ called a *constraint system*. We write $\mathcal{P} = \text{con}(\mathcal{C})$. The subsets of equality and inequality constraints in system $\mathcal{C}$ are denoted by $\text{eq}(\mathcal{C})$ and $\text{ineq}(\mathcal{C})$, respectively. When $\mathcal{P} = \text{con}(\mathcal{C}) \neq \varnothing$, we say that the constraint system $\mathcal{C}$ is in *minimal form* if $\#\,\text{eq}(\mathcal{C}) = n - \dim(\mathcal{P})$ and there does not exist $\mathcal{C}' \subset \mathcal{C}$ such that $\text{con}(\mathcal{C}') = \mathcal{P}$. All the constraint systems in minimal form describing a given polyhedron have the same cardinality. When the constraint system $\mathcal{C}$ is not in minimal form, a constraint $\gamma \in \mathcal{C}$ is said to be *redundant* in $\mathcal{C}$ if $\text{con}\bigl(\mathcal{C} \setminus \{\gamma\}\bigr) = \text{con}(\mathcal{C})$.

Let $\mathcal{P} \in \mathbb{CP}_n$. A vector $\vec{p} \in \mathcal{P}$ is called a *point* of $\mathcal{P}$; a vector $\vec{r} \in \mathbb{R}^n$, where $\vec{r} \neq \vec{0}$, is called a *ray* of $\mathcal{P}$ if $\mathcal{P} \neq \varnothing$ and $\vec{p} + \rho\vec{r} \in \mathcal{P}$, for all points $\vec{p} \in \mathcal{P}$ and all $\rho \in \mathbb{R}_+$; a vector $\vec{l} \in \mathbb{R}^n$ is called a *line* of $\mathcal{P}$ if both $\vec{l}$ and $-\vec{l}$ are rays of $\mathcal{P}$. We do not distinguish between rays (resp., lines) differing by a positive (resp., non-null) factor so that, for example, $(1,3)^{\mathrm{T}}$ and $(2,6)^{\mathrm{T}}$ are the same ray.

Given three finite sets of vectors $L, R, P \subseteq \mathbb{R}^n$ such that $L = \{\vec{l}_1, \dots, \vec{l}_\ell\}$, $R = \{\vec{r}_1, \dots, \vec{r}_r\}$, $P = \{\vec{p}_1, \dots, \vec{p}_p\}$ and $\vec{0} \notin L \cup R$, then the triple $\mathcal{G} = (L, R, P)$ is called a *generator system* for the polyhedron

$$\text{gen}(\mathcal{G}) := \left\{ \sum_{i=1}^{\ell} \lambda_i \vec{l}_i + \sum_{i=1}^{r} \rho_i \vec{r}_i + \sum_{i=1}^{p} \pi_i \vec{p}_i \;\middle|\; \begin{array}{l} \vec{\lambda} \in \mathbb{R}^\ell, \vec{\rho} \in \mathbb{R}^r_+, \vec{\pi} \in \mathbb{R}^p_+, \\ \sum_{i=1}^{p} \pi_i = 1 \end{array} \right\}.$$

The polyhedron $\text{gen}(\mathcal{G})$ is empty if and only if $P = \varnothing$. If $P \neq \varnothing$, the vectors in L, R and P are lines, rays and points of $\text{gen}(\mathcal{G})$, respectively. We define an ordering '$\sqsubseteq_G$' on generator systems such that, for any generator systems $\mathcal{G}_1 = (L_1, R_1, P_1)$ and $\mathcal{G}_2 = (L_2, R_2, P_2)$, $\mathcal{G}_1 \sqsubseteq_G \mathcal{G}_2$ if and only if $L_1 \subseteq L_2$, $R_1 \subseteq R_2$ and $P_1 \subseteq P_2$; if, in addition, $\mathcal{G}_1 \neq \mathcal{G}_2$, we write $\mathcal{G}_1 \sqsubset_G \mathcal{G}_2$. When $\text{gen}(\mathcal{G}) \neq \varnothing$, the generator system $\mathcal{G} = (L, R, P)$ is said to be in *minimal form* if $\#\,L = \dim\bigl(\text{lin.space}(\mathcal{P})\bigr)$ and there does not exist a generator system $\mathcal{G}' \sqsubset_G \mathcal{G}$ such that $\text{gen}(\mathcal{G}') = \text{gen}(\mathcal{G})$. All the generator systems in minimal form describing a given polyhedron have the same cardinalities for the lines, rays and points components.

The possibility of representing a convex polyhedron by means of both constraint and generator systems is the basis of the *double description* method [26], which exploits the duality principle to compute each representation starting from the other one, possibly minimizing both descriptions. Clever implementations of this *conversion* procedure, such as those based on the extension by Le Verge [27] of the Chernikova's algorithms [28–30], are the starting point for the development of software libraries based on the double description method. [1]

Let $\beta = \bigl(\langle \vec{a}, \vec{x} \rangle \bowtie b\bigr)$ be a linear constraint, where $\bowtie \in \{\geq, =\}$. We say that a point (resp., a ray or a line) $\vec{v}$ *saturates* constraint β if and only if $\langle \vec{a}, \vec{v} \rangle = b$ (resp., $\langle \vec{a}, \vec{v} \rangle = 0$). For each point $\vec{p}$ and constraint system $\mathcal{C}$, we define the constraint system

$$\mathrm{sat\_con}(\vec{p}, \mathcal{C}) := \{\, \beta \in \mathcal{C} \mid \vec{p} \text{ saturates } \beta \,\};$$

for each constraint β and generator system $\mathcal{G} = (L, R, P)$, we define the generator system $\mathrm{sat\_gen}(\beta, \mathcal{G}) := (L', R', P')$, where

$$\begin{aligned} L' &:= \{\, \vec{l} \in L \mid \vec{l} \text{ saturates } \beta \,\}, \\ R' &:= \{\, \vec{r} \in R \mid \vec{r} \text{ saturates } \beta \,\}, \\ P' &:= \{\, \vec{p} \in P \mid \vec{p} \text{ saturates } \beta \,\}. \end{aligned}$$

A generator system $\mathcal{G} = (L, R, P)$ is in *orthogonal form* if it is in minimal form and $R \cup P \subseteq L^{\perp}$. All generator systems in orthogonal form describing a given polyhedron have identical sets of rays and points. A generator system in minimal form can be transformed into an equivalent system in orthogonal form by means of the well-known Gram-Shmidt method. By duality, orthogonal forms can also be defined for constraint systems. For each linear constraint $\beta = \bigl(\langle \vec{a}, \vec{x} \rangle \bowtie b\bigr)$, where $\bowtie \in \{\geq, =\}$, let $\mathrm{slope}(\beta) := \vec{a}$. A constraint system $\mathcal{C}$ is in *orthogonal form* if it is in minimal form and we have $I \subseteq E^{\perp}$, where

$$\begin{aligned} I &:= \bigl\{\, \mathrm{slope}(\beta) \bigm| \beta \in \mathrm{ineq}(\mathcal{C}) \,\bigr\}, \\ E &:= \bigl\{\, \mathrm{slope}(\beta) \bigm| \beta \in \mathrm{eq}(\mathcal{C}) \,\bigr\}. \end{aligned}$$

All constraint systems in orthogonal form describing a given polyhedron have identical sets of inequality constraints.

3 The Standard Widening

The first widening on convex polyhedra was introduced in [1]. Intuitively, if $\mathcal{P}_1$ is the polyhedron obtained in the previous step of the upward iteration

[1] These libraries include: Polylib, designed and written by H. Le Verge and D. K. Wilde [27,31]; *PolyLib*, the successor of the library by Le Verge and Wilde [32]; New Polka, by B. Jeannet [33]; the polyhedra library that comes with the HyTech tool [6]; the *Parma Polyhedra Library* [34,35].

sequence and the current step yields polyhedron $\mathcal{P}_2$, then the widening of $\mathcal{P}_2$ with respect to $\mathcal{P}_1$ is the polyhedron defined by all the constraints of $\mathcal{P}_1$ that are satisfied by all the points of $\mathcal{P}_2$. An improvement on the above idea was defined in [17]. This operator, termed *standard widening*, has indeed been used almost universally.

The formal specification of the standard widening requires that each equality constraint is split into the two corresponding linear inequalities; thus, for each constraint system $\mathcal{C}$, we define

$$\begin{aligned}\mathrm{repr}_{\geq}(\mathcal{C}) := \Bigl\{\, \langle -\vec{a}, \vec{x}\rangle \geq -b \Bigm| \bigl(\langle \vec{a}, \vec{x}\rangle = b\bigr) \in \mathcal{C} \,\Bigr\} \\ \cup \Bigl\{\, \langle \vec{a}, \vec{x}\rangle \geq b \Bigm| \bigl(\langle \vec{a}, \vec{x}\rangle \bowtie b\bigr) \in \mathcal{C}, \bowtie \in \{\geq, =\} \,\Bigr\}.\end{aligned}$$

Definition 1 (Standard widening on $\mathbb{CP}_n$.) [17, Définition 5.3.3, p. 57] *For $i = 1, 2$, let $\mathcal{P}_i \in \mathbb{CP}_n$ be such that $\mathcal{P}_i = \mathrm{con}(\mathcal{C}_i)$ and $\mathcal{I}_i = \mathrm{repr}_{\geq}(\mathcal{C}_i)$* [and let $\mathcal{C}_1$ be either inconsistent or in minimal form]. *Then the polyhedron $\mathcal{P}_1 \nabla_S \mathcal{P}_2 \in \mathbb{CP}_n$ is defined as*

$$\mathcal{P}_1 \nabla_S \mathcal{P}_2 := \begin{cases} \mathcal{P}_2, & \text{if } \mathcal{P}_1 = \varnothing; \\ \mathrm{con}(\mathcal{I}_1' \cup \mathcal{I}_2'), & \text{otherwise;} \end{cases}$$

where

$$\begin{aligned}\mathcal{I}_1' &:= \Bigl\{\, \beta \in \mathcal{I}_1 \Bigm| \mathcal{P}_2 \subseteq \mathrm{con}\bigl(\{\beta\}\bigr) \,\Bigr\}, \\ \mathcal{I}_2' &:= \Bigl\{\, \gamma \in \mathcal{I}_2 \Bigm| \exists \beta \in \mathcal{I}_1 \,.\, \mathcal{P}_1 = \mathrm{con}\bigl(\mathcal{I}_1[\gamma/\beta]\bigr) \,\Bigr\}.\end{aligned}$$

The constraints in $\mathcal{I}_1'$ are those that would have been selected when using the original proposal of [1], whereas the constraints in $\mathcal{I}_2'$ are added to ensure that this widening is a well-defined operator on the domain of polyhedra (i.e., it does not depend on the particular constraint representations).

Note that, in Definition 1, the condition in square brackets was implicit from the context of [17, Définition 5.3.3, p. 57], though not explicitly present in the definition itself. Such a requirement has been sometimes neglected in later papers discussing the standard widening (and also in some implementations), but it is actually needed in order to obtain a correct definition. In fact, the following two examples show that if a non-minimal constraint description is taken into account, then not only is the widening operator not well defined (see Example 2), but also the chain condition may be violated (see Example 3).

Example 2 *For $i = 1, 2$, let $\mathcal{P}_i = \mathrm{con}(\mathcal{C}_i) \in \mathbb{CP}_2$, where*

$$\begin{aligned} \mathcal{C}_1 &= \{x \geq 0, y \geq 0, x - y \geq 2\}, \\ \mathcal{C}_2 &= \{x \geq 2, y \geq 0\}. \end{aligned}$$

Note that the constraint $x \geq 0$ is redundant in $\mathcal{C}_1$. By applying [17, Définition 5.3.3, p. 57] *verbatim, without enforcing minimization, we would obtain the polyhedron*

$$\mathcal{P} = \mathrm{con}\Bigl(\{x \geq 0, y \geq 0\}\Bigr).$$

In contrast, by applying Definition 1, *i.e., by enforcing minimization, we obtain the polyhedron*

$$\mathcal{P}' = \mathrm{con}\Bigl(\{y \geq 0\}\Bigr).$$

Example 3 *Consider, for each $k \in \mathbb{N}$, the polyhedron $\mathcal{P}_k := \mathrm{con}(\mathcal{C}_k) \in \mathbb{CP}_1$, where*

$$\mathcal{C}_k := \left\{0 \leq x, x \leq \frac{k}{k+1}\right\} \cup \{x \leq 2\},$$

and note that no $\mathcal{C}_k$ is minimal since the constraint $x \leq 2$ is redundant in all of them. Moreover, the infinite chain constituted by the $\mathcal{P}_k$'s, that is, using an interval notation,

$$\mathcal{P}_0 = [0, 0],\ \mathcal{P}_1 = \left[0, \frac{1}{2}\right],\ \mathcal{P}_2 = \left[0, \frac{2}{3}\right],\ \mathcal{P}_3 = \left[0, \frac{3}{4}\right],\ \ldots,$$

is strictly increasing. We will now show that, if we do not enforce minimization in the computation of the standard widening '∇_s', then for the infinite chain $\mathcal{Q}_0 = \mathcal{P}_0, \ldots, \mathcal{Q}_{k+1} = \mathcal{Q}_k \nabla_s \mathcal{P}_{k+1}, \ldots$ we have $\mathcal{Q}_n = \mathcal{P}_n$ for each $n \in \mathbb{N}$, so that the chain condition is violated.

For each $n \in \mathbb{N}$ we have $\mathcal{Q}_n = \mathrm{con}(\mathcal{D}_n)$, where $\mathcal{D}_0 := \mathcal{C}_0$ and

$$\begin{aligned} \mathcal{D}_{k+1} := \Bigl\{ \beta \in \mathcal{D}_k \Bigm| \mathcal{P}_{k+1} \subseteq \mathrm{con}\bigl(\{\beta\}\bigr) \Bigr\} \\ \cup \Bigl\{ \gamma \in \mathcal{C}_{k+1} \Bigm| \exists \beta \in \mathcal{D}_k \mathrel{.} \mathcal{Q}_k = \mathrm{con}\bigl(\mathcal{D}_k[\gamma/\beta]\bigr) \Bigr\}. \end{aligned}$$

We will show by induction that $\mathcal{D}_n = \mathcal{C}_n$ for each $n \in \mathbb{N}$. First we note that $\{0 \leq x, x \leq 2\} \subseteq \mathcal{D}_0 = \mathcal{C}_0$ and thus $\{0 \leq x, x \leq 2\} \subseteq \mathcal{D}_k$ for each $k \in \mathbb{N}$, since $\mathcal{P}_{k+1} \subseteq \mathrm{con}\bigl(\{0 \leq x\}\bigr)$ and $\mathcal{P}_{k+1} \subseteq \mathrm{con}\bigl(\{x \leq 2\}\bigr)$. Now assume $\mathcal{D}_k = \mathcal{C}_k$

and take $\beta = (x \leq 2) \in \mathcal{D}_k$ *and* $\gamma = \bigl(x \leq \frac{k+1}{k+2}\bigr) \in \mathcal{C}_{k+1}$*, so that*

$$\begin{aligned}
\operatorname{con}\bigl(\mathcal{D}_k[\gamma/\beta]\bigr) &= \operatorname{con}\left(\left\{0 \leq x, x \leq \frac{k}{k+1}, x \leq \frac{k+1}{k+2}\right\}\right) \\
&= \operatorname{con}\left(\left\{0 \leq x, x \leq \frac{k}{k+1}\right\}\right) \\
&= \operatorname{con}(\mathcal{D}_k).
\end{aligned}$$

We thus have $\mathcal{D}_{k+1} = \bigl\{0 \leq x, x \leq \frac{k+1}{k+2}, x \leq 2\bigr\} = \mathcal{C}_{k+1}$.

3.1 Implementation of the Standard Widening

The proposition below provides an algorithm for computing the standard widening of the pair of polyhedra $\mathcal{P}_1$ and $\mathcal{P}_2$ when $\mathcal{P}_1 \subseteq \mathcal{P}_2$. The idea, which was proposed in [17] and later reported in [23], is to replace the expensive test in the specification of $\mathcal{I}'_2$ in Definition 1 with an appropriate saturation condition to be checked on any generator system for $\mathcal{P}_1$. This is worthwhile in all implementations based on the double description method. The algorithm here is an improved version over these proposals since neither the addition of the set of constraints $\mathcal{I}'_1$ as given in Definition 1 nor the splitting of equality constraints into pairs of inequalities is required. A similar result, but without the use of saturation conditions, can be found in [9, Chapter 6].

Proposition 4 *Let* $\mathcal{P}_1 = \operatorname{con}(\mathcal{C}_1) = \operatorname{gen}(\mathcal{G}_1) \in \mathbb{CP}_n$ *and* $\mathcal{P}_2 = \operatorname{con}(\mathcal{C}_2) \in \mathbb{CP}_n$*, where* $\mathcal{P}_1 \subseteq \mathcal{P}_2$ *and* $\mathcal{C}_1$ *is either inconsistent or in minimal form. Then*

$$\mathcal{P}_1 \nabla_S \mathcal{P}_2 = \begin{cases} \mathcal{P}_2, & \text{if } \mathcal{P}_1 = \varnothing; \\ \operatorname{con}(\mathcal{C}_S), & \text{otherwise;} \end{cases}$$

where $\mathcal{C}_S := \bigl\{\, \gamma \in \mathcal{C}_2 \bigm| \exists \beta \in \mathcal{C}_1 \,.\, \operatorname{sat\_gen}(\gamma, \mathcal{G}_1) = \operatorname{sat\_gen}(\beta, \mathcal{G}_1) \,\bigr\}$.

PROOF. The result holds trivially when $\mathcal{P}_1 = \varnothing$. Therefore, we assume that $\mathcal{P}_1 \neq \varnothing$, so that by hypothesis $\mathcal{C}_1$ is in minimal form, and prove that $\mathcal{P}_1 \nabla_S \mathcal{P}_2 = \operatorname{con}(\mathcal{C}_S)$ by considering the two inclusions separately.

Assume the notation introduced in Definition 1 for the constraint systems $\mathcal{I}_1$, $\mathcal{I}_2$ and $\mathcal{I}'_1$, $\mathcal{I}'_2$, so that $\mathcal{P}_1 \nabla_S \mathcal{P}_2 = \operatorname{con}(\mathcal{I}'_1 \cup \mathcal{I}'_2)$. Let also $\mathcal{I}_S = \operatorname{repr}_{\geq}(\mathcal{C}_S)$, so that $\operatorname{con}(\mathcal{C}_S) = \operatorname{con}(\mathcal{I}_S)$ and

$$\mathcal{I}_S := \bigl\{\, \gamma \in \mathcal{I}_2 \bigm| \exists \beta \in \mathcal{I}_1 \,.\, \operatorname{sat\_gen}(\gamma, \mathcal{G}_1) = \operatorname{sat\_gen}(\beta, \mathcal{G}_1) \,\bigr\}.$$

First we prove $\mathcal{P}_1 \nabla_S \mathcal{P}_2 \subseteq \mathrm{con}(\mathcal{C}_S)$ by showing that $\mathcal{I}_S \subseteq \mathcal{I}_2'$. Suppose, for some $\vec{a} \in \mathbb{R}^n$ and $b \in \mathbb{R}$, $\gamma := \bigl(\langle \vec{a}, \vec{x} \rangle \geq b\bigr) \in \mathcal{I}_S$. By definition of $\mathcal{C}_S$, we have $\gamma \in \mathcal{I}_2$. We will show that there exists $\beta \in \mathcal{I}_1$ such that

$$\mathcal{P}_1 = \mathrm{con}\Bigl(\mathcal{I}_1[\gamma/\beta]\Bigr) \tag{1}$$

so that, by Definition 1, $\gamma \in \mathcal{I}_2'$ as required. First note that, as $\gamma \in \mathcal{I}_S$, by the hypothesis, we can assume that there exists $\beta' \in \mathcal{C}_1$ such that

$$\mathrm{sat\_gen}(\gamma, \mathcal{G}_1) = \mathrm{sat\_gen}(\beta', \mathcal{G}_1). \tag{2}$$

We consider the two subcases $\beta' \in \mathrm{eq}(\mathcal{C}_1)$ and $\beta' \in \mathrm{ineq}(\mathcal{C}_1)$ separately. Suppose first that $\beta' \in \mathrm{eq}(\mathcal{C}_1)$. Then $\mathrm{sat\_gen}(\beta', \mathcal{G}_1) = \mathcal{G}_1$ so that, by (2), we have $\mathrm{sat\_gen}(\gamma, \mathcal{G}_1) = \mathcal{G}_1$. Let $\gamma' := \bigl(\langle \vec{a}, \vec{x} \rangle = b\bigr)$ be the equality constraint corresponding to the inequality γ, so that we obtain $\mathrm{sat\_gen}(\gamma', \mathcal{G}_1) = \mathcal{G}_1$. Thus, γ' is a valid equality for polyhedron $\mathcal{P}_1$. By hypothesis, $\mathcal{C}_1$ is in minimal form, so that γ' (and hence, also γ) can be expressed as a linear combination of some of the constraints in $\mathrm{eq}(\mathcal{C}_1)$. Namely, there exist $k > 0$ equality constraints $\{\gamma_1, \dots, \gamma_k\} \subseteq \mathrm{eq}(\mathcal{C}_1)$ such that both $\vec{a} = \sum_{i=1}^{k} \lambda_i \vec{a}_i$ and $b = \sum_{i=1}^{k} \lambda_i b_i$ hold where, for $1 \leq i \leq k$, $\lambda_i \in \mathbb{R} \setminus \{0\}$ and $\gamma_i := \bigl(\langle \vec{a}_i, \vec{x} \rangle = b_i\bigr)$. Thus

$$\lambda_1 \vec{a}_1 = \vec{a} - \sum_{i=2}^{k} \lambda_i \vec{a}_i; \qquad\qquad \lambda_1 b_1 = b - \sum_{i=2}^{k} \lambda_i b_i. \tag{3}$$

For $1 \leq i \leq k$, let $\gamma_i^+ := \bigl(\langle \vec{a}_i, \vec{x} \rangle \geq b_i\bigr)$ and $\gamma_i^- := \bigl(\langle -\vec{a}_i, \vec{x} \rangle \geq -b_i\bigr)$; moreover, take

$$\beta_1 := \begin{cases} \gamma_1^+, & \text{if } \lambda_1 > 0; \\ \gamma_1^-, & \text{if } \lambda_1 < 0; \end{cases}$$

and, for $2 \leq i \leq k$, let

$$\beta_i := \begin{cases} \gamma_i^+, & \text{if } \lambda_i < 0; \\ \gamma_i^-, & \text{if } \lambda_i > 0. \end{cases}$$

Note that, by definition of '$\mathrm{repr}_{\geq}$', we have $\{\beta_1, \beta_2, \dots, \beta_k\} \subseteq \mathcal{I}_1$. Then, the two equations (3) show that the inequality constraint $\beta_1 \in \mathcal{I}_1$ can be computed as a positive combination of the inequality constraint $\gamma \in \mathcal{I}_2'$ and the inequality constraints $\{\beta_2, \dots, \beta_k\} \subseteq \mathcal{I}_1$. Therefore, (1) holds by letting $\beta = \beta_1$.

For the second subcase, suppose $\beta' \in \mathrm{ineq}(\mathcal{C}_1)$, so that $\beta' \in \mathcal{I}_1$. As $\mathcal{C}_1$ is in minimal form, $\mathrm{sat\_gen}(\beta', \mathcal{G}_1) \neq \mathcal{G}_1$. Informally, β' can be seen as identifying one of the facets of $\mathcal{P}_1$. Since (2) holds, the constraint γ identifies the same facet of $\mathcal{P}_1$; since $\mathcal{P}_1 \subseteq \mathcal{P}_2 \subseteq \mathrm{con}\bigl(\{\gamma\}\bigr)$, it is also a valid constraint for $\mathcal{P}_1$, so that $\mathcal{P}_1 = \mathrm{con}\Bigl(\mathcal{I}_1[\gamma/\beta']\Bigr)$. Therefore, (1) holds by letting $\beta = \beta'$.

Secondly we prove $\mathrm{con}(\mathcal{C}_S) \subseteq \mathcal{P}_1 \nabla_S \mathcal{P}_2$; as $\mathcal{P}_1 \nabla_S \mathcal{P}_2 = \mathrm{con}(\mathcal{I}'_1 \cup \mathcal{I}'_2)$, we have to prove

$$\mathrm{con}(\mathcal{C}_S) \subseteq \mathrm{con}(\mathcal{I}'_1), \tag{4}$$
$$\mathrm{con}(\mathcal{C}_S) \subseteq \mathrm{con}(\mathcal{I}'_2). \tag{5}$$

To prove (4), we first show

$$\mathrm{con}(\mathcal{C}_S) \subseteq \mathrm{aff.hull}(\mathcal{P}_2) \subseteq \mathrm{aff.hull}\Big(\mathrm{con}(\mathcal{I}'_1)\Big). \tag{6}$$

Suppose $\gamma \in \mathcal{C}_2$ is a constraint defining the affine hull of $\mathcal{P}_2$, so that it is saturated by all the points of $\mathcal{P}_2$. Since $\mathcal{P}_1 \subseteq \mathcal{P}_2$, γ is also saturated by all the points of $\mathcal{P}_1$. Hence, there exists $\beta \in \mathcal{C}_1$ such that $\mathrm{sat\_gen}(\beta, \mathcal{G}_1) = \mathcal{G}_1 = \mathrm{sat\_gen}(\gamma, \mathcal{G}_1)$. Thus, by definition of $\mathcal{C}_S$, we have $\gamma \in \mathcal{C}_S$. As this holds for all the constraints defining the affine hull of $\mathcal{P}_2$, $\mathrm{aff.hull}\Big(\mathrm{con}(\mathcal{C}_S)\Big) \subseteq \mathrm{aff.hull}(\mathcal{P}_2)$. Since $\mathcal{P}_2 \subseteq \mathcal{P}_1 \nabla_S \mathcal{P}_2 \subseteq \mathrm{con}(\mathcal{I}'_1)$, we have $\mathrm{aff.hull}(\mathcal{P}_2) \subseteq \mathrm{aff.hull}\Big(\mathrm{con}(\mathcal{I}'_1)\Big)$ and hence, as $\mathrm{con}(\mathcal{C}_S) \subseteq \mathrm{aff.hull}\Big(\mathrm{con}(\mathcal{C}_S)\Big)$, (6) holds.

We next show that, if β is any constraint in $\mathcal{I}'_1$, then $\mathrm{con}(\mathcal{C}_S) \subseteq \mathrm{con}\Big(\{\beta\}\Big)$. If β is a constraint defining the affine hull of $\mathrm{con}(\mathcal{I}'_1)$, then this follows from (6). Suppose next that β is not a constraint defining the affine hull of $\mathrm{con}(\mathcal{I}'_1)$. By Definition 1, $\mathcal{I}'_1 \subseteq \mathcal{I}_1$; so that, as $\beta \in \mathcal{I}'_1$ and $\mathcal{C}_1$ is in minimal form, β defines a facet of $\mathrm{con}(\mathcal{I}'_1)$ and $\beta \in \mathcal{I}_1$. By hypothesis, $\mathcal{P}_1 \subseteq \mathcal{P}_2$ and, by Definition 1, $\mathcal{P}_2 \subseteq \mathrm{con}\Big(\{\beta\}\Big)$; therefore there exists a constraint $\gamma \in \mathcal{I}_2$ that is saturated by the same points in $\mathcal{P}_2$ that saturate β. Hence $\mathrm{sat\_gen}(\beta, \mathcal{G}_1) = \mathrm{sat\_gen}(\gamma, \mathcal{G}_1)$ so that $\gamma \in \mathcal{I}_S$ and $\mathrm{con}(\mathcal{C}_S) = \mathrm{con}(\mathcal{I}_S) \subseteq \mathrm{con}\Big(\{\gamma\}\Big)$; moreover, we also obtain

$$\mathrm{con}\Big(\{\gamma\}\Big) \cap \mathrm{aff.hull}(\mathcal{P}_2) = \mathrm{con}\Big(\{\beta\}\Big) \cap \mathrm{aff.hull}(\mathcal{P}_2) \subseteq \mathrm{con}\Big(\{\beta\}\Big)$$

so that, by (6), we have $\mathrm{con}(\mathcal{C}_S) = \mathrm{con}\Big(\{\gamma\}\Big) \cap \mathrm{con}(\mathcal{C}_S) \subseteq \mathrm{con}\Big(\{\beta\}\Big)$. Therefore $\mathrm{con}(\mathcal{C}_S) \subseteq \mathrm{con}\Big(\{\beta\}\Big)$ for all $\beta \in \mathcal{I}'_1$; hence (4) holds.

We now show that (5) holds. Suppose $\gamma \in \mathcal{I}'_2$ so that, by Definition 1, $\gamma \in \mathcal{I}_2$ and there exists $\beta \in \mathcal{I}_1$ such that $\mathcal{P}_1 = \mathrm{con}\Big(\mathcal{I}_1[\gamma/\beta]\Big)$. As $\mathcal{C}_1$ is in minimal form, $\mathrm{sat\_gen}(\gamma, \mathcal{G}_1) = \mathrm{sat\_gen}(\beta, \mathcal{G}_1)$ so that $\gamma \in \mathcal{I}_S$; and hence $\mathrm{con}(\mathcal{C}_S) \subseteq \mathrm{con}\Big(\{\gamma\}\Big)$. As the choice of $\gamma \in \mathcal{I}'_2$ was arbitrary, (5) holds. $\square$

The next example shows that the inclusion hypothesis $\mathcal{P}_1 \subseteq \mathcal{P}_2$ in Proposition 4, which is only implicitly present in [17,23], is vital in guaranteeing that the algorithm computes an upper approximation of $\mathcal{P}_1$ and $\mathcal{P}_2$. Note that this is independent of the two improvements mentioned above.

Example 5 *Let* $\mathcal{P}_1 := \operatorname{con}(\mathcal{C}_1) \in \mathbb{CP}_2$ *and* $\mathcal{P}_2 := \operatorname{con}(\mathcal{C}_2) \in \mathbb{CP}_2$*, where*

$$\begin{aligned}\mathcal{C}_1 &:= \{x = 0, 0 \leq y \leq 2\},\\ \mathcal{C}_2 &:= \{y \geq 2\}.\end{aligned}$$

Then $\mathcal{P}_1 = \operatorname{gen}(\mathcal{G}_1)$*, where* $\mathcal{G}_1 = (\varnothing, \varnothing, P)$ *and* $P = \bigl\{(0,0)^{\mathrm{T}}, (2,0)^{\mathrm{T}}\bigr\}$*. Note that* $\mathcal{P}_1 \nsubseteq \mathcal{P}_2$*. By* Definition 1*, we obtain* $\mathcal{I}'_1 = \mathcal{I}'_2 = \varnothing$*, so that* $\mathcal{P}_1 \nabla_S \mathcal{P}_2 = \mathbb{R}^2$*. Considering the constraints* $\beta = (-y \geq -2) \in \mathcal{C}_1$ *and* $\gamma = (y \geq 2) \in \mathcal{C}_2$*, we have*

$$\operatorname{sat\_gen}(\beta, \mathcal{G}_1) = \Bigl(\varnothing, \varnothing, \bigl\{(2,0)^{\mathrm{T}}\bigr\}\Bigr) = \operatorname{sat\_gen}(\gamma, \mathcal{G}_1),$$

so that $\gamma \in \mathcal{C}_S$*. Thus, the result of the algorithm specified by* Proposition 4 *would be* $\mathcal{P}_2$*, which is different from* $\mathcal{P}_1 \nabla_S \mathcal{P}_2$ *and, moreover, is not an upper approximation of* $\mathcal{P}_1$*.*

As far as the implementation of the standard widening is concerned, it is worth noting the following result, which provides the justification for an alternative algorithm based on the original proposal in [1]. A similar result has also been proved in [9, Chapter 6].

Proposition 6 *Let* $\mathcal{P}_1, \mathcal{P}_2 \in \mathbb{CP}_n$*, where* $\mathcal{P}_1 \subseteq \mathcal{P}_2$ *and* $\dim(\mathcal{P}_1) = \dim(\mathcal{P}_2)$*. Let also* $\mathcal{P}_1 = \operatorname{con}(\mathcal{C}_1)$*, where the constraint system* $\mathcal{C}_1$ *is either inconsistent or in minimal form. Then*

$$\mathcal{P}_1 \nabla_S \mathcal{P}_2 = \begin{cases}\mathcal{P}_2, & \text{if } \mathcal{P}_1 = \varnothing;\\ \operatorname{con}(\mathcal{C}_d), & \text{otherwise,}\end{cases}$$

where $\mathcal{C}_d := \Bigl\{\, \beta \in \mathcal{C}_1 \Bigm| \mathcal{P}_2 \subseteq \operatorname{con}\bigl(\{\beta\}\bigr) \,\Bigr\}$.

PROOF. The result trivially holds if $\mathcal{C}_1$ is inconsistent. Thus, in the rest of the proof, we assume that $\mathcal{C}_1$ is consistent and in minimal form.

Let $\mathcal{I}_1 = \operatorname{repr}_{\geq}(\mathcal{C}_1)$ and $\mathcal{I}_d = \operatorname{repr}_{\geq}(\mathcal{C}_d)$; let also $\mathcal{I}'_1$ be as given in Definition 1. Since $\mathcal{I}_d \subseteq \mathcal{I}'_1$, we have $\mathcal{P}_1 \nabla_S \mathcal{P}_2 \subseteq \operatorname{con}(\mathcal{I}_d) = \operatorname{con}(\mathcal{C}_d)$. Thus, to prove that $\mathcal{P}_1 \nabla_S \mathcal{P}_2 = \operatorname{con}(\mathcal{C}_d)$, we show that $\operatorname{con}(\mathcal{C}_d) \subseteq \mathcal{P}_1 \nabla_S \mathcal{P}_2$.

As $\dim(\mathcal{P}_1) = \dim(\mathcal{P}_2)$ and $\mathcal{P}_1 \subseteq \mathcal{P}_2$, we also have $\operatorname{aff.hull}(\mathcal{P}_1) = \operatorname{aff.hull}(\mathcal{P}_2)$. Thus, there exists a constraint system $\mathcal{C}_2$ which is in minimal form and such that $\mathcal{P}_2 = \operatorname{con}(\mathcal{C}_2)$ and $\operatorname{eq}(\mathcal{C}_1) = \operatorname{eq}(\mathcal{C}_2)$.

Let $\mathcal{P}_1 = \operatorname{gen}(\mathcal{G}_1)$ for some generator system $\mathcal{G}_1$ and

$$\mathcal{C}_S := \Bigl\{\, \gamma \in \mathcal{C}_2 \Bigm| \exists \beta \in \mathcal{C}_1 \mathrel{.} \operatorname{sat\_gen}(\gamma, \mathcal{G}_1) = \operatorname{sat\_gen}(\beta, \mathcal{G}_1) \,\Bigr\}.$$

By Proposition 4, $\mathcal{P}_1 \nabla_S \mathcal{P}_2 = \operatorname{con}(\mathcal{C}_S)$. Therefore, it remains for us to show that $\operatorname{con}(\mathcal{C}_d) \subseteq \operatorname{con}(\mathcal{C}_S)$. Suppose $\gamma \in \mathcal{C}_S$. Then, by definition of $\mathcal{C}_S$, there exists

$\beta \in \mathcal{C}_1$ such that

$$\operatorname{con}\bigl(\{\beta\}\bigr) \cap \operatorname{aff.hull}(\mathcal{P}_1) = \operatorname{con}\bigl(\{\gamma\}\bigr) \cap \operatorname{aff.hull}(\mathcal{P}_1). \tag{7}$$

As $\mathcal{P}_2 \subseteq \operatorname{con}\bigl(\{\gamma\}\bigr)$ and $\mathcal{P}_2 \subseteq \operatorname{aff.hull}(\mathcal{P}_1)$, we obtain $\mathcal{P}_2 \subseteq \operatorname{con}\bigl(\{\beta\}\bigr)$, so that we also have $\beta \in \mathcal{C}_d$. For any $\beta' \in \operatorname{eq}(\mathcal{C}_1)$, $\operatorname{aff.hull}(\mathcal{P}_1) \subseteq \operatorname{con}\bigl(\{\beta'\}\bigr)$ so that, since $\operatorname{aff.hull}(\mathcal{P}_1) = \operatorname{aff.hull}(\mathcal{P}_2)$, $\mathcal{P}_2 \subseteq \operatorname{con}\bigl(\{\beta'\}\bigr)$; and hence, by definition of $\mathcal{C}_d$, $\beta' \in \mathcal{C}_d$ so that $\operatorname{aff.hull}(\mathcal{P}_1) = \operatorname{aff.hull}\bigl(\operatorname{con}(\mathcal{C}_d)\bigr)$. Therefore it follows from (7) that $\operatorname{con}(\mathcal{C}_d) \subseteq \operatorname{con}\bigl(\{\gamma\}\bigr)$. As this holds for all $\gamma \in \mathcal{C}_s$, we obtain $\operatorname{con}(\mathcal{C}_d) \subseteq \operatorname{con}(\mathcal{C}_s)$. □

The interesting fact about an algorithm based on Proposition 6 is that, in most cases, the computation of a constraint system for the polyhedron $\mathcal{P}_2$ can be avoided, because any generator system for $\mathcal{P}_2$ can be used to efficiently check if $\dim(\mathcal{P}_1) = \dim(\mathcal{P}_2)$ and, if so, to select the constraints from $\mathcal{C}_1$; only if $\dim(\mathcal{P}_1) \neq \dim(\mathcal{P}_2)$ do we have to fall back to an implementation based on Proposition 4. Note that it is almost always the case that polyhedron $\mathcal{P}_2$ has been obtained as the result of a poly-hull operation so that, in a "lazy" implementation based on the double description method, the polyhedron will be described by a generator system only (since, in such implementations, the poly-hull is computed by taking the union of the generator systems of the arguments).

4 A Framework for Improving Upon a Widening

In this section, generalising an idea originally proposed in [3], we present a framework for the systematic definition of new and precise widening operators improving upon an existing widening.

Since a generic widening operator is a partial function, our framework has to make some assumptions about its domain of definition, so as to ensure that any call to this operator is well defined. For this reason, in the following we adopt a minor variation of the classical definition of the widening operator given in Section 1 (see the footnote in [15, p. 275]).

Definition 7 (Widening.) *Let $\langle L, \perp, \sqsubseteq, \sqcup \rangle$ be a join-semi-lattice (i.e., the least upper bound $x \sqcup y$ exists for all $x, y \in L$). The operator $\nabla \colon L \times L \rightarrowtail L$ is a* widening *if*

(1) for all $x, y \in L$, $x \sqsubseteq y$ implies that $x \nabla y$ is defined and $y \sqsubseteq x \nabla y$;

(2) for all increasing chains $y_0 \sqsubseteq y_1 \sqsubseteq \cdots$, the increasing chain defined by $x_0 := y_0$ and $x_{i+1} := x_i \mathbin{\nabla} (x_i \sqcup y_{i+1})$, for $i \in \mathbb{N}$, is not strictly increasing.

It can be proved that, for any monotonic operator $\mathcal{F} \colon L \to L$, the upward iteration sequence with widenings starting at the bottom element $x_0 := \bot$ and defined by

$$x_{i+1} := \begin{cases} x_i, & \text{if } \mathcal{F}(x_i) \sqsubseteq x_i; \\ x_i \mathbin{\nabla} \Bigl(x_i \sqcup \mathcal{F}(x_i)\Bigr), & \text{otherwise;} \end{cases}$$

converges to a post-fixpoint of '$\mathcal{F}$' after a finite number of iterations [15]. Note that the widening is always applied to arguments $x = x_i$ and $y = x_i \sqcup \mathcal{F}(x_i)$ satisfying $x \sqsubset y$. Therefore, problems such as the one outlined in Example 5 will be automatically avoided.

The framework is based on a class of preorders formalizing a notion of, so to speak, "guaranteed limited growth."

Definition 8 (∇-compatible limited growth ordering.) *Let '∇' be a widening operator on the join-semi-lattice $\langle L, \bot, \sqsubseteq, \sqcup \rangle$. A* limited growth ordering *(*lgo*, for short) is the strict version of any finitely computable preorder on L that satisfies the ascending chain condition. A* ∇-compatible lgo *$\curvearrowright \subseteq L \times L$ is a limited growth ordering such that*

$$\forall x, y \in L : x \sqsubset y \implies x \curvearrowright x \mathbin{\nabla} y.$$

The computability requirement is important because we will directly use the lgo relation to provide an executable specification of the new widenings. The ∇-compatibility requirement ensures that, in the definition of the new widening, we can use the widening '∇' as a last resort operator without compromising the convergence guarantee. As a matter of fact, even the finite convergence guarantee for the widening '∇' is a direct consequence of the ∇-compatibility requirement for the lgo relation.

The next result shows how a ∇-compatible lgo simplifies the definition of a new widening that improves on '∇'.

Theorem 9 *Let '∇' be a widening on the join-semi-lattice $\langle L, \bot, \sqsubseteq, \sqcup \rangle$. Suppose that $\curvearrowright \subseteq L \times L$ is a ∇-compatible lgo and $h \colon L \times L \to L$ is an upper bound operator. For all $x, y \in L$ such that $x \sqsubseteq y$, let*

$$x \mathbin{\tilde{\nabla}} y := \begin{cases} h(x, y), & \text{if } x \curvearrowright h(x, y) \sqsubset x \mathbin{\nabla} y; \\ x \mathbin{\nabla} y, & \text{otherwise.} \end{cases}$$

Then the '$\tilde{\nabla}$' operator is a widening at least as precise as '∇'.

PROOF. By hypothesis, 'h' is an upper bound operator and, by Definition 7, the same holds for the widening '∇'. Thus, in all cases we have $y \sqsubseteq x \mathbin{\tilde{\nabla}} y$, so that the first condition in Definition 7 holds. Note that, in both the cases of the definition of '$\tilde{\nabla}$', we have $x \curvearrowright x \mathbin{\tilde{\nabla}} y$: in the first case, this property holds by construction, whereas in the second case it holds by hypothesis, since the limited growth ordering '$\curvearrowright$' is ∇-compatible. By Definition 8, '$\curvearrowright$' satisfies the ascending chain condition, so that the second condition in Definition 7 also holds. Hence the '$\tilde{\nabla}$' operator is a widening. Finally, the fact that '$\tilde{\nabla}$' is at least as precise as '∇' follows directly from the definition of '$\tilde{\nabla}$'. $\square$

The above schema is easily extended to a framework for combining any finite set of upper bound operators with an existing widening to form a new widening operator with improved precision.

It should be stressed that Theorem 9 is not strong enough to ensure that the final results of upward iteration sequences computed by using the improved widening operator '$\tilde{\nabla}$' are *uniformly* more precise than those obtained by using the existing widening operator '∇'. This property would hold if both widenings were monotonic on both of their arguments. However, such a stronger requirement is rarely satisfied when considering accurate widening operators on abstract domains having infinite ascending chains.

5 An Improvement Upon the Standard Widening

In this section, we instantiate the framework presented in Theorem 9 to a new widening on the domain of convex polyhedra. In particular, we will define a widening that improves upon the precision of the standard widening '∇_S'. To do this, we need to define both a specific ∇_S-compatible lgo on $\mathbb{CP}_n$ as well as a set of upper bound operators for this domain.

5.1 A ∇_S-Compatible Limited Growth Ordering

The ∇_S-compatible lgo we use for the new widening is defined as a combination of several simpler lgo relations on the domain of convex polyhedra; for one of these, we need the following ancilliary definition.

Definition 10 (Number of non-null coordinates of a vector.) *For each $\vec{v} \in \mathbb{R}^n$, we write $\kappa(\vec{v})$ to denote the number of non-null coordinates of $\vec{v}$. For each finite set $V \subseteq \mathbb{R}^n$, we define $\kappa(V)$ to be the multiset obtained by applying 'κ' to each of the vectors in V.*

We now define a specific lgo relation as (the strict version of) the lexicographic product of five preorders on $\mathbb{CP}_n$.

Definition 11 **($\curvearrowright_N \subseteq \mathbb{CP}_n \times \mathbb{CP}_n$.)** *For $i = 1, 2$, let $\mathcal{P}_i = \mathrm{con}(\mathcal{C}_i) = \mathrm{gen}(\mathcal{G}_i) \in \mathbb{CP}_n$ be a non-empty polyhedron, where the constraint system $\mathcal{C}_i$ is in minimal form and the generator system $\mathcal{G}_i = (L_i, R_i, P_i)$ is in orthogonal form. Then the preorders $\preceq_d, \preceq_\ell, \preceq_c, \preceq_p, \preceq_r \subseteq \mathbb{CP}_n \times \mathbb{CP}_n$ are defined, respectively, as the $\varnothing$-liftings of the following relations:*

$$\mathcal{P}_1 \preceq_d \mathcal{P}_2 \quad \overset{\text{def}}{\iff} \quad \#\,\mathrm{eq}(\mathcal{C}_1) \geq \#\,\mathrm{eq}(\mathcal{C}_2); \tag{8}$$

$$\mathcal{P}_1 \preceq_\ell \mathcal{P}_2 \quad \overset{\text{def}}{\iff} \quad \#\,L_1 \leq \#\,L_2; \tag{9}$$

$$\mathcal{P}_1 \preceq_c \mathcal{P}_2 \quad \overset{\text{def}}{\iff} \quad \#\,\mathcal{C}_1 \geq \#\,\mathcal{C}_2; \tag{10}$$

$$\mathcal{P}_1 \preceq_p \mathcal{P}_2 \quad \overset{\text{def}}{\iff} \quad \#\,P_1 \geq \#\,P_2; \tag{11}$$

$$\mathcal{P}_1 \preceq_r \mathcal{P}_2 \quad \overset{\text{def}}{\iff} \quad \kappa(R_1) \sqsubseteq_{\mathrm{ms}} \kappa(R_2). \tag{12}$$

The relation $\curvearrowright_N \subseteq \mathbb{CP}_n \times \mathbb{CP}_n$ is the strict version of the lexicographic product $\preceq_n := \preceq_{d\ell cpr} \subseteq \mathbb{CP}_n \times \mathbb{CP}_n$ of the five relations '$\preceq_d$', '$\preceq_\ell$', '$\preceq_c$', '$\preceq_p$', and '$\preceq_r$', taken in this order.

Note that the relation '$\curvearrowright_N$' is well defined, since it does not depend on the particular constraint and generator representations chosen. In particular, the minimality conditions for the constraint (resp., generator) systems ensure that the relations '$\preceq_d$' and '$\preceq_c$' (resp., '$\preceq_\ell$' and '$\preceq_p$') are well defined; moreover, the orthogonality condition for the generator systems ensures that the computation of the multisets $\kappa(R_i)$ is not ambiguous, so that '$\preceq_r$' is also well-defined.

The next result shows that '$\curvearrowright_N$' satisfies the hypotheses of the framework and can be used to improve upon the standard widening.

Theorem 12 *The '$\curvearrowright_N$' relation is a ∇_S-compatible lgo on $\mathbb{CP}_n$.*

PROOF. It follows directly from Definition 11 that all the five preorders '$\preceq_d$', '$\preceq_\ell$', '$\preceq_c$', '$\preceq_p$', and '$\preceq_r$' are finitely computable. We show that all the preorders satisfy the ascending chain condition.

To see this, consider their restriction to the set $\mathcal{S} = \mathbb{CP}_n \setminus \{\varnothing\}$ of all the non-empty polyhedra and assume the notation introduced in Definition 11. As the constraint systems $\mathcal{C}_i$ and the generator systems $\mathcal{G}_i$ are in minimal form, then we have $n - \#\,\mathrm{eq}(\mathcal{C}_i) = \dim(\mathcal{P}_i)$ and $\#\,L_i = \dim\big(\mathrm{lin.space}(\mathcal{P}_i)\big)$. As these dimensions can only have values in the finite set $\{0, \ldots, n\}$, the preorders '$\preceq_d$' and '$\preceq_\ell$' both satisfy the ascending chain condition on $\mathcal{S}$. As the cardinalities of the constraint systems $\mathcal{C}_i$ and of the sets of points P_i are nonnegative and finite, the preorders '$\preceq_c$' and '$\preceq_p$' both satisfy the ascending chain condition on $\mathcal{S}$. As the cardinalities of the sets of rays R_i are nonnegative and finite, the multisets $\kappa(R_i)$ are also finite, so that the preorder '$\preceq_r$' inherits the ascending chain condition (on $\mathcal{S}$) from the multiset partial order '$\sqsubseteq_{\mathrm{ms}}$'. The extension of all the preorders on $\mathbb{CP}_n$ does not pose problems because, as noted in Section 2, the $\varnothing$-lifting preserves the ascending chain condition.

Since '$\preceq_n$' is defined as the lexicographic product of these five relations, it is still finitely computable and it satisfies the ascending chain condition so that, by Definition 8, its strict version '$\curvearrowright_N$' is a limited growth ordering on $\mathbb{CP}_n$. To complete the proof, we show that '$\curvearrowright_N$' is ∇_S-compatible. Namely, assuming that $\mathcal{P}_1 \subset \mathcal{P}_2$, we prove that $\mathcal{P}_1 \curvearrowright_N \mathcal{P}_1 \nabla_S \mathcal{P}_2$.

If $\mathcal{P}_1 = \varnothing$, then $\mathcal{P}_1 \nabla_S \mathcal{P}_2 = \mathcal{P}_2$ and, since $\mathcal{P}_1 \subset \mathcal{P}_2$, by Definition 11 we obtain $\mathcal{P}_1 \curvearrowright_N \mathcal{P}_2$. Now suppose $\mathcal{P}_1 \neq \varnothing$, so that also $\mathcal{P}_2 \neq \varnothing$, and assume the notation introduced in Definition 11.

Let $\mathcal{P}_1 \nabla_S \mathcal{P}_2 = \mathcal{P}$ and consider the constraint systems $\mathcal{I}'_1$ and $\mathcal{I}'_2$ as specified in Definition 1. Then, $\mathcal{P}_2 \subseteq \mathrm{con}(\mathcal{I}'_1)$ since $\mathcal{P}_2 \subseteq \mathrm{con}\bigl(\{\beta\}\bigr)$ for all $\beta \in \mathcal{I}'_1$; and also $\mathcal{P}_2 \subseteq \mathrm{con}(\mathcal{I}'_2)$ since $\mathcal{I}'_2 \subseteq \mathrm{repr}_{\geq}(\mathcal{C}_2)$. Thus,

$$\mathcal{P}_2 \subseteq \mathrm{con}(\mathcal{I}'_1) \cap \mathrm{con}(\mathcal{I}'_2) = \mathrm{con}(\mathcal{I}'_1 \cup \mathcal{I}'_2) = \mathcal{P}.$$

Since $\mathcal{P}_1 \subset \mathcal{P}_2$, we also obtain $\mathcal{P}_1 \subset \mathcal{P}$ so that $\dim(\mathcal{P}_1) \leq \dim(\mathcal{P})$ and $\dim\bigl(\mathrm{lin.space}(\mathcal{P}_1)\bigr) \leq \dim\bigl(\mathrm{lin.space}(\mathcal{P})\bigr)$. Let $\mathcal{P} = \mathrm{con}(\mathcal{C}) = \mathrm{gen}(\mathcal{G})$, where the constraint system $\mathcal{C}$ is in minimal form and the generator system $\mathcal{G} = (L, R, P)$ is in orthogonal form. From the previous dimensionality properties, it follows that $\#\,\mathrm{eq}(\mathcal{C}_1) \geq \#\,\mathrm{eq}(\mathcal{C})$ and $\#\,L_1 \leq \#\,L$ so that, by Definition 11, $\mathcal{P}_1 \preceq_d \mathcal{P}$ and $\mathcal{P}_1 \preceq_\ell \mathcal{P}$.

If $\mathcal{P}_1 \prec_d \mathcal{P}$ or $\mathcal{P}_1 \prec_\ell \mathcal{P}$, then we obtain $\mathcal{P}_1 \curvearrowright_N \mathcal{P}$. Otherwise, let $\mathcal{P}_1 \equiv_d \mathcal{P}$ and $\mathcal{P}_1 \equiv_\ell \mathcal{P}$. As $\mathcal{P}_1 \subset \mathcal{P}_2 \subseteq \mathcal{P}$, from $\mathcal{P}_1 \equiv_d \mathcal{P}$ we also obtain $\mathcal{P}_1 \equiv_d \mathcal{P}_2$, so that $\dim(\mathcal{P}_1) = \dim(\mathcal{P}_2)$. Thus, Proposition 6 applies and we obtain $\mathcal{P} = \mathrm{con}(\mathcal{C}_d)$, where

$$\mathcal{C}_d = \Bigl\{\, \beta \in \mathcal{C}_1 \Bigm| \mathcal{P}_2 \subseteq \mathrm{con}\bigl(\{\beta\}\bigr) \,\Bigr\}.$$

As $\mathcal{C}_1$ is in minimal form and $\mathcal{C}_d \subseteq \mathcal{C}_1$, also $\mathcal{C}_d$ is in minimal form. Moreover, $\mathcal{P}_1 \subset \mathcal{P}_2 \subseteq \mathcal{P}$ implies $\mathcal{C}_d \neq \mathcal{C}_1$, so that $\#\,\mathcal{C}_1 > \#\,\mathcal{C}_d$. Thus we obtain $\mathcal{P}_1 \equiv_d \mathcal{P}$, $\mathcal{P}_1 \equiv_\ell \mathcal{P}$ and $\mathcal{P}_1 \prec_c \mathcal{P}$, which together imply $\mathcal{P}_1 \curvearrowright_N \mathcal{P}$. $\square$

The '$\curvearrowright_N$' relation is a variant of a similar notion of limited growth defined in [3, Theorem 3]. These two proposals are not formally comparable since neither one of the relations refines the other. On one hand, in Definition 11 we consider preorders that were not considered in [3], namely '$\preceq_c$' and '$\preceq_r$'; on the other hand, due to the specific lexicographic product computed, the preorder '$\preceq_p$' comes into play only when the iteration is stable with respect

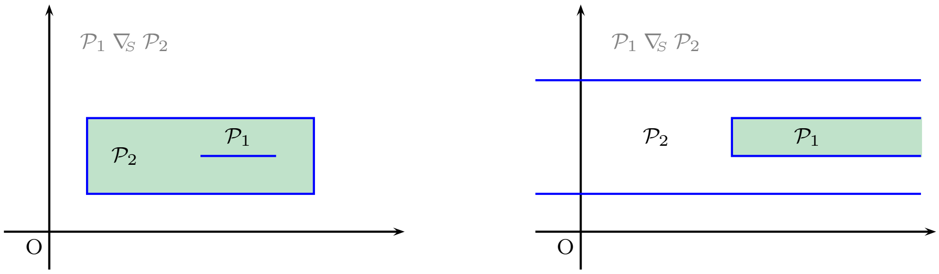

Fig. 1. The "do not widen" heuristics improving on the standard widening.

to '$\preceq_c$'. Moreover, the relation defined in [3] is not ∇_S-compatible: neither the standard widening '∇_S', nor the heuristics informally sketched in [3] can ensure that consecutive iterates satisfy the given notion of limited growth. In summary, the overall approach in [3] does not define a widening operator in the precise sense of Definition 7 [F. Besson, personal communication, 2002].

5.2 The Heuristic Techniques

We now present the four different heuristic techniques, later shown to be upper bound operators, that we will use for constructing the new widening.

5.2.1 First Technique: Do Not Widen

The simplest heuristics, already suggested in [15] and adopted in [3], is the one saying "do not widen": if we are along an iteration chain having finite length, there is no need to provide further approximations, so that we can safely return the most precise upper bound $\mathcal{P}_2$ (remember that we assume $\mathcal{P}_1 \subset \mathcal{P}_2$). In our context, this is the case whenever $\mathcal{P}_1 \curvearrowright_N \mathcal{P}_2$.

Figure 1 shows two examples where the "do not widen" technique is able to improve on the standard widening. In the left hand diagram, the result of the application of the standard widening to the line segment $\mathcal{P}_1$ and the rectangle $\mathcal{P}_2$ is $\mathcal{P}_1 \nabla_S \mathcal{P}_2 = \mathbb{R}^2$. Since $\dim(\mathcal{P}_1) = 1 < 2 = \dim(\mathcal{P}_2)$, we have $\mathcal{P}_1 \prec_d \mathcal{P}_2$, which implies $\mathcal{P}_1 \curvearrowright_N \mathcal{P}_2$. In the right hand diagram, the result of the application of the standard widening to the half stripe polyhedron $\mathcal{P}_1$ and the full stripe polyhedron $\mathcal{P}_2$ is again $\mathbb{R}^2$. Since $\dim(\mathcal{P}_1) = \dim(\mathcal{P}_2) = 2$ and $\dim\bigl(\mathrm{lin.space}(\mathcal{P}_1)\bigr) = 0 < 1 = \dim\bigl(\mathrm{lin.space}(\mathcal{P}_2)\bigr)$, we obtain $\mathcal{P}_1 \equiv_d \mathcal{P}_2$ and $\mathcal{P}_1 \prec_\ell \mathcal{P}_2$, which again imply $\mathcal{P}_1 \curvearrowright_N \mathcal{P}_2$. Thus, in both cases, the "do not widen" heuristics will return the most precise upper bound $\mathcal{P}_2$.

As this heuristic technique, when applicable, returns the most precise result, it has to be tried first. As a consequence, all the other widening techniques considered here including the standard widening, are only applied to a pair of polyhedra $\mathcal{P}_1$ and $\mathcal{P}_2$ such that $\mathcal{P}_1 \not\curvearrowright_N \mathcal{P}_2$: by Definition 11, this implies both $\dim(\mathcal{P}_1) \geq \dim(\mathcal{P}_2)$ and $\dim\bigl(\mathrm{lin.space}(\mathcal{P}_1)\bigr) \geq \dim\bigl(\mathrm{lin.space}(\mathcal{P}_2)\bigr)$ so that, by the hypothesis $\mathcal{P}_1 \subset \mathcal{P}_2$, we also obtain $\mathrm{aff.hull}(\mathcal{P}_1) = \mathrm{aff.hull}(\mathcal{P}_2)$ and $\mathrm{lin.space}(\mathcal{P}_1) = \mathrm{lin.space}(\mathcal{P}_2)$, respectively. For these other techniques, since we cannot return the most precise upper bound $\mathcal{P}_2$, we have to select what information will be lost. Informally, we will try to preserve the information provided by stable components, whereas the information of components that have changed will be extrapolated according to a hypothetical "change pattern." For instance, in the case of the widening in [1], each element of a constraint system is regarded as a separate component and the extrapolation just forgets about the constraints that have changed.

5.2.2 Second Technique: Combining Constraints

The second heuristics, which is a variant of a similar one sketched in [3], can be seen as an application of the above approach, where instead of the constraints we consider the points in the generator system describing the polyhedron of the previous iteration. When using the standard widening it may happen that points that are common to the boundaries of $\mathcal{P}_1$ and $\mathcal{P}_2$ (and, hence, likely to be an invariant feature along the chain of polyhedra) will not lie on the boundary of the widened polyhedron. This is the case, for instance, for the points $\vec{p}$ and $\vec{q}$ in Figure 2. For such a point, the technique forces the presence of an inequality constraint that is saturated by the point, so that they will lie on the boundary of the result.

Definition 13 (Combining Constraints.) *Let $\mathcal{P}_1, \mathcal{P}_2 \in \mathbb{CP}_n$ be such that $\mathcal{P}_1 \subset \mathcal{P}_2$, $\mathrm{aff.hull}(\mathcal{P}_1) = \mathrm{aff.hull}(\mathcal{P}_2)$ and $\mathrm{lin.space}(\mathcal{P}_1) = \mathrm{lin.space}(\mathcal{P}_2)$. Let $\mathcal{P}_1 = \mathrm{gen}(\mathcal{G}_1)$, $\mathcal{P}_2 = \mathrm{con}(\mathcal{C}_2)$ and $\mathcal{P}_1 \nabla_S \mathcal{P}_2 = \mathrm{con}(\mathcal{C}_S)$, where the constraint systems $\mathcal{C}_2$, $\mathcal{C}_S$ and the generator system $\mathcal{G}_1 = (L_1, R_1, P_1)$ are in orthogonal form. Let also*

$$\mathcal{C}_\oplus := \left\{ \oplus(\mathcal{C}_{\vec{p}}) \;\middle|\; \begin{array}{l} \vec{p} \in P_1, \mathrm{sat\_con}\bigl(\vec{p}, \mathrm{ineq}(\mathcal{C}_S)\bigr) = \varnothing, \\ \mathcal{C}_{\vec{p}} = \mathrm{sat\_con}\bigl(\vec{p}, \mathrm{ineq}(\mathcal{C}_2)\bigr) \neq \varnothing \end{array} \right\},$$

where the operator '$\oplus$' computes a convex combination of a non-empty set of linear inequality constraints (i.e., of the corresponding coefficients), returning another linear inequality constraint. Then $h_c(\mathcal{P}_1, \mathcal{P}_2) := \mathrm{con}(\mathcal{C}_S \cup \mathcal{C}_\oplus)$.

Since the operator 'h_c' is only defined for arguments having the same affine hull and lineality space, by requiring orthogonal forms we ensure that the result does not depend on the particular representations considered.

Note that the particular convex combination encoded by function '$\oplus$' is deliberately left unspecified so as to allow for a very liberal definition of 'h_c'

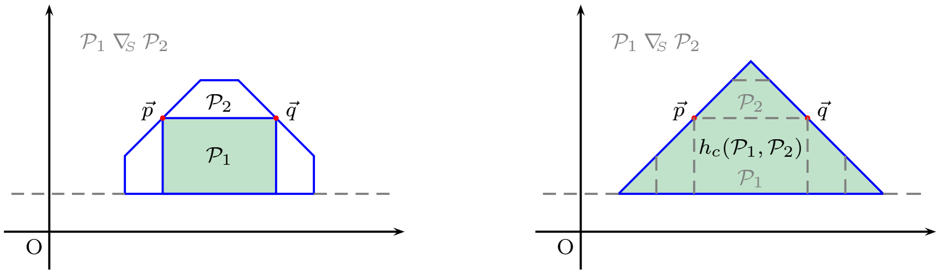

Fig. 2. The heuristics 'h_c' improving on the standard widening.

that still possesses the required properties. For instance, in [3] it was argued that a good heuristics could be obtained by letting '$\oplus$' compute a normed linear combination (i.e., a sort of average) of the chosen constraints. Another legitimate choice would be to "bless" one of the constraints in $\mathcal{C}_{\vec{p}}$ and forget all the others. In both cases, by keeping just one constraint for each point $\vec{p}$, we hopefully reduce the cardinality of the constraint system describing the result, so that it is more likely that a strict increase on the preorder '$\preceq_c$' will be obtained. Actually, this attempt at reducing the number of constraints is the main difference between the technique presented in Definition 13 and the extrapolation operator proposed in [19, Section 3.3], which could itself be included in the current framework as a more refined widening heuristics.

5.2.3 Third Technique: Evolving Points

Our third heuristic technique is a variant of the extrapolation operator '$\propto$' defined in [18]. The technique examines each new point $\vec{p}_2$ of the polyhedron $\mathcal{P}_2$ as if it was obtained from each old point $\vec{p}_1$ of the polyhedron $\mathcal{P}_1$: we say that $\vec{p}_2$ is an evolution of $\vec{p}_1$. The extrapolation is defined as continuing this evolution towards infinity, therefore generating the ray having direction $\vec{p}_2 - \vec{p}_1$. To ensure the resulting polyhedron is at least as precise as the standard widening, any new rays that violate a constraint of the standard widening are dropped. Note that any remaining new rays will subsume the point $\vec{p}_2$, so that it is likely that a strict increase in the preorder '$\preceq_p$' will be obtained.

Definition 14 (Evolving Points.) *Let $\mathcal{P}_1, \mathcal{P}_2 \in \mathbb{CP}_n$ be such that $\mathcal{P}_1 \subset \mathcal{P}_2$ and $\operatorname{lin.space}(\mathcal{P}_1) = \operatorname{lin.space}(\mathcal{P}_2)$. For each $i = 1, 2$, consider a generator system $\mathcal{G}_i = (L_i, R_i, P_i)$ in orthogonal form such that $\mathcal{P}_i = \operatorname{gen}(\mathcal{G}_i)$ and let*

$$R := \left\{ \vec{p}_2 - \vec{p}_1 \;\middle|\; \vec{p}_1 \in P_1, \vec{p}_2 \in P_2 \setminus P_1 \right\}.$$

Then we define $h_p(\mathcal{P}_1, \mathcal{P}_2) = \operatorname{gen}\Bigl((L_2, R_2 \cup R, P_2)\Bigr) \cap (\mathcal{P}_1 \nabla_S \mathcal{P}_2)$.

Since the operator 'h_p' is only defined for arguments having the same lineality space, by requiring orthogonal forms we ensure that the result does not depend on the particular generator system representations considered.

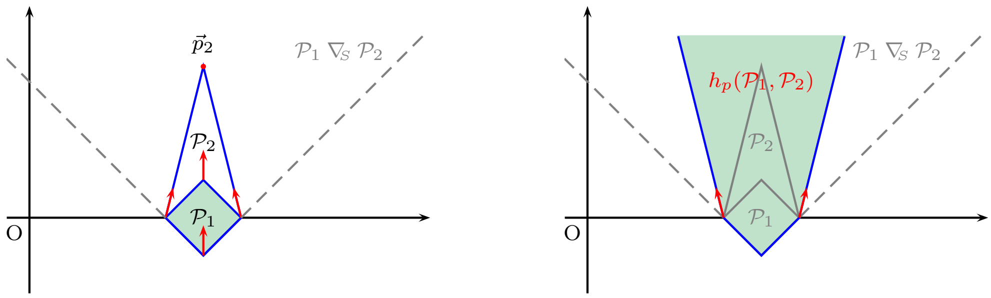

Fig. 3. The heuristics 'h_p' improving on the standard widening.

Figure 3 shows an example where the "evolving points" technique is able to improve on the standard widening. Note that the boundary of $\mathcal{P}_1 \nabla_S \mathcal{P}_2$ contains the intersection of the boundaries of $\mathcal{P}_1$ and $\mathcal{P}_2$, so that the "combining constraints" technique is not applicable. Besides having the same affine hull and lineality space, polyhedra $\mathcal{P}_1$, $\mathcal{P}_2$ and $h_p(\mathcal{P}_1, \mathcal{P}_2)$ are defined by constraint systems in minimal form having the same cardinality, so that $\mathcal{P}_1 \curvearrowright_N h_r(\mathcal{P}_1, \mathcal{P}_2)$ holds because we have a strict increase in the preorder '$\preceq_p$'.

The difference with respect to the extrapolation operator '$\propto$' is that we do not require the two points to lie on the same 1-dimensional face of $\mathcal{P}_2$; moreover, the result of '$\propto$' may be less precise than the standard widening. Note that, as in the "combining constraints" technique, it is possible to add just a single ray which is a convex combination of the rays in R instead of the complete set R, yielding a more precise widening technique. However, this technique and the one defined by the 'h_p' operator are incomparable with respect to the '$\curvearrowright_N$' relation and one can fail the '$\curvearrowright_N$' convergence criteria when the other succeeds.

5.2.4 Fourth Technique: Evolving Rays

In the fourth heuristic technique (which is new), we try to extrapolate the way rays have evolved since the last iteration. The technique examines each new ray $\vec{r}_2$ of the polyhedron $\mathcal{P}_2$ as if it was generated by rotation of each old ray $\vec{r}_1$ of the polyhedron $\mathcal{P}_1$: we say that $\vec{r}_2$ is an evolution of $\vec{r}_1$. The extrapolation is defined as continuing this evolution until one or more of the non-null coordinates of ray $\vec{r}_2$ become zero. This way, it is likely that a strict increase in the preorder '$\preceq_r$' will be obtained. Intuitively, the new ray will reach one of the boundaries of the orthant where $\vec{r}_2$ lies, without trespassing it. As for the previous heuristics, to ensure the resulting polyhedron is at least as precise as the standard widening, any new rays that violate a constraint of the standard widening are dropped.

Definition 15 (evolve.) *The function* evolve: $\mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ *is defined, for* *each* $\vec{u}, \vec{v} \in \mathbb{R}^n$*, as* $\mathrm{evolve}(\vec{u}, \vec{v}) := \vec{w}$*, where*

$$w_i := \begin{cases} 0, & \text{if } \exists j \in \{1, \ldots, n\} \mathrel{.} (u_i v_j - u_j v_i) u_i u_j < 0, \\ u_i, & \text{otherwise.} \end{cases}$$

To understand this definition consider a pair of coordinates i and j and suppose that the vectors $\vec{u}$ and $\vec{v}$ are projected onto the two-dimensional plane defined by i (for the first coordinate) and j (for the second coordinate). Then, we identify the direction of the rotation of the vector $(u_i, u_j)^{\mathrm{T}}$ with respect to the vector $(v_i, v_j)^{\mathrm{T}}$ by using the well-known cross-product test [36, Chapter 35]; the direction is clockwise if $c := u_i v_j - u_j v_i > 0$ and anti-clockwise when $c < 0$. Moreover, vector $(u_i, u_j)^{\mathrm{T}}$ lies inside the first or third quadrant when $q = u_i u_j > 0$ and it lies inside the second or fourth quadrant when $q < 0$. Then, the condition $cq < 0$ states that the evolution is clockwise and $(u_i, u_j)^{\mathrm{T}}$ is in the second or fourth quadrant or the evolution is anti-clockwise and $(u_i, u_j)^{\mathrm{T}}$ is in the first or third quadrant: in all these cases, the evolution is towards the j axis. Thus, for a fixed i, if there exists j such that the evolution is towards the j axis, then we define $w_i = 0$. Otherwise, we let $w_i = u_i$. We are now ready to define our last widening heuristics.

Definition 16 (Evolving Rays.) *Let* $\mathcal{P}_1, \mathcal{P}_2 \in \mathbb{CP}_n$ *be such that* $\mathcal{P}_1 \subset \mathcal{P}_2$ *and* $\mathrm{lin.space}(\mathcal{P}_1) = \mathrm{lin.space}(\mathcal{P}_2)$*. For each* $i = 1, 2$*, consider a generator system* $\mathcal{G}_i = (L_i, R_i, P_i)$ *in orthogonal form such that* $\mathcal{P}_i = \mathrm{gen}(\mathcal{G}_i)$ *and let*

$$R := \Bigl\{ \mathrm{evolve}(\vec{r}_2, \vec{r}_1) \Bigm| \vec{r}_1 \in R_1, \vec{r}_2 \in R_2 \setminus R_1 \Bigr\}.$$

Then we define $h_r(\mathcal{P}_1, \mathcal{P}_2) := \mathrm{gen}\Bigl((L_2, R_2 \cup R, P_2)\Bigr) \cap (\mathcal{P}_1 \nabla_S \mathcal{P}_2)$*.*

Figure 4 shows an example where the "evolving rays" technique is able to improve on the standard widening. It should be noted that the boundary of $\mathcal{P}_1 \nabla_S \mathcal{P}_2$ contains the intersection of the boundaries of $\mathcal{P}_1$ and $\mathcal{P}_2$, so that the "combining constraints" technique is not applicable. Neither the "evolving points" technique can be applied, since $\mathcal{P}_1$ and $\mathcal{P}_2$ are defined by generator systems in orthogonal form having the same set of points. Besides having the same affine hull and lineality space, polyhedra $\mathcal{P}_1$, $\mathcal{P}_2$ and $h_r(\mathcal{P}_1, \mathcal{P}_2)$, are defined by constraint and generator systems in minimal form having the same number of constraints and points, so that $\mathcal{P}_1 \curvearrowright_N h_r(\mathcal{P}_1, \mathcal{P}_2)$ holds because we have a strict increase in the preorder '$\preceq_r$'.

5.3 The New Widening

In order to use these heuristic techniques in the general framework of Theorem 9, we must show that each of them is an upper bound operator. The first

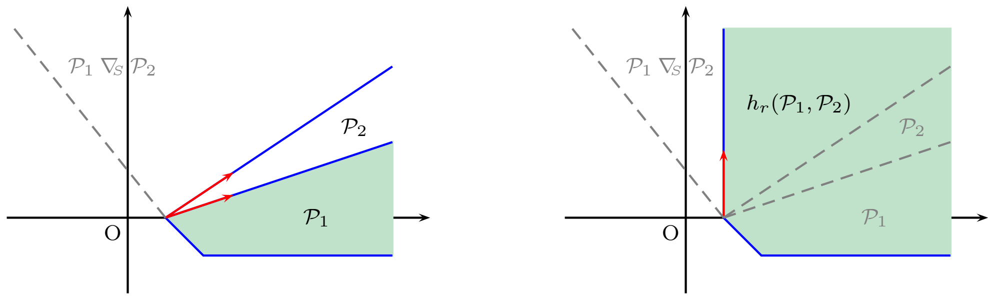

Fig. 4. The heuristics 'h_r' improving on the standard widening.

technique is by definition an upper bound operator. We now show that the other three techniques are also upper bound operators.

Proposition 17 *Let* $\mathcal{P}_1, \mathcal{P}_2 \in \mathbb{CP}_n$*, where* $\mathcal{P}_1 \subset \mathcal{P}_2$, $\operatorname{aff.hull}(\mathcal{P}_1) = \operatorname{aff.hull}(\mathcal{P}_2)$ *and* $\operatorname{lin.space}(\mathcal{P}_1) = \operatorname{lin.space}(\mathcal{P}_2)$*. Then, for each technique* $h \in \{h_c, h_p, h_r\}$, $\mathcal{P}_2 \subseteq h(\mathcal{P}_1, \mathcal{P}_2) \subseteq \mathcal{P}_1 \nabla_S \mathcal{P}_2$.

PROOF. Let $\mathcal{P}_t = h(\mathcal{P}_1, \mathcal{P}_2)$. Consider first the case when $h = h_c$ and assume the notation introduced in Definition 13. The proof for $\mathcal{P}_t \subseteq \mathcal{P}_1 \nabla_S \mathcal{P}_2$ is immediate, since $\mathcal{P}_t$ is defined by a constraint system $\mathcal{C}_S \cup \mathcal{C}_\oplus$ including all of the constraints defining $\mathcal{P}_1 \nabla_S \mathcal{P}_2$. To prove that $\mathcal{P}_2 \subseteq \mathcal{P}_t$ we show that $\mathcal{P}_2 \subseteq \operatorname{con}\bigl(\{\beta\}\bigr)$, for each constraint $\beta \in \mathcal{C}_S \cup \mathcal{C}_\oplus$ defining $\mathcal{P}_t$. Clearly, if $\beta \in \mathcal{C}_S$ then the inclusion holds by the fact that the standard widening is an upper bound operator. If otherwise $\beta \in \mathcal{C}_\oplus$, then, for some $\mathcal{C}_{\vec{p}} \subseteq \operatorname{ineq}(\mathcal{C}_2)$, we have $\beta = \oplus(\mathcal{C}_{\vec{p}})$, so that $\mathcal{P}_2 \subseteq \operatorname{con}(\mathcal{C}_{\vec{p}}) \subseteq \operatorname{con}\bigl(\{\beta\}\bigr)$.

Next, consider the cases when $h \in \{h_p, h_r\}$ and assume the notation introduced in Definitions 14 and 16. Let $\mathcal{G}' = (L_2, R_2 \cup R, P_2)$ and $\mathcal{P}' = \operatorname{gen}(\mathcal{G}')$; then $\mathcal{P}_t = \mathcal{P}' \cap (\mathcal{P}_1 \nabla_S \mathcal{P}_2)$. Thus $\mathcal{P}_t \subseteq \mathcal{P}_1 \nabla_S \mathcal{P}_2$. As $\mathcal{G}_2 \sqsubseteq_G \mathcal{G}'$, we obtain $\mathcal{P}_2 \subseteq \mathcal{P}'$. Moreover, since the standard widening is an upper bound operator, we also have $\mathcal{P}_2 \subseteq \mathcal{P}_1 \nabla_S \mathcal{P}_2$. Therefore, by the monotonicity of set intersection, we conclude $\mathcal{P}_2 \subseteq \mathcal{P}_t$. $\square$

We now present our new widening operator.

Definition 18 (The '∇_N' widening.) *Let* $\mathcal{P}_1, \mathcal{P}_2 \in \mathbb{CP}_n$*, where* $\mathcal{P}_1 \subset \mathcal{P}_2$*. Then*

$$\mathcal{P}_1 \nabla_N \mathcal{P}_2 := \begin{cases} \mathcal{P}_2, & \text{if } \mathcal{P}_1 \curvearrowright_N \mathcal{P}_2; \\ h_c(\mathcal{P}_1, \mathcal{P}_2), & \text{if } \mathcal{P}_1 \curvearrowright_N h_c(\mathcal{P}_1, \mathcal{P}_2) \subset \mathcal{P}_1 \nabla_S \mathcal{P}_2; \\ h_p(\mathcal{P}_1, \mathcal{P}_2), & \text{if } \mathcal{P}_1 \curvearrowright_N h_p(\mathcal{P}_1, \mathcal{P}_2) \subset \mathcal{P}_1 \nabla_S \mathcal{P}_2; \\ h_r(\mathcal{P}_1, \mathcal{P}_2), & \text{if } \mathcal{P}_1 \curvearrowright_N h_r(\mathcal{P}_1, \mathcal{P}_2) \subset \mathcal{P}_1 \nabla_S \mathcal{P}_2; \\ \mathcal{P}_1 \nabla_S \mathcal{P}_2, & \text{otherwise.} \end{cases}$$

It can be seen that '∇_N' is an instance of the framework given in Theorem 9: in particular, when applying the first heuristics, the omission of the applicability condition $\mathcal{P}_2 \subset \mathcal{P}_1 \nabla_S \mathcal{P}_2$ is a simple and inconsequential optimization. Thus, the following result is a direct consequence of Theorems 9 and 12 and Proposition 17.

Proposition 19 *The '∇_N' operator is a widening at least as precise as '∇_S'.*

PROOF. Suppose that $\mathcal{P}_1, \mathcal{P}_2 \in \mathbb{CP}_n$, where $\mathcal{P}_1 \subset \mathcal{P}_2$, so that Definition 18 applies. If $\mathcal{P}_2 = \mathcal{P}_1 \nabla_S \mathcal{P}_2$, then $\mathcal{P}_1 \nabla_N \mathcal{P}_2 = \mathcal{P}_1 \nabla_S \mathcal{P}_2$. By Theorem 12, '$\curvearrowright_N$' is a ∇_S-compatible lgo on $\mathbb{CP}_n$. Moreover, when $\mathcal{P}_2 \subset \mathcal{P}_1 \nabla_S \mathcal{P}_2$, all the heuristic techniques used in Definition 18 are upper bound operators since the first technique returns the least upper bound $\mathcal{P}_2$ while, for the other techniques, this is a consequence of Proposition 17. Therefore we can apply Theorem 9 to obtain the thesis. $\square$

As already explained at the end of the previous section, Proposition 19 cannot ensure that a static analysis that is using the new widening will never be less precise than the same analysis but using the standard widening. The reasons are illustrated in the next example, where we show that the standard widening is not monotonic on its first argument [1] and the new widening is not monotonic on both arguments.

Example 20 *Consider the polyhedral domain $\mathbb{CP}_2$ and let*

$$\begin{aligned}
\mathcal{P}_1 &= \operatorname{con}\Big(\{2 \le x \le 3, 2 \le y \le 3\}\Big),\\
\mathcal{P}_1' &= \operatorname{con}\Big(\{0 \le x \le 5, 0 \le y \le 5, 2 \le x + y \le 8, -3 \le x - y \le 3\}\Big),\\
\mathcal{P}_2 &= \operatorname{con}\Big(\{0 \le x \le 5, 0 \le y \le 5, 1 \le x + y \le 9, -4 \le x - y \le 4\}\Big),
\end{aligned}$$

so that $\mathcal{P}_1 \subset \mathcal{P}_1' \subset \mathcal{P}_2$. By Definitions 1 *and* 18*, noting that both $\mathcal{P}_1 \not\curvearrowright_N \mathcal{P}_2$ and $\mathcal{P}_1' \not\curvearrowright_N \mathcal{P}_2$, we obtain*

$$\begin{aligned}
\mathcal{P}_1 \nabla_S \mathcal{P}_2 &= \mathcal{P}_1 \nabla_N \mathcal{P}_2 = \mathbb{R}^2,\\
\mathcal{P}_1' \nabla_S \mathcal{P}_2 &= \mathcal{P}_1' \nabla_N \mathcal{P}_2 = \operatorname{con}\Big(\{0 \le x \le 5, 0 \le y \le 5\}\Big).
\end{aligned}$$

Thus, we have $\mathcal{P}_1 \nabla_S \mathcal{P}_2 \nsubseteq \mathcal{P}_1' \nabla_S \mathcal{P}_2$ and $\mathcal{P}_1 \nabla_N \mathcal{P}_2 \nsubseteq \mathcal{P}_1' \nabla_N \mathcal{P}_2$, showing that neither the standard widening nor the new widening are monotonic on the first argument. Consider now

$$\begin{aligned}
\mathcal{Q}_1 &= \operatorname{con}\Big(\{1 \le x \le 2, 1 \le y \le 2\}\Big),\\
\mathcal{Q}_2 &= \operatorname{con}\Big(\{0 \le x \le 3, 0 \le y \le 3\}\Big),\\
\mathcal{Q}_2' &= \operatorname{con}\Big(\{x \ge 0, y \ge 0, x + y \le 6\}\Big),
\end{aligned}$$

so that $\mathcal{Q}_1 \subset \mathcal{Q}_2 \subset \mathcal{Q}_2'$. *By* Definition 18, *as* $\mathcal{Q}_1 \not\curvearrowright_N \mathcal{Q}_2$ *but* $\mathcal{Q}_1 \curvearrowright_N \mathcal{Q}_2'$,

$$\begin{aligned} \mathcal{Q}_1 \nabla_N \mathcal{Q}_2 &= \mathcal{Q}_1 \nabla_S \mathcal{Q}_2 = \mathbb{R}^2, \\ \mathcal{Q}_1 \nabla_N \mathcal{Q}_2' &= \mathcal{Q}_2'. \end{aligned}$$

Thus, we obtain $\mathcal{Q}_1 \nabla_N \mathcal{Q}_2 \not\subseteq \mathcal{Q}_1 \nabla_N \mathcal{Q}_2'$, *showing that the new widening is monotonic neither on its second argument.*

Note that in spite of this lack of monotonicity the experimental evaluation reported in the next section shows that, for the considered application, precision degradations are very rare.

6 Experimental Evaluation

We have extended the *Parma Polyhedra Library* (PPL) [34,35], a modern C++ library for the manipulation of convex polyhedra, with a prototype implementation of the widening of Definition 18. The PPL has been integrated with the China analyzer [37] for the purpose of detecting linear argument size relations [8]. Our benchmark suite consists of 361 Prolog programs, ranging from small synthetic benchmarks to real-world applications. They define 23279 predicates whose analysis with China requires the direct use of a widening and about as many predicates for which no widening is used. In this respect, it must be noted that China employs a sophisticate chaotic iteration strategy proposed in [38,39] that, among other benefits, allows to greatly reduce the number of widenings' applications. [2] This is an important point, since it would be quite easy to improve on an iteration strategy applying widenings "everywhere or improperly" [38]. The results of this experimental evaluation are summarized in Tables 1 and 2, where each row corresponds to a different choice for the value of the extrapolation threshold k, controlling the delay before the applications of both the standard and the new widening operators.

Table 1 shows the obtained precision improvements and degradations, both in terms of the number of programs and the number of predicates affected; in the columns labeled 'incomp' we report those cases where incomparable results have been obtained. For $k = 0$, we observe a precision improvement on one third of the considered programs; not surprisingly, fewer improvements are obtained for higher values of k, but we still have an improvement on 7% of the benchmarks when considering $k = 10$. While confirming, as informally argued in [8], that for this particular analysis there is little incentive in using

[2] China uses the recursive fixpoint iteration strategy on the weak topological ordering defined by partitioning of the call graph into strongly-connected subcomponents [39].

	# programs			# predicates		
k (delay)	improve	degr	incomp	improve	degr	incomp
0	121	0	2	1340	3	2
1	34	0	0	273	0	0
2	29	0	0	222	0	0
3	28	0	0	160	0	0
4	25	0	2	126	2	0
10	25	0	0	124	0	0

Table 1
Precision comparisons.

values of k greater than 4, our experiments show that the new widening captures growth patterns that do happen in practice and that for the standard widening (no matter how delayed) are out of reach. This is important since the results obtained in practice are, besides correctness, what really matters when evaluating widening operators. The experimentation also shows that the idea of delaying the widening [16] maintains its validity: even though the new widening is less sensible to the amount of delay applied, the results are still sensibly improved by delaying.

Table 2 shows the sum, over all the benchmarks, of the fixpoint computation times. This is expressed as a proportion of the time spent when using the standard widening with $k = 0$. Since smaller benchmarks may affect the outcome of this summarization, in the columns labeled 'top 20' we also show the same values but restricted to the 20 benchmarks whose analysis takes more time. It can be seen that the new widening has a negative, but relatively modest impact on efficiency, which anyway is smaller than the cost of increasing the value of k. When looking at these time results, it should be considered that we are comparing a prototype implementation of the new widening with respect to a rather optimized implementation of the standard widening. It is also important to remark that the good performance degradation observed for both widenings when increasing the value of k is essentially due to the iteration strategy employed by China and should not be expected to automatically carry over to systems using other fixpoint computation techniques.

7 An Improved Delay Technique

The technique of employing an extrapolation threshold k has been traditionally implemented (and our experimental evaluation makes no exception) in a

	$\overset{k}{\nabla}_S$		$\overset{k}{\nabla}_N$	
k (delay)	all	top 20	all	top 20
0	1.00	0.72	1.05	0.77
1	1.09	0.79	1.11	0.80
2	1.16	0.83	1.18	0.84
3	1.23	0.88	1.25	0.89
4	1.32	0.95	1.34	0.95
10	1.82	1.23	1.85	1.24

Table 2
Time comparisons.

"simple way" [15], as a blind delay in the application of the widening. Namely, for each widening operator '∇', the widening operator '$\overset{k}{\nabla}$' is formalized as follows, where each abstract value is a pair recording, in its second component, the iterations in which it has been computed:

$$\langle x, i\rangle \overset{k}{\nabla} \langle y, i+1\rangle := \begin{cases} \langle x, i+1\rangle, & \text{if } y \sqsubseteq x; \\ \langle x \sqcup y, i+1\rangle, & \text{if } i < k; \\ \langle x \nabla y, i+1\rangle, & \text{otherwise.} \end{cases}$$

Thus, no matter what abstract value would have been computed by the widening, the widening is never applied in the first k iteration steps and it is always applied in all the following iteration steps.

In our opinion, a better approximation strategy can be obtained by interpreting the value k as the maximum number of iterations for which the computation of the widening can be safely avoided. Thus, an abstract value is a pair carrying a number of "tokens" t, each of them allowing for the replacement of one widening application by the exact upper bound. Aiming at an improvement in the final result, each widening operator should be left free to choose when to use the available tokens. For instance, tokens should not be wasted when the widening is precise, that is, when it simply computes the least upper bound of its arguments. The following definition of '$\overset{\circ}{\nabla}$' (widening with tokens) formalizes this idea:

$$\langle x, t\rangle \overset{\circ}{\nabla} \langle y, \cdot\rangle := \begin{cases} \langle x, t\rangle, & \text{if } y \sqsubseteq x; \\ \langle x \nabla y, t\rangle, & \text{if } x \nabla y = x \sqcup y; \\ \langle x \sqcup y, t-1\rangle, & \text{if } t > 0; \\ \langle x \nabla y, 0\rangle, & \text{otherwise.} \end{cases}$$

The iteration sequence will begin with abstract values of the form $\langle x_0, k\rangle$, that is, with k tokens where k is a parameter of the analysis; the number of tokens will decrease along the iteration chain and, when there are no tokens left, the widening will always be applied. Notice that, when instantiating the above construction with our new widening operator '∇_N' (and assuming the inclusion hypothesis), the conditional guard for the second case of the definition of '$\mathring{\nabla}$' becomes $\mathcal{P}_1 \nabla_N \mathcal{P}_2 = \mathcal{P}_2$, which can be easily implemented by performing the test $\mathcal{P}_1 \curvearrowright_N \mathcal{P}_2$.

Also note that more general definitions for '$\mathring{\nabla}$' are possible: for instance, when $x \nabla y \neq x \sqcup y$ and $t > 0$ (i.e., the widening does not compute the exact upper bound and there still are tokens available), we may nonetheless choose to apply the widening operator, provided the corresponding approximation is good enough. This way, we may preserve the tokens and use them to avoid some later approximations, which could be much coarser than the current one. Clearly, such an approach depends on the particular formalization of the notion of "good enough", which is, along with the value of k, intrinsically application dependent.

8 Conclusion and Related Work

For the domain of convex polyhedra, the convergence guarantee of the fixpoint computation sequence has been traditionally obtained thanks to the widening operator proposed by Cousot and Halbwachs. Though remarkably precise, this operator does not fulfill the requirements of a number of systems' analysis and verification applications that are particularly sensitive to the precision of the deduced numerical information. In this paper, elaborating on an idea proposed in [3], we have defined a framework for the systematic specification of new widening operators improving on the precision of an existing widening. The framework allows any upper bound operator on the abstract domain to be used as a heuristic technique for improving precision, while still ensuring the termination of the abstract computation. We have instantiated the framework on the domain of convex polyhedra with a selection of extrapolation operators, some of which embody improvements of heuristics already proposed in the literature. A first experimental evaluation has yielded promising results. The experimental work has also suggested that the well-known widening delay technique can be improved, yet retaining its overall simplicity. Our proposal is to delay the widening application only when this prevents *actual* (as opposed to *potential*) precision losses. The resulting widening would thus adapt, to some extent, to the abstract description chain being traversed.

It is worth noticing that the framework presented in this paper is indeed quite general. It is based on the specification of a *computational ordering* [14] that satisfies the ascending chain condition. Then a preexisting widening and any finite set of extrapolation heuristics are combined so as to ensure that the combination turns every ascending chain (with respect to the *approximation ordering*) into a chain for the computational ordering. We have shown that this is sufficient to ensure that the combination defines a widening. Moreover, we have also shown that the new widening so obtained is never less precise than the preexisting widening: this is an important feature for those cases, such as the one of convex polyhedra, where the preexisting widening has proved its adequacy on a number of different applications. This general idea is exploited in [40] to obtain widenings for finite powerset domains (i.e., particular refinements of an abstract domain that allow for the exact representation of finite disjunctions), and we expect it can be successfully adopted for any abstract domain.

Acknowledgments. We would like to express our gratitude to Frédéric Besson for his useful comments and observations on the ideas sketched in [3]; Fred Mesnard for the information and the discussions we had with him about the impact of precision on termination inference for Prolog programs; and the reviewers of [24] for their careful comments that helped us improve the paper.

References

[1] P. Cousot, N. Halbwachs, Automatic discovery of linear restraints among variables of a program, in: Conference Record of the Fifth Annual ACM Symposium on Principles of Programming Languages, ACM Press, Tucson, Arizona, 1978, pp. 84–96.

[2] P. Cousot, 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, ACM Press, New York, 1977, pp. 238–252.

[3] F. Besson, T. P. Jensen, J.-P. Talpin, Polyhedral analysis for synchronous languages, in: A. Cortesi, G. Filé (Eds.), Static Analysis: Proceedings of the 6th International Symposium, Vol. 1694 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, Venice, Italy, 1999, pp. 51–68.

[4] N. Halbwachs, Delay analysis in synchronous programs, in: C. Courcoubetis (Ed.), Computer Aided Verification: Proceedings of the 5th International Conference, Vol. 697 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, Elounda, Greece, 1993, pp. 333–346.

[5] N. Halbwachs, Y.-E. Proy, P. Raymond, Verification of linear hybrid systems by means of convex approximations, in: B. Le Charlier (Ed.), Static Analysis:

Proceedings of the 1st International Symposium, Vol. 864 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, Namur, Belgium, 1994, pp. 223–237.

[6] T. A. Henzinger, P.-H. Ho, H. Wong-Toi, HyTech: A model checker for hybrid systems, Software Tools for Technology Transfer 1 (1+2) (1997) 110–122.

[7] Z. Manna, N. S. Bjørner, A. Browne, M. Colón, B. Finkbeiner, M. Pichora, H. B. Sipma, T. E. Uribe, An update on STeP: Deductive-algorithmic verification of reactive systems, in: R. Berghammer, Y. Lakhnech (Eds.), Tool Support for System Specification, Development and Verification, Advances in Computing Sciences, Springer-Verlag, Berlin, 1999, pp. 174–188.

[8] F. Benoy, A. King, Inferring argument size relationships with CLP($\mathcal{R}$), in: J. P. Gallagher (Ed.), Logic Program Synthesis and Transformation: Proceedings of the 6th International Workshop, Vol. 1207 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, Stockholm, Sweden, 1997, pp. 204–223.

[9] P. M. Benoy, Polyhedral domains for abstract interpretation in logic programming, Ph.D. thesis, Computing Laboratory, University of Kent, Canterbury, Kent, UK (Jan. 2002).

[10] W. Pugh, A practical algorithm for exact array dependence analysis, Communications of the ACM 35 (8) (1992) 102–114.

[11] N. Dor, M. Rodeh, S. Sagiv, Cleanness checking of string manipulations in C programs via integer analysis, in: Cousot [41], pp. 194–212.

[12] M. A. Colón, H. B. Sipma, Synthesis of linear ranking functions, in: T. Margaria, W. Yi (Eds.), Tools and Algorithms for Construction and Analysis of Systems, 7th International Conference, TACAS 2001, Vol. 2031 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, Genova, Italy, 2001, pp. 67–81.

[13] P. Cousot, R. Cousot, Static determination of dynamic properties of programs, in: B. Robinet (Ed.), Proceedings of the Second International Symposium on Programming, Dunod, Paris, France, 1976, pp. 106–130.

[14] P. Cousot, R. Cousot, Abstract interpretation frameworks, Journal of Logic and Computation 2 (4) (1992) 511–547.

[15] P. Cousot, R. Cousot, Comparing the Galois connection and widening/narrowing approaches to abstract interpretation, in: M. Bruynooghe, M. Wirsing (Eds.), Proceedings of the 4th International Symposium on Programming Language Implementation and Logic Programming, Vol. 631 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, Leuven, Belgium, 1992, pp. 269–295.

[16] P. Cousot, Semantic foundations of program analysis, in: S. S. Muchnick, N. D. Jones (Eds.), Program Flow Analysis: Theory and Applications, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1981, Ch. 10, pp. 303–342.

[17] N. Halbwachs, Détermination automatique de relations linéaires vérifiées par les variables d'un programme, Thèse de 3$^{\text{ème}}$ cycle d'informatique, Université scientifique et médicale de Grenoble, Grenoble, France (Mar. 1979).

[18] T. A. Henzinger, P.-H. Ho, A note on abstract interpretation strategies for hybrid automata, in: P. J. Antsaklis, W. Kohn, A. Nerode, S. Sastry (Eds.), Hybrid Systems II, Vol. 999 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, 1995, pp. 252–264.

[19] T. A. Henzinger, J. Preussig, H. Wong-Toi, Some lessons from the HYTECH experience, in: Proceedings of the 40th Annual Conference on Decision and Control, IEEE Computer Society Press, 2001, pp. 2887–2892.

[20] T. Bultan, R. Gerber, W. Pugh, Model-checking concurrent systems with unbounded integer variables: Symbolic representations, approximations, and experimental results, ACM Transactions on Programming Languages and Systems 21 (4) (1999) 747–789.

[21] G. Delzanno, A. Podelski, Model checking in CLP, in: R. Cleaveland (Ed.), Tools and Algorithms for Construction and Analysis of Systems, 5th International Conference, TACAS '99, Vol. 1579 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, Amsterdam, The Netherlands, 1999, pp. 223–239.

[22] F. Mesnard, U. Neumerkel, Applying static analysis techniques for inferring termination conditions of logic programs, in: Cousot [41], pp. 93–110.

[23] N. Halbwachs, Y.-E. Proy, P. Roumanoff, Verification of real-time systems using linear relation analysis, Formal Methods in System Design 11 (2) (1997) 157–185.

[24] R. Bagnara, P. M. Hill, E. Ricci, E. Zaffanella, Precise widening operators for convex polyhedra, in: R. Cousot (Ed.), Static Analysis: Proceedings of the 10th International Symposium, Vol. 2694 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, San Diego, California, USA, 2003, pp. 337–354.

[25] N. Dershowitz, Z. Manna, Proving termination with multiset orderings, Communications of the ACM 22 (8) (1979) 465–476.

[26] T. S. Motzkin, H. Raiffa, G. L. Thompson, R. M. Thrall, The double description method, in: H. W. Kuhn, A. W. Tucker (Eds.), Contributions to the Theory of Games – Volume II, no. 28 in Annals of Mathematics Studies, Princeton University Press, Princeton, New Jersey, 1953, pp. 51–73.

[27] H. Le Verge, A note on Chernikova's algorithm, *Publication interne* 635, IRISA, Campus de Beaulieu, Rennes, France (1992).

[28] N. V. Chernikova, Algorithm for finding a general formula for the non-negative solutions of system of linear equations, U.S.S.R. Computational Mathematics and Mathematical Physics 4 (4) (1964) 151–158.

[29] N. V. Chernikova, Algorithm for finding a general formula for the non-negative solutions of system of linear inequalities, U.S.S.R. Computational Mathematics and Mathematical Physics 5 (2) (1965) 228–233.

[30] N. V. Chernikova, Algorithm for discovering the set of all solutions of a linear programming problem, U.S.S.R. Computational Mathematics and Mathematical Physics 8 (6) (1968) 282–293.

[31] D. K. Wilde, A library for doing polyhedral operations, Master's thesis, Oregon State University, Corvallis, Oregon, also published as IRISA *Publication interne* 785, Rennes, France, 1993 (Dec. 1993).

[32] V. Loechner, *PolyLib*: A library for manipulating parameterized polyhedra, Available at `http://icps.u-strasbg.fr/~loechner/polylib/`, declares itself to be a continuation of [31] (Mar. 1999).

[33] B. Jeannet, Convex Polyhedra Library, release 1.1.3c Edition, documentation of the "New Polka" library available at `http://www.irisa.fr/prive/Bertrand.Jeannet/newpolka.html` (Mar. 2002).

[34] R. Bagnara, P. M. Hill, E. Ricci, E. Zaffanella, The Parma Polyhedra Library User's Manual, Department of Mathematics, University of Parma, Parma, Italy, release 0.5 Edition, available at `http://www.cs.unipr.it/ppl/` (Apr. 2003).

[35] R. Bagnara, E. Ricci, E. Zaffanella, P. M. Hill, Possibly not closed convex polyhedra and the Parma Polyhedra Library, in: M. V. Hermenegildo, G. Puebla (Eds.), Static Analysis: Proceedings of the 9th International Symposium, Vol. 2477 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, Madrid, Spain, 2002, pp. 213–229.

[36] T. H. Cormen, T. E. Leiserson, R. L. Rivest, Introduction to Algorithms, The MIT Press, Cambridge, MA, 1990.

[37] R. Bagnara, Data-flow analysis for constraint logic-based languages, Ph.D. thesis, Dipartimento di Informatica, Università di Pisa, Pisa, Italy, printed as Report TD-1/97 (Mar. 1997).

[38] F. Bourdoncle, Efficient chaotic iteration strategies with widenings, in: D. Bjørner, M. Broy, I. V. Pottosin (Eds.), Proceedings of the International Conference on "Formal Methods in Programming and Their Applications", Vol. 735 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, Academgorodok, Novosibirsk, Russia, 1993, pp. 128–141.

[39] F. Bourdoncle, Sémantiques des langages impératifs d'ordre supérieur et interprétation abstraite, PRL Research Report 22, DEC Paris Research Laboratory (1993).

[40] R. Bagnara, P. M. Hill, E. Zaffanella, Widening operators for powerset domains, in: G. Levi, B. Steffen (Eds.), Proceedings of the Fifth International Conference on Verification, Model Checking and Abstract Interpretation, Lecture Notes in Computer Science, Springer-Verlag, Berlin, Venice, Italy, 2004, to appear.

[41] P. Cousot (Ed.), Static Analysis: 8th International Symposium, SAS 2001, Vol. 2126 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, Paris, France, 2001.