Software Tools for Technology Transfer manuscript No.
(will be inserted by the editor)

Widening Operators for Powerset Domains*

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

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

The date of receipt and acceptance will be inserted by the editor

Abstract. The *finite powerset construction* upgrades an abstract domain by allowing for the representation of finite disjunctions of its elements. While most of the operations on the finite powerset abstract domain are easily obtained by "lifting" the corresponding operations on the base-level domain, the problem of endowing finite powersets with a provably correct widening operator is still open. In this paper we define three generic widening methodologies for the finite powerset abstract domain. The widenings are obtained by lifting any widening operator defined on the base-level abstract domain and are parametric with respect to the specification of a few additional operators that allow all the flexibility required to tune the complexity/precision trade-off. As far as we know, this is the first time that the problem of deriving non-trivial, provably correct widening operators in a domain refinement is tackled successfully. We illustrate the proposed techniques by instantiating our widening methodologies on powersets of convex polyhedra, a domain for which no non-trivial widening operator was previously known.

Key words: Abstract interpretation, abstract domain refinement, powersets, widening operators, convex polyhedra.

1 Introduction

The design and implementation of effective, expressive and efficient abstract domains for data-flow analysis and model-checking is a very difficult task. For this reason, starting with [16], there continues to be strong interest in techniques that derive enhanced abstract domains by applying systematic constructions on simpler, existing domains. Disjunctive completion, direct product, reduced product and reduced power are the first and most famous constructions of this kind [16]; several variations of them as well as others constructions have been proposed in the literature.

Once the carrier of the enhanced abstract domain has been obtained by one of these systematic constructions, the abstract operations can be defined, as usual, as the optimal approximations of the concrete ones. While this completely solves the specification problem, it usually leaves the implementation problem with the designer and gives no guarantees about the efficiency (or even the computability) of the resulting operations. This motivates the importance of generic techniques whereby correct, even though not necessarily optimal, domain operations are derived automatically or semi-automatically from those of the domains the construction operates upon [13, 16, 23].

When an abstract domain has very long or infinite ascending chains, the standard abstract iteration sequence [15]

$$\mathcal{A}^0(\mathbf{0}), \mathcal{A}^1(\mathbf{0}), \mathcal{A}^2(\mathbf{0}), \dots, \tag{1}$$

where $\mathcal{A}$ is the abstract semantic function and $\mathbf{0}$ is the bottom element of the domain, may converge very slowly or fail to converge altogether. This problem can be attacked by resorting to a binary operator '∇' and defining an alternate abstract semantic function $\mathcal{B}$ such that, for each domain element d, $\mathcal{B}(d) := d \nabla \bigl(d \oplus \mathcal{A}(d)\bigr)$, where '$\oplus$' is the least upper bound operator of the domain. The

* 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," "Constraint Based Verification of Reactive Systems," and "AIDA — Abstract Interpretation: Design and Applications," and by a Royal Society (UK) International Joint Project (ESEP) award.

following sequence is then computed instead of (1):

$$\mathcal{B}^0(\mathbf{0}), \mathcal{B}^1(\mathbf{0}), \mathcal{B}^2(\mathbf{0}), \ldots. \tag{2}$$

If '∇' always results in an upper bound of its operands, then it is called *extrapolation operator* and the elements of the sequence (2) are upper approximations of the corresponding elements in (1). If, in addition, '∇' ensures that (for each possible choice of the $\mathcal{A}$ we started with) the sequence (2) is ultimately stationary, then the (finitely computable) fixpoint of $\mathcal{B}$ is a post-fixpoint of $\mathcal{A}$ and '∇' is called *widening operator*.

This paper focuses on the derivation of widening operators for a kind of disjunctive refinement we call *finite powerset construction* [2], in which finite disjunctions are implemented by explicit collections of elements of the base-level abstract domain. The enhanced abstract domain obtained by means of this construction contains ascending chains whose cardinality is greater than or equal to the cardinality of the base-level abstract domain. As a result, every large or infinite abstract domain, when enhanced by means of the finite powerset construction, results in a domain with long or infinite ascending chains whose practicality is thus dependent on the availability of suitable widening operators.

It should be stressed that what we are facing here is a dichotomy. On the one hand, the design of a successful widening is a very delicate task that is not only dependent on the considered abstract domain but also on the particular analysis or verification application *and* on the class of systems being analyzed or verified. A widening is successful for a class of problem instances to the extent it captures common growth patterns that do happen in practice in the class. Experimentations is all what is available today to evaluate the goodness of a widening with respect to its potential applications. On the other hand, the extreme specificity of the widening design problem must be contrasted with the complete generality of the finite powerset construction. The important contribution of this paper is constituted by three methodologies for the design of widening operators on *any* finite powerset domain. These methodologies make it very easy to ensure that the resulting operator is indeed a widening (a non-trivial problem, as witnessed by the fact that previous attempts at defining a widening on powersets of polyhedra have failed) while leaving enough latitude to the designer to attack the precision problem in a domain-dependent, problem-dependent way. As far as we know, this is the first time that the problem of deriving non-trivial, provably correct widening operators in a domain refinement is tackled successfully. In previous works, such as [10], the main focus was in the definition of extrapolation operators:[1] the given examples finitely converge thanks to the adoption of *ad-hoc* operators and no general methodology is suggested that can turn these extrapolators into proper widenings.

In this paper, we also present specializations of our widening methodologies to finite powersets of convex polyhedra. Not only is this included to help the reader gain a better intuition regarding the underlying approach but also to provide a definitely non-toy instance that is practically useful for applications such as data-flow analysis and model checking. Sets of polyhedra are implemented in Polylib [30,35] and its successor *PolyLib* [31], even though no widenings are provided. Sets of polyhedra, represented with Presburger formulas made available by the Omega library [29,32], are used in the verifier described in [11]; there, an extrapolation operator (i.e., a widening without convergence guarantee) on sets of polyhedra is described. Another extrapolation operator is implemented in the automated verification tool described in [21], where sets of polyhedra are represented using the clp(q, r) constraint library [28].

The rest of the paper is structured as follows: Section 2 recalls the basic concepts and notations needed in this paper; Section 3 defines the finite powerset construction as a disjunctive refinement for any abstract domain that is a join-semilattice; Section 4 shows the divergence problems that arise when upgrading any widening for the base-level domain to work on the finite powerset domain; Sections 5, 6 and 7 give three alternative strategies for solving these issues so as to obtain proper widenings for the finite powerset domain; Section 8 shows a way to control the precision/efficiency trade-off of these widenings. Section 9 concludes. Appendix A contains the proofs of all the stated results.

2 Preliminaries

For a set S, $\wp(S)$ is the powerset of S, whereas $\wp_{\mathrm{f}}(S)$ is the set of all the *finite* subsets of S; the cardinality of S is denoted by $\# S$. The set of all the finite multisets having elements in S is denoted by $\mathcal{M}(S)$. The operators working on multisets are denoted by the corresponding operators working on sets: any ambiguity will be resolved by context. The set of all the *partial* (resp., *total*) functions from set S to set T is denoted by $S \rightarrowtail T$ (resp., $S \rightarrow T$). The first limit ordinal is denoted by ω. A *poset* $\langle \mathcal{O}, \preceq \rangle$ is a set $\mathcal{O}$ equipped with a partial order '$\preceq$'. The strict version of the partial order relation is denoted by '$\prec$'. Each poset $\langle \mathcal{O}, \preceq \rangle$ induces a corresponding poset of finite multisets $\bigl\langle \mathcal{M}(\mathcal{O}), \mathrel{\underline{\ll}} \bigr\rangle$ where the *multiset partial order* '$\mathrel{\underline{\ll}}$' is defined, for all $M, N \in \mathcal{M}(\mathcal{O})$, as follows [22]:

$$M \mathrel{\underline{\ll}} N \overset{\mathrm{def}}{\iff} \begin{cases} \exists X, Y \in \mathcal{M}(\mathcal{O}) \mathrel{.} N = (M \setminus X) \cup Y \\ \text{and } \forall y \in Y : \exists x \in X \mathrel{.} x \prec y. \end{cases}$$

A *chain* over the poset $\langle \mathcal{O}, \preceq \rangle$ is a totally ordered subset $C \subseteq \mathcal{O}$, i.e., for each $x, y \in C$ we have $x \preceq y$ or

[1] In [10], a "widening" is any extrapolation operator; the finite convergence guarantee is only achieved when considering so-called *tractable* representations.

$y \preceq x$. A poset satisfies the *ascending chain condition* if all its strictly increasing chains are finite. The induced poset of finite multisets $\bigl\langle \mathcal{M}(\mathcal{O}), \mathrel{\underline{\prec\!\!\prec}} \bigr\rangle$ satisfies the ascending chain condition if and only if the base-level poset $\langle \mathcal{O}, \preceq \rangle$ does [22].[2] When the carrier of the relation is made clear from context, we will abuse terminology by saying that (the strict version of) a partial order relation satisfies the ascending chain condition to mean that the corresponding poset satisfies this property.

We assume some familiarity with the basic notions of lattice and fixpoint theory [9].

2.1 Abstract Interpretation

In the literature, several abstract interpretation frameworks have been proposed that are able to establish a formal relationship between the behaviors of programs when observed at different levels of abstraction. The main difference between these frameworks usually concerns the trade-off between their general applicability and the strength of the formal results that can be established. In this paper we will adopt the framework proposed in [18, Section 7], where the correspondence between the concrete and the abstract domains is induced from a concrete approximation relation and a concretization function. Since we are not aiming at maximum generality, for the sole purpose of simplifying the presentation, we will consider a particular instance of the framework by assuming a few additional but non-essential domain properties. The resulting construction will be adequate for our purposes, since it still allows for algebraically weak abstract domains.

The concrete domain is modeled as a complete lattice of semantic properties $\langle C, \sqsubseteq, \bot, \top, \sqcup, \sqcap \rangle$; as usual, the concrete approximation relation $c_1 \sqsubseteq c_2$ holds if c_1 is a stronger property than c_2 (i.e., c_2 approximates c_1). The concrete semantics $c \in C$ of a program is formalized as the least fixpoint of a continuous (concrete) semantic function $\mathcal{F} \colon C \to C$, which is iteratively computed starting from the bottom element, so that

$$c = \mathcal{F}^{\omega}(\bot) := \bigsqcup_{\delta < \omega} \mathcal{F}^{\delta}(\bot).$$

The abstract domain $\hat{D} = \langle D, \vdash, \mathbf{0}, \oplus \rangle$ is modeled as a join-semilattice (i.e., the least upper bound $d_1 \oplus d_2$ exists for all $d_1, d_2 \in D$). We will overload '$\oplus$' so that, for each $S \in \wp_{\mathrm{f}}(D)$, $\bigoplus S$ denotes the least upper bound of S. The abstract domain $\hat{D}$ is related to the concrete domain by a monotonic and injective concretization function $\gamma \colon D \to C$. Monotonicity and injectivity mean that the abstract partial order '$\vdash$' is indeed the approximation relation induced on D by the concretization function γ. For all $d_1, d_2 \in D$, we will use the notation $d_1 \Vdash d_2$ to mean that $d_1 \vdash d_2$ and $d_1 \neq d_2$. We assume the existence of a monotonic abstract semantic function $\mathcal{A} \colon D \to D$ that is sound with respect to $\mathcal{F} \colon C \to C$:

$$\forall c \in C : \forall d \in D : c \sqsubseteq \gamma(d) \implies \mathcal{F}(c) \sqsubseteq \gamma\bigl(\mathcal{A}(d)\bigr). \quad (3)$$

This local correctness condition ensures that each concrete iterate can be safely approximated by computing the corresponding abstract iterate (starting from the bottom element $\mathbf{0} \in D$). However, due to the weaker algebraic properties satisfied by the abstract domain, the abstract upward iteration sequence $\mathcal{A}^0(\mathbf{0})$, $\mathcal{A}^1(\mathbf{0})$, ..., may not converge. Even when it converges, it may fail to do so in a finite number of steps, therefore being useless for the purposes of static analysis.

Widening operators [14,15,18,19] provide a simple and general characterization for enforcing and accelerating convergence. We will adopt a minor variation of the classical definition of widening operator (see footnote 6 in [19, p. 275]).

Definition 1. (Widening.) Let $\langle D, \vdash, \mathbf{0}, \oplus \rangle$ be a join-semilattice. The partial operator $\nabla \colon D \times D \rightarrowtail D$ is a *widening* if

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

Any widening operator '∇' induces a corresponding partial ordering '$\vdash_\nabla$' on the domain D; this is defined as the reflexive and transitive closure of the relation

$$\bigl\{\, (d_1, d) \in D \times D \bigm| \exists d_2 \in D \mathrel{.} d_1 \Vdash d_2 \land d = d_1 \nabla d_2 \,\bigr\}.$$

The relation '$\vdash_\nabla$' satisfies the ascending chain condition. We write $d_1 \Vdash_\nabla d$ to denote $d_1 \vdash_\nabla d$ and $d_1 \neq d$.

It can be proved that the *upward iteration sequence with widenings* starting at the bottom element $d_0 := \mathbf{0}$ and defining the rest by

$$d_{i+1} := \begin{cases} d_i, & \text{if } \mathcal{A}(d_i) \vdash d_i, \\ d_i \nabla \bigl(d_i \oplus \mathcal{A}(d_i)\bigr), & \text{otherwise,} \end{cases}$$

converges after a finite number $j \in \mathbb{N}$ of iterations [19]. Note that the widening is only applied to arguments d_i and $d'_i = d_i \oplus \mathcal{A}(d_i)$ satisfying $d_i \Vdash d'_i$. Also, when condition (3) holds, the post-fixpoint $d_j \in D$ of $\mathcal{A}$ is a correct approximation of the concrete semantics, i.e., $\mathcal{F}^{\omega}(\bot) \sqsubseteq \gamma(d_j)$.

When trying to prove that an upper bound operator $\boxplus \colon D \times D \to D$ is indeed a widening, a possible tactic is to provide a "convergence certificate." This is constituted by a structure that disallows indefinite growth and a way of mapping elements of D to elements of the structure, such that the application of the upper bound operator results in a strict growth: as growth cannot be

[2] Note the systematic replacement of the notion of *well-founded* poset, adopted in [22], by the dual notion of poset satisfying the ascending chain condition, which is more standard in the field of Abstract Interpretation.

indefinite, convergence is certified. Formally, a *finite convergence certificate* for '$\boxplus$' (on $\hat{D}$) is a triple $(\mathcal{O}, \preceq, \mu)$ where $\langle \mathcal{O}, \preceq \rangle$ is a poset satisfying the ascending chain condition and $\mu \colon D \to \mathcal{O}$, which is called *level mapping*, is such that

$$\forall d_1, d_2 \in D : d_1 \Vdash d_2 \implies \mu(d_1) \prec \mu(d_1 \boxplus d_2).$$

We will abuse notation by writing 'μ' to denote the certificate $(\mathcal{O}, \preceq, \mu)$.

2.2 The Abstract Domain of Polyhedra

In this section, we instantiate the abstract interpretation framework sketched above by presenting the well-known abstract domain of closed convex polyhedra. This domain will be used throughout the paper to illustrate the generic widening techniques that will be defined.

Let $\mathbb{R}^n$, where $n > 0$, be the n-dimensional real vector space. The set $\mathcal{P} \subseteq \mathbb{R}^n$ is a *closed and convex polyhedron* (*polyhedron*, for short) if and only if $\mathcal{P}$ can be expressed as the intersection of a finite number of closed affine half-spaces of $\mathbb{R}^n$. The set $\mathbb{CP}_n$ of closed convex polyhedra on $\mathbb{R}^n$, when partially ordered by subset inclusion, is a lattice having the empty set and $\mathbb{R}^n$ as the bottom and top elements, respectively; the binary meet operation is set-intersection, whereas the binary join operation, denoted by '$\uplus$', is called *convex polyhedral hull* (*poly-hull*, for short). Therefore, we have the abstract domain

$$\widehat{\mathbb{CP}}_n := \langle \mathbb{CP}_n, \subseteq, \varnothing, \mathbb{R}^n, \uplus, \cap \rangle.$$

This domain can be related to several concrete domains, depending on the intended application. One example of a concrete domain is the complete lattice

$$\hat{\mathsf{A}}_n := \big\langle \wp(\mathbb{R}^n), \subseteq, \varnothing, \mathbb{R}^n, \cup, \cap \big\rangle.$$

Note that $\widehat{\mathbb{CP}}_n$ is a meet-sublattice of $\hat{\mathsf{A}}_n$, sharing the same bottom and top elements. Another example of concrete domain is the complete lattice

$$\hat{\mathsf{B}}_n := \big\langle \wp(\mathbb{CP}_n), \subseteq, \varnothing, \mathbb{CP}_n, \cup, \cap \big\rangle.$$

The domain $\hat{\mathsf{A}}_n$ is the one commonly adopted in semantic constructions for imperative programs, whereas the domain $\hat{\mathsf{B}}_n$ is useful when modelling those constraint programming languages where affine inequalities (i.e., convex polyhedra) can be used as a first class datatype.

The abstract domain $\widehat{\mathbb{CP}}_n$, which is a join-semilattice, is related to the domains $\hat{\mathsf{A}}_n$ and $\hat{\mathsf{B}}_n$ by the concretization functions $\gamma^{\mathsf{A}} \colon \mathbb{CP}_n \to \wp(\mathbb{R}^n)$ and $\gamma^{\mathsf{B}} \colon \mathbb{CP}_n \to \wp(\mathbb{CP}_n)$ defined as follows, for each $\mathcal{P} \in \mathbb{CP}_n$:

$$\gamma^{\mathsf{A}}(\mathcal{P}) := \mathcal{P}, \tag{4}$$

$$\gamma^{\mathsf{B}}(\mathcal{P}) := {\downarrow} \mathcal{P} := \{\, \mathcal{Q} \in \mathbb{CP}_n \mid \mathcal{Q} \subseteq \mathcal{P} \,\}. \tag{5}$$

These functions are trivially monotonic and injective.

For each choice of concrete domain carrier C, that is $C \in \big\{\wp(\mathbb{R}^n), \wp(\mathbb{CP}_n)\big\}$, the continuous semantic function $\mathcal{F} \colon C \to C$ and the corresponding monotonic abstract semantic function $\mathcal{A} \colon \mathbb{CP}_n \to \mathbb{CP}_n$, which is assumed to be correct, are deliberately left unspecified. The domain $\widehat{\mathbb{CP}}_n$ contains infinite ascending chains having no least upper bound in $\mathbb{CP}_n$. Thus, the convergence of the abstract iteration sequence has to be guaranteed by the adoption of widening operators.

2.3 Widening the Polyhedral Domain

The first widening on polyhedra was introduced in [20] and refined in [25]. This operator, denoted by '∇_s', has been termed *standard widening* and used almost universally. Its formal specification requires some further notation and concepts related to the domain of polyhedra.

Any vector $\mathbf{v} \in \mathbb{R}^n$ is regarded as a matrix in $\mathbb{R}^{n \times 1}$ so that it can be manipulated with the usual matrix operations of addition, multiplication (both by a scalar and by another matrix), and transposition, which is denoted by $\mathbf{v}^{\mathrm{T}}$. For each $i = 1, \ldots\ n$, the i-th component of the vector $\mathbf{v} \in \mathbb{R}^n$ is denoted by v_i. The *scalar product* of $\mathbf{v}, \mathbf{w} \in \mathbb{R}^n$, denoted $\langle \mathbf{v}, \mathbf{w} \rangle$, is $\mathbf{v}^{\mathrm{T}}\mathbf{w} = \sum_{i=1}^{n} v_i w_i$. The vector of $\mathbb{R}^n$ having all components equal to zero is denoted by $\mathbf{0}$.

Let $V = \{\mathbf{v}_1, \ldots, \mathbf{v}_m\} \subseteq \mathbb{R}^n$ be a finite set of vectors. The vectors in V are said *affinely independent* if the only solution of the system of equations $\big\{\sum_{i=1}^{m} \lambda_i \mathbf{v}_i = \mathbf{0},\ \sum_{i=1}^{m} \lambda_i = 0\big\}$ is $\lambda_i = 0$, for each $i = 1, \ldots, m$. 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 $k - 1$.

For each vector $\mathbf{a} \in \mathbb{R}^n$ and scalar $b \in \mathbb{R}$, where $\mathbf{a} \neq \mathbf{0}$, the linear non-strict inequality constraint $\langle \mathbf{a}, \mathbf{x} \rangle \geq b$ defines a topologically closed affine half-space of $\mathbb{R}^n$. The linear equality constraint $\langle \mathbf{a}, \mathbf{x} \rangle = b$ defines an affine hyperplane of $\mathbb{R}^n$ (i.e., the intersection of the affine half-spaces $\langle \mathbf{a}, \mathbf{x} \rangle \geq b$ and $\langle -\mathbf{a}, \mathbf{x} \rangle \geq -b$). 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. Thus, each polyhedron $\mathcal{P}$ can be represented by a finite system of linear equality and non-strict inequality constraints $\mathcal{C}$ and we write $\mathcal{P} = \mathrm{con}(\mathcal{C})$. The subsets of equality and inequality constraints in $\mathcal{C}$ are denoted by $\mathrm{eq}(\mathcal{C})$ and $\mathrm{ineq}(\mathcal{C})$, respectively. When $\mathcal{P} = \mathrm{con}(\mathcal{C}) \neq \varnothing$, we say that $\mathcal{C}$ is in *minimal form* if and only if $\#\,\mathrm{eq}(\mathcal{C}) = n - \dim(\mathcal{P})$ and there does not exist $\mathcal{C}' \subset \mathcal{C}$ such that $\mathrm{con}(\mathcal{C}') = \mathcal{P}$. All constraint systems in minimal form describing a given polyhedron have the same cardinality.

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

$$\mathrm{repr}_{\geq}(\mathcal{C}) := \Bigl\{ \langle -\mathbf{a}, \mathbf{x} \rangle \geq -b \Bigm| \bigl(\langle \mathbf{a}, \mathbf{x} \rangle = b\bigr) \in \mathcal{C} \Bigr\} \cup \Bigl\{ \langle \mathbf{a}, \mathbf{x} \rangle \geq b \Bigm| \bigl(\langle \mathbf{a}, \mathbf{x} \rangle \geq b\bigr) \in \mathcal{C} \text{ or } \bigl(\langle \mathbf{a}, \mathbf{x} \rangle = b\bigr) \in \mathcal{C} \Bigr\}.$$

Definition 2. (Standard widening.) For $i = 1, 2$, let $\mathcal{P}_i = \mathrm{con}(\mathcal{C}_i) \in \mathbb{CP}_n$, where the constraint system $\mathcal{C}_1$ is either inconsistent or in minimal form. Then, the polyhedron $\mathcal{P}_1 \nabla_s \mathcal{P}_2 \in \mathbb{CP}_n$ is defined as

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

where

$$\mathcal{C}'_1 := \Bigl\{ \beta_1 \in \mathrm{repr}_{\geq}(\mathcal{C}_1) \Bigm| \mathcal{P}_2 \subseteq \mathrm{con}\bigl(\{\beta_1\}\bigr) \Bigr\},$$

$$\mathcal{C}'_2 := \left\{ \beta_2 \in \mathrm{repr}_{\geq}(\mathcal{C}_2) \;\middle|\; \begin{array}{l} \exists \beta_1 \in \mathrm{repr}_{\geq}(\mathcal{C}_1) \text{ .} \\ \mathcal{P}_1 = \mathrm{con}\bigl(\mathrm{repr}_{\geq}(\mathcal{C}_1)[\beta_2/\beta_1]\bigr) \end{array} \right\}$$

and $\mathrm{repr}_{\geq}(\mathcal{C}_1)[\beta_2/\beta_1] := \bigl(\mathrm{repr}_{\geq}(\mathcal{C}_1) \setminus \{\beta_1\}\bigr) \cup \{\beta_2\}$.

The constraints in $\mathcal{C}'_1$ are those that would have been selected when using the original proposal of [20], whereas the constraints in $\mathcal{C}'_2$ 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).

We now define a finite convergence certificate for the standard widening '∇_s': the basic intuition is that a standard widening application will either result in an increase of the dimension of the polyhedron (which happens when some equalities are turned into inequalities or dropped altogether) or, if the dimension is unchanged, in a decrease of the number of constraints.

Definition 3. ($(\mathcal{O}_s, \preceq_s, \mu_s)$.) Let $\mathcal{O}_s = (\mathbb{N}, \mathbb{N})$ and '$\preceq_s$' denote the lexicographic ordering for $\mathcal{O}_s$ that uses '$\geq$' for the individual ordering of the components. A certificate for '∇_s' is $(\mathcal{O}_s, \preceq_s, \mu_s)$ where the level mapping $\mu_s \colon \mathbb{CP}_n \to \mathcal{O}_s$, for each $\mathcal{P} = \mathrm{con}(\mathcal{C}) \in \mathbb{CP}_n$ such that $\mathcal{C}$ is either inconsistent or in minimal form, is defined by

$$\mu_s(\mathcal{P}) := \begin{cases} (n+1, 0), & \text{if } \mathcal{P} = \varnothing; \\ \bigl(n - \dim(\mathcal{P}), \# \mathcal{C}\bigr), & \text{otherwise.} \end{cases}$$

3 A Disjunctive Refinement

Traditionally, semantic domains have been designed incrementally by applying suitable domain constructors to basic components. In this respect, the theory of abstract interpretation makes no exception and systematic ways of composing or enhancing abstract domains have been proposed since [16]. In this section, we present the *finite powerset* operator [2], which is a domain refinement similar to disjunctive completion [16] and is obtained by a variant of the *down-set completion* construction presented in [17]. The following notation and definitions are mainly borrowed from [2, Section 6].

Definition 4. (Non-redundancy.) Let us consider a join-semilattice $\hat{D} = \langle D, \vdash, \mathbf{0}, \oplus \rangle$. The set $S \in \wp(D)$ is called *non-redundant* with respect to '$\vdash$' if and only if $\mathbf{0} \notin S$ and $\forall d_1, d_2 \in S : d_1 \vdash d_2 \implies d_1 = d_2$. The set of finite non-redundant subsets of D (with respect to '$\vdash$') is denoted by $\wp^{\vdash}_{\mathrm{fn}}(D)$. The reduction function $\Omega^{\vdash}_{D} \colon \wp_{\mathrm{f}}(D) \to \wp^{\vdash}_{\mathrm{fn}}(D)$ mapping a finite set into its non-redundant counterpart is defined, for each $S \in \wp_{\mathrm{f}}(D)$, by

$$\Omega^{\vdash}_{D}(S) := S \setminus \{\, d \in S \mid d = \mathbf{0} \text{ or } \exists d' \in S \mathrel{.} d \Vdash d' \,\}.$$

The restriction to the finite subsets reflects the fact that here we are mainly interested in an abstract domain where disjunctions are implemented by explicit collections of elements of the base-level abstract domain. As a consequence of this restriction, for any $S \in \wp_{\mathrm{f}}(D)$ such that $S \neq \{\mathbf{0}\}$, $\Omega^{\vdash}_{D}(S)$ is the (finite) set of the maximal elements of S.

Definition 5. (Finite powerset domain.) Let $\hat{D} = \langle D, \vdash, \mathbf{0}, \oplus \rangle$ be a join-semilattice. The *finite powerset domain* over $\hat{D}$ is the join-semilattice

$$\hat{D}_{\mathrm{P}} := \bigl\langle \wp^{\vdash}_{\mathrm{fn}}(D), \vdash_{\mathrm{P}}, \mathbf{0}_{\mathrm{P}}, \oplus_{\mathrm{P}} \bigr\rangle,$$

where $\mathbf{0}_{\mathrm{P}} := \varnothing$ and $S_1 \oplus_{\mathrm{P}} S_2 := \Omega^{\vdash}_{D}(S_1 \cup S_2)$.

The approximation ordering '$\vdash_{\mathrm{P}}$' induced by '$\oplus_{\mathrm{P}}$' is the Hoare powerdomain partial order [1], so that $S_1 \vdash_{\mathrm{P}} S_2$ if and only if

$$\forall d_1 \in S_1 : \exists d_2 \in S_2 \mathrel{.} d_1 \vdash d_2.$$

In abstract interpretation terms, this states that each element of S_1 is correctly approximated by some elements of S_2; hence S_1 is correctly approximated by S_2.

A sort of Egli-Milner partial order relation[3] will also be useful: $S_1 \vdash_{\mathrm{EM}} S_2$ holds if and only if either $S_1 = \mathbf{0}_{\mathrm{P}}$ or $S_1 \vdash_{\mathrm{P}} S_2$ and

$$\forall d_2 \in S_2 : \exists d_1 \in S_1 \mathrel{.} d_1 \vdash d_2.$$

This states that every element of S_2 serves the purpose of approximating some elements of S_1.

An *(Egli-Milner) connector* for $\hat{D}_{\mathrm{P}}$, denoted by '$\boxplus_{\mathrm{EM}}$' is any upper bound operator for the partial order '$\vdash_{\mathrm{EM}}$' on $\wp^{\vdash}_{\mathrm{fn}}(D)$. The reason we call such an operator a "connector" is because it will typically work by joining (connecting) elements in its arguments so as to ensure that all the resulting elements approximate an element in the arguments. Note that although a *least* upper bound for '$\vdash_{\mathrm{EM}}$' may not exist, a connector can always be defined; for instance, we can let $S_1 \boxplus_{\mathrm{EM}} S_2 := \bigl\{ \bigoplus (S_1 \cup S_2) \bigr\}$.

Besides the requirement on finiteness, another difference with respect to the down-set completion of [17] is that we are dropping the assumption about the complete

[3] Note that '$\vdash_{\mathrm{EM}}$' is similar to, but formally different from the partial order defined on the Egli-Milner powerdomain [1], since in its specification we consider the non-redundant elements only.

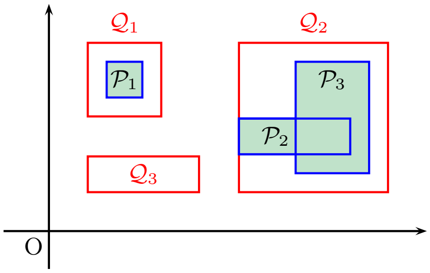

Fig. 1. The collection of $\mathcal{Q}_i$ approximates the collection of $\mathcal{P}_i$.

distributivity of the concrete domain. This is possible because our semantic domains are not necessarily related by Galois connections, so that this property does not have to be preserved.

The finite powerset domain is related to the concrete domain by means of the concretization function $\gamma_{\mathrm{P}} \colon \wp_{\mathrm{fn}}^{\vdash}(D) \to C$ defined by

$$\gamma_{\mathrm{P}}(S) := \bigsqcup \bigl\{ \gamma(d) \bigm| d \in S \bigr\}.$$

Note that γ_{P} is monotonic but not necessarily injective. For $S_1, S_2 \in \wp_{\mathrm{fn}}^{\vdash}(D)$, we write $S_1 \equiv_{\gamma_{\mathrm{P}}} S_2$ to denote that the two abstract elements actually denote the same concrete element, i.e., when $\gamma_{\mathrm{P}}(S_1) = \gamma_{\mathrm{P}}(S_2)$. It is easy to see that '$\equiv_{\gamma_{\mathrm{P}}}$' is a congruence relation on $\hat{D}_{\mathrm{P}}$. As noted in [17], non-redundancy only provides a partial, syntactic form of reduction. On the other hand, requiring the full, semantic form of reduction for a finite powerset domain can be computationally very expensive.

An abstract semantic function $\mathcal{A}_{\mathrm{P}} \colon \wp_{\mathrm{fn}}^{\vdash}(D) \to \wp_{\mathrm{fn}}^{\vdash}(D)$ on the finite powerset domain may be provided by an ad-hoc definition, which needs to be matched by a corresponding proof of correctness. More often, if the concrete semantic function $\mathcal{F} \colon C \to C$ satisfies suitable hypotheses, $\mathcal{A}_{\mathrm{P}}$ can be safely induced from the abstract semantic function $\mathcal{A} \colon D \to D$. For instance, if $\mathcal{F}$ is additive, we can define $\mathcal{A}_{\mathrm{P}}$ as follows [16,23]:

$$\mathcal{A}_{\mathrm{P}}(S) := \Omega_D^{\vdash}\Bigl(\bigl\{ \mathcal{A}(d) \bigm| d \in S \bigr\}\Bigr).$$

3.1 The Finite Powerset Domain of Polyhedra

The domain $(\widehat{\mathbb{CP}}_n)_{\mathrm{P}}$, having carrier the polyhedral domain $\wp_{\mathrm{fn}}^{\subseteq}(\mathbb{CP}_n)$, is the finite powerset domain over $\widehat{\mathbb{CP}}_n$. The approximation ordering induced by '$\subseteq$' is thus defined, for each $\mathcal{S}_1, \mathcal{S}_2 \in \wp_{\mathrm{fn}}^{\subseteq}(\mathbb{CP}_n)$, by

$$\mathcal{S}_1 \vdash_{\mathrm{P}} \mathcal{S}_2 \iff \forall \mathcal{P}_1 \in \mathcal{S}_1 : \exists \mathcal{P}_2 \in \mathcal{S}_2 \mathrel{.} \mathcal{P}_1 \subseteq \mathcal{P}_2.$$

Example 1. Consider the polyhedra in Figure 1 and let

$$\begin{aligned}
\mathcal{T}_0 &:= \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3\},\\
\mathcal{T}_1 &:= \{\mathcal{P}_1, \mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3\},\\
\mathcal{T}_2 &:= \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3\},\\
\mathcal{T}_3 &:= \{\mathcal{Q}_1, \mathcal{Q}_2\}.
\end{aligned}$$

Then $\mathcal{T}_0, \mathcal{T}_2, \mathcal{T}_3 \in \wp_{\mathrm{fn}}^{\subseteq}(\mathbb{CP}_2)$, but $\mathcal{T}_1 \notin \wp_{\mathrm{fn}}^{\subseteq}(\mathbb{CP}_2)$ since $\mathcal{P}_1 \subset \mathcal{Q}_1$ is redundant in $\mathcal{T}_1$, so that $\mathcal{T}_2 = \Omega_{\mathbb{CP}_2}^{\subseteq}(\mathcal{T}_1)$. Moreover, $\mathcal{T}_0 \vdash_{\mathrm{P}} \mathcal{T}_2$ but $\mathcal{T}_0 \nvdash_{\mathrm{EM}} \mathcal{T}_2$, because $\mathcal{Q}_3$ approximates none of the $\mathcal{P}_i$'s. Finally, $\mathcal{T}_0 \vdash_{\mathrm{EM}} \mathcal{T}_3$.

Let $\gamma_{\mathrm{P}}^{\mathsf{A}}$ and $\gamma_{\mathrm{P}}^{\mathsf{B}}$ denote the (powerset) concretization functions induced by the functions γ^{A} and γ^{B} defined by Eqs. (4) and (5), respectively. Then, the relation '$\equiv_{\gamma_{\mathrm{P}}^{\mathsf{A}}}$' makes two finite sets of polyhedra equivalent if and only if they have the same set-union. The general problem of deciding the semantic equivalence with respect to $\gamma_{\mathrm{P}}^{\mathsf{A}}$ of two finite (non-redundant) collections of polyhedra is known to be computationally hard [33]. On the other hand, since $\gamma_{\mathrm{P}}^{\mathsf{B}}$ is injective, '$\equiv_{\gamma_{\mathrm{P}}^{\mathsf{B}}}$' coincides with the identity congruence relation.

4 Extrapolation Operators on the Finite Powerset Domain

If the domain refinement of the previous section is meant to be used for static analysis, then a key ingredient that is still missing is a systematic way of ensuring the termination of the analysis. Following the spirit underlying the domain refinement methodology, one may try and *lift* any widening operator $\nabla \colon D \times D \rightarrowtail D$ defined on the base-level abstract domain $\hat{D}$ so as to be applied to elements of the finite powerset domain $\hat{D}_{\mathrm{P}}$.[4] Unfortunately, unless suitable counter-measures are taken, most of the lifting heuristics will break the convergence guarantee, resulting in an extrapolation operator for the finite powerset domain. Here we introduce a very general class of extrapolation operators lifting the base-level widening '∇'.

Definition 6. (Extrapolation heuristics.) An operator $h_{\mathrm{P}}^{\nabla} \colon \wp_{\mathrm{fn}}^{\vdash}(D) \times \wp_{\mathrm{fn}}^{\vdash}(D) \rightarrowtail \wp_{\mathrm{fn}}^{\vdash}(D)$ is an *extrapolation heuristics for* $\hat{D}_{\mathrm{P}}$ if, for all $S_1, S_2 \in \wp_{\mathrm{fn}}^{\vdash}(D)$ such that $S_1 \Vdash_{\mathrm{P}} S_2$, $h_{\mathrm{P}}^{\nabla}(S_1, S_2)$ is defined and satisfies the following conditions:

$$S_2 \vdash_{\mathrm{EM}} h_{\mathrm{P}}^{\nabla}(S_1, S_2); \tag{6}$$

$$\forall d \in h_{\mathrm{P}}^{\nabla}(S_1, S_2) \setminus S_2 : \exists d_1 \in S_1 \mathrel{.} d_1 \Vdash_{\nabla} d. \tag{7}$$

Informally, condition (6) ensures that the result is an upper approximation of S_2 in which every element covers at least one element of S_2 (i.e., the heuristics cannot add elements that are unrelated to S_2); condition (7) ensures that any element in the result that is not in S_2 must originate from an application of '∇' to an element of S_1.

It is straightforward to construct an algorithm for computing an extrapolation heuristics for any given base-level widening '∇'. The basic idea was proposed in [11]

[4] We assume the base-level abstract domain $\hat{D}$ is provided with at least one widening operator, as is the case for most abstract domains used in the context of static analysis. If $\hat{D}$ satisfies the ascending chain condition, so that it is not necessarily endowed with an explicit widening operator, then a dummy widening can be obtained by considering the least upper bound operator '$\oplus$'.

for an abstract domain encoding a set of integer vectors by means of a Presburger formula. Informally, for all pairs $(d_1, d_2) \in S_1 \times S_2$ that can be built using the two arguments S_1 and S_2, return $d_1 \nabla d_2$ if defined and return d_2 if not.

Definition 7. **(H_{P}^{∇}.)** For all $S_1, S_2 \in \wp_{\mathrm{fn}}^{\vdash}(D)$ such that $S_1 \vdash_{\mathrm{P}} S_2$, let $H_{\mathrm{P}}^{\nabla}(S_1, S_2) := S_2 \oplus_{\mathrm{P}} \Omega_D^{\vdash}(S)$ where

$$S := \{\, d_1 \nabla d_2 \in D \mid d_1 \in S_1, d_2 \in S_2, d_1 \Vdash d_2 \,\}.$$

Proposition 1. *The H_{P}^{∇} operator is an extrapolation heuristics for $\hat{D}_{\mathrm{P}}$.*

For the finite powerset domain over $\widehat{\mathbb{CP}}_n$, lines 10–15 of the algorithm specified in [11, Figure 8, page 773] provide an implementation of the heuristics H_{P}^{∇}, instantiated with the standard widening, '∇_s', on $\widehat{\mathbb{CP}}_n$.

Example 2. To see that the heuristics H_{P}^{∇} is not a widening for $(\widehat{\mathbb{CP}}_n)_{\mathrm{P}}$, consider the strictly increasing sequence $\mathcal{T}_0 \vdash_{\mathrm{P}} \mathcal{T}_1 \vdash_{\mathrm{P}} \cdots$ in $\mathbb{CP}_1$ defined by[5]

$$\mathcal{T}_j := \bigl\{\, \{x = i\} \bigm| i \in \mathbb{N}, 0 \leq i \leq j \,\bigr\}.$$

Then, no matter what the specification for '∇' is, we obtain $H_{\mathrm{P}}^{\nabla}(\mathcal{T}_j, \mathcal{T}_{j+1}) = \mathcal{T}_{j+1}$, for all $j \in \mathbb{N}$. Thus, the "widened" sequence is diverging.

The iteration sequence in Example 2 is diverging because there is no finite upper bound on the cardinality of the iterates. To overcome this problem, the operator sketched in [11], which uses the H_{P}^{∇} heuristics of Definition 7, assumes that a further approximation is applied whenever the cardinality of the set to be widened exceeds a fixed bound $k \in \mathbb{N}$. However, no matter how this further approximation step is defined, this approach is not enough to obtain a proper widening, so that termination cannot be guaranteed. In fact, the following example shows that the extrapolation heuristics H_{P}^{∇} may result in an infinite increasing sequence on $\hat{D}_{\mathrm{P}}$ whose elements all have bounded cardinality.

Example 3. Consider the extrapolation heuristics H_{P}^{∇} for $(\widehat{\mathbb{CP}}_2)_{\mathrm{P}}$, as specified in Definition 7, with '∇_s' as the widening on the base-level abstract domain $\widehat{\mathbb{CP}}_2$ (so that H_{P}^{∇} is the one used in [11]). Let $\mathcal{P}_0, \mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4 \in \mathbb{CP}_2$ be defined as

$$\begin{aligned}
\mathcal{P}_0 &= \{0 \leq x \leq 4,\ 0 \leq y \leq 4,\ x - y \leq 3,\ x + y \geq 1\},\\
\mathcal{P}_1 &= \{0 \leq x \leq 4,\ 0 \leq y \leq 4,\ x - y \leq 3\},\\
\mathcal{P}_2 &= \{0 \leq x \leq 4,\ 0 \leq y \leq 4\},\\
\mathcal{P}_3 &= \{0 \leq x \leq 8,\ 0 \leq y \leq 8,\ x + y \leq 14,\\
&\qquad x - y \geq -6,\ 5x - y \geq -2,\ x + 3y \geq 3\},\\
\mathcal{P}_4 &= \{0 \leq x \leq 8,\ 0 \leq y \leq 8,\ x + y \leq 14,\\
&\qquad x - y \geq -6,\ 4x - y \geq -3,\ x + 2y \geq 2\}.
\end{aligned}$$

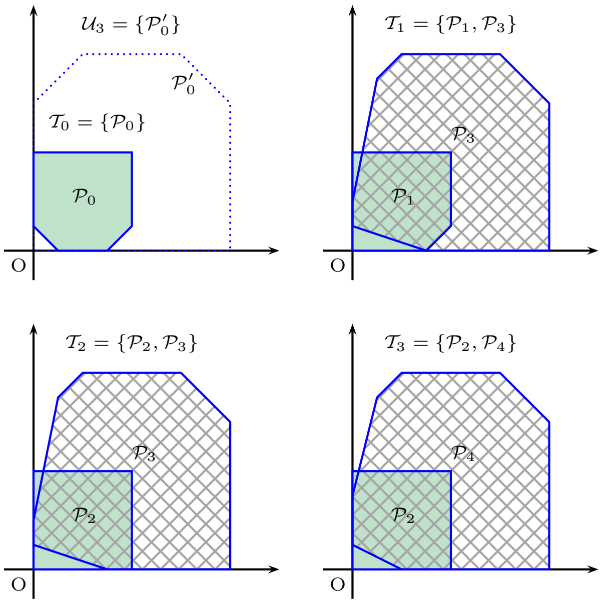
$\mathcal{U}_3 = \{\mathcal{P}'_0\}$
$\mathcal{P}'_0$
$\mathcal{T}_0 = \{\mathcal{P}_0\}$
$\mathcal{P}_0$
O
$\mathcal{T}_1 = \{\mathcal{P}_1, \mathcal{P}_3\}$
$\mathcal{P}_3$
$\mathcal{P}_1$
O
$\mathcal{T}_2 = \{\mathcal{P}_2, \mathcal{P}_3\}$
$\mathcal{P}_3$
$\mathcal{P}_2$
O
$\mathcal{T}_3 = \{\mathcal{P}_2, \mathcal{P}_4\}$
$\mathcal{P}_4$
$\mathcal{P}_2$
O

Fig. 2. The iterates $\mathcal{T}_0$, $\mathcal{T}_1$, $\mathcal{T}_2$, $\mathcal{T}_3$ and $\mathcal{U}_3$ in Example 3.

Note that $\mathcal{P}_1 = \mathcal{P}_0 \nabla_s \mathcal{P}_1$ and $\mathcal{P}_2 = \mathcal{P}_1 \nabla_s \mathcal{P}_2$; moreover, for all $i \in \{0, 1, 2\}$, we have $\mathcal{P}_i \not\subseteq \mathcal{P}_3$ and $\mathcal{P}_i \not\subseteq \mathcal{P}_4$.

Consider an increasing sequence $\mathcal{T}_0 \vdash_{\mathrm{P}} \mathcal{T}_1 \vdash_{\mathrm{P}} \mathcal{T}_2 \vdash_{\mathrm{P}} \mathcal{T}_3 \vdash_{\mathrm{P}} \ldots$ starting with elements

$$\begin{aligned}
\mathcal{T}_0 &= \{\mathcal{P}_0\},\\
\mathcal{T}_1 &= \{\mathcal{P}_1, \mathcal{P}_3\},\\
\mathcal{T}_2 &= \{\mathcal{P}_2, \mathcal{P}_3\},\\
\mathcal{T}_3 &= \{\mathcal{P}_2, \mathcal{P}_4\},
\end{aligned}$$

as shown in Figure 2. Then, the corresponding "widened" sequence $\mathcal{U}_0 \vdash_{\mathrm{P}} \mathcal{U}_1 \vdash_{\mathrm{P}} \mathcal{U}_2 \vdash_{\mathrm{P}} \mathcal{U}_3 \vdash_{\mathrm{P}} \ldots$, will be computed as follows. Since $\mathcal{U}_0 = \mathcal{T}_0$, in the first iteration we compute

$$\begin{aligned}
\mathcal{U}_1 &= H_{\mathrm{P}}^{\nabla}(\mathcal{U}_0, \mathcal{T}_1)\\
&= \{\mathcal{P}_3\} \uplus_{\mathrm{P}} \{\mathcal{P}_0 \nabla_s \mathcal{P}_1\}\\
&= \{\mathcal{P}_1, \mathcal{P}_3\}\\
&= \mathcal{T}_1.
\end{aligned}$$

In the second iteration, we obtain

$$\begin{aligned}
\mathcal{U}_2 &= H_{\mathrm{P}}^{\nabla}(\mathcal{U}_1, \mathcal{T}_2)\\
&= \{\mathcal{P}_3\} \uplus_{\mathrm{P}} \{\mathcal{P}_1 \nabla_s \mathcal{P}_2\}\\
&= \{\mathcal{P}_2, \mathcal{P}_3\}\\
&= \mathcal{T}_2.
\end{aligned}$$

In the third iteration, letting

$$\begin{aligned}
\mathcal{P}'_0 &:= \mathcal{P}_3 \nabla_s \mathcal{P}_4\\
&= \{0 \leq x \leq 8, 0 \leq y \leq 8, x + y \leq 14, x - y \geq -6\},
\end{aligned}$$

[5] In this and the following examples, we abuse notation by writing a constraint system $\mathcal{C}$ to denote the polyhedron $\mathcal{P} = \mathrm{con}(\mathcal{C})$.

we obtain

$$\begin{aligned}\mathcal{U}_3 &= H^{\nabla}_{\mathrm{P}}(\mathcal{U}_2, \mathcal{T}_3)\\ &= \{\mathcal{P}_2\} \uplus_{\mathrm{P}} \{\mathcal{P}_3 \nabla_s \mathcal{P}_4\}\\ &= \{\mathcal{P}_3 \nabla_s \mathcal{P}_4\}\\ &= \{\mathcal{P}'_0\}.\end{aligned}$$

Note that the polyhedron $\mathcal{P}_2$ does not occur in $\mathcal{U}_3$ because it is made redundant by $\mathcal{P}'_0$ (i.e., $\mathcal{P}_2 \subseteq \mathcal{P}'_0$).

Now, the singleton $\mathcal{U}_3 = \{\mathcal{P}'_0\}$ has the same structure as the singleton $\mathcal{U}_0 = \{\mathcal{P}_0\}$, because the polyhedron $\mathcal{P}'_0$ can be obtained from $\mathcal{P}_0$ by a scaling (by a factor 2) followed by a rotation. As a consequence, it is possible to indefinitely extend the sequence $\mathcal{T}_i$ and the corresponding "widened" sequence $\mathcal{U}_i$ without obtaining convergence (in a finite number of steps). Since, in the above computation, all the abstract elements have cardinality less than or equal to 2, the addition of any (non-trivial) upper bound on the cardinality of the abstract descriptions will have no effect on termination.

5 Powerset Widenings using Set Cardinality

The first methodology we propose shares with [11] the idea of posing constraints on the cardinality of the arguments of the widening. As observed in the previous section, this approximation on its own is insufficient to ensure the operator is a widening. A closer inspection of the iterates in Example 3 shows that divergence is actually caused by the reduction function $\Omega^{\vdash}_{D}$, which interferes with the cardinality control mechanism by removing redundant elements, so that the cardinality threshold is never reached. The problem caused by $\Omega^{\vdash}_{D}$ is avoided by requiring the extrapolation heuristics h^{∇}_{P} being used by the widening to satisfy an additional property.

Definition 8. (∇-covered heuristics.) The extrapolation heuristics h^{∇}_{P} is said to be ∇-*covered* if, for all $S_1, S_2 \in \wp^{\vdash}_{\mathrm{fn}}(D)$ such that $S_1 \Vdash_{\mathrm{P}} S_2$, we have

$$\forall d_1 \in S_1 : \exists d \in h^{\nabla}_{\mathrm{P}}(S_1, S_2) \;.\; d_1 \vdash_{\nabla} d. \qquad (8)$$

Although H^{∇}_{P} of Definition 7 is not ∇-covered,[6] a ∇-covered extrapolation heuristics for $\hat{D}_{\mathrm{P}}$ can be obtained for any widening '∇' on the base-level domain $\hat{D}$ by selectively replacing the standard reduction map $\Omega^{\vdash}_{D}$ with a non-standard, widening-based reduction map Ω^{∇}_{D}. The idea being that if, in a set in $\wp_{\mathrm{f}}(D)$, one element d entails another d', then instead of just removing the redundant element d, the map Ω^{∇}_{D} replaces both d and d' by $d \nabla d'$.

Definition 9. (∇-reduction map, Ω^{∇}_{D}.) A function $\Omega^{\nabla}_{D} \colon \wp_{\mathrm{f}}(D) \rightarrow \wp^{\vdash}_{\mathrm{fn}}(D)$ is called ∇-*reduction map* if it satisfies the following property: for all $S \in \wp_{\mathrm{f}}(D)$, if $\Omega^{\nabla}_{D}(S) = S'$, then there exists a sequence $T_0, \ldots, T_m$ of elements of $\wp_{\mathrm{f}}(D)$ such that $T_0 = S$, $T_m = S'$ and, for each $0 < i \leq m$, $T_i = \bigl(T_{i-1} \setminus \{d, d'\}\bigr) \cup \{d \nabla d'\}$, where $d, d' \in T_{i-1}$ and $d \Vdash d'$. The (overloaded) operator $\Omega^{\nabla}_{D} \colon \wp^{\vdash}_{\mathrm{fn}}(D) \times \wp^{\vdash}_{\mathrm{fn}}(D) \rightarrow \wp^{\vdash}_{\mathrm{fn}}(D)$ is defined, for all $S_1, S_2 \in \wp^{\vdash}_{\mathrm{fn}}(D)$, by

$$\Omega^{\nabla}_{D}(S_1, S_2) := \Omega^{\nabla}_{D}(S_1 \cup S_2).$$

Proposition 2. *The Ω^{∇}_{D} operator is a ∇-covered extrapolation heuristics for $\hat{D}_{\mathrm{P}}$.*

Example 4. Consider the iteration sequence of Example 3 and suppose now that, instead of using H^{∇}_{P}, we adopt a ∇-covered extrapolation heuristics satisfying Definition 9. Then, we obtain the widened sequence $\mathcal{U}'_0 \vdash_{\mathrm{P}} \mathcal{U}'_1 \vdash_{\mathrm{P}} \mathcal{U}'_2 \vdash_{\mathrm{P}} \ldots$, where $\mathcal{U}'_0 = \mathcal{U}_0$, $\mathcal{U}'_1 = \mathcal{U}_1$ and $\mathcal{U}'_2 = \mathcal{U}_2$ would be computed as before. However in the third iteration we will obtain

$$\begin{aligned}\mathcal{U}'_3 &= \Omega^{\nabla}_{D}(\mathcal{U}_2, \mathcal{T}_3)\\ &= \Omega^{\nabla}_{D}\bigl(\{\mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4\}\bigr)\\ &= \Omega^{\nabla}_{D}\bigl(\{\mathcal{P}_2, \mathcal{P}_3 \nabla_s \mathcal{P}_4\}\bigr)\\ &= \Omega^{\nabla}_{D}\bigl(\{\mathcal{P}_2, \mathcal{P}'_0\}\bigr)\\ &= \{\mathcal{P}_2 \nabla_s \mathcal{P}'_0\}\\ &= \bigl\{\{x \geq 0, y \geq 0\}\bigr\},\end{aligned}$$

therefore breaking the divergence pattern.

In order to control the cardinality of the abstract iterates, several possible solutions could be adopted. In the following proposal, if the cardinality of the second argument of the widening S_2 exceeds a fixed bound k by some $\ell > 0$, we first collapse S_2 to a smaller set S'_2 by replacing a subset of cardinality $\ell + 1$ by its join (so that $\# S'_2 \leq k$ and $S_2 \vdash_{\mathrm{EM}} S'_2$).

Definition 10. (Collapsor.) For each $k \geq 1$, a unary operator $\Uparrow_k \colon \wp^{\vdash}_{\mathrm{fn}}(D) \rightarrow \wp^{\vdash}_{\mathrm{fn}}(D)$ is called a k-*collapsor* for $\hat{D}_{\mathrm{P}}$ if, for each $S \in \wp^{\vdash}_{\mathrm{fn}}(D)$, either $\# S \leq k$ and $\Uparrow_k S = S$ or there exists $S' \subseteq S$ with $\# S' > \# S - k$ such that

$$\Uparrow_k S = (S \setminus S') \oplus_{\mathrm{P}} \{\oplus S'\}.$$

We now define a widening on the powerset domain that uses a k-collapsor with a ∇-covered extrapolation heuristics to ensure convergence of the abstract iterates.

Definition 11. (The '${}_k\nabla_{\mathrm{P}}$' widening.) Let h^{∇}_{P} be a ∇-covered extrapolation heuristics and '$\Uparrow_k$' be a k-collapsor for $\hat{D}_{\mathrm{P}}$, for some $k \geq 1$. For all $S_1, S_2 \in \wp^{\vdash}_{\mathrm{fn}}(D)$ such that $S_1 \Vdash_{\mathrm{P}} S_2$, let

$$S_1 \;{}_k\nabla_{\mathrm{P}}\; S_2 := h^{\nabla}_{\mathrm{P}}(S_1, \Uparrow_k S_2).$$

Then '${}_k\nabla_{\mathrm{P}}$' is said to be a *cardinality-based widening.*

Theorem 1. *The '${}_k\nabla_{\mathrm{P}}$' operator is a widening on $\hat{D}_{\mathrm{P}}$.*

In general, the precision of this widening will depend on both the value of the parameter k and the particular heuristics used to select the subset to be collapsed.

[6] In particular, in Example 3 we saw that $\mathcal{P}_2 \in \mathcal{T}_2$ and $H^{\nabla}_{\mathrm{P}}(\mathcal{T}_2, \mathcal{T}_3) = \{\mathcal{P}'_0\}$, but $\mathcal{P}_2 \nvdash_{\nabla} \mathcal{P}'_0$.

Example 5. To illustrate the widening operator '${}_k\nabla_{\mathrm{P}}$' for $k = 2$, we consider the powerset domain $(\widehat{\mathbb{CP}}_1)_{\mathrm{P}}$ with the standard widening '∇_s' on $\widehat{\mathbb{CP}}_1$, a ∇-covered extrapolation heuristics Ω_D^{∇} satisfying Definition 9 and a 2-collapsor that, given a non-redundant and finite set of intervals on the x-axis, reduces its cardinality to 2 by taking the poly-hull of all the intervals but the one having the smallest lower bound. Consider the sequence $\mathcal{T}_0 \vdash_{\mathrm{P}} \mathcal{T}_1 \vdash_{\mathrm{P}} \cdots$ of Example 2 and the widened sequence $\mathcal{U}_0 \vdash_{\mathrm{P}} \mathcal{U}_1 \vdash_{\mathrm{P}} \cdots$ where $\mathcal{U}_0 = \mathcal{T}_0$ and $\mathcal{U}_i = \mathcal{U}_{i-1} \; {}_2\nabla_{\mathrm{P}} \; (\mathcal{U}_{i-1} \uplus_{\mathrm{P}} \mathcal{T}_i)$, for each $i > 0$. As $\mathcal{U}_0 \subset \mathcal{T}_1$, and $\# \mathcal{T}_1 = 2$, we obtain $\mathcal{U}_1 = \mathcal{T}_1$. Again $\mathcal{U}_1 \subset \mathcal{T}_2$. As $\# \mathcal{T}_2 = 3$, we compute $\Uparrow_2 \mathcal{T}_2$ before applying an Ω_D^{∇} operator. Thus, we obtain

$$\begin{aligned}
\mathcal{U}_1 &= \Omega_D^{\nabla}(\mathcal{U}_1, \Uparrow_2 \mathcal{T}_2)\\
&= \Omega_D^{\nabla}\Bigl(\mathcal{U}_1, \bigl\{\{x = 0\}, \{1 \leq x \leq 2\}\bigr\}\Bigr)\\
&= \Omega_D^{\nabla}\Bigl(\bigl\{\{x = 0\}, \{x = 1\}, \{1 \leq x \leq 2\}\bigr\}\Bigr)\\
&= \bigl\{\{x = 0\}, \{x = 1\} \nabla_s \{1 \leq x \leq 2\}\bigr\}\\
&= \bigl\{\{x = 0\}, \{1 \leq x\}\bigr\}.
\end{aligned}$$

In the next iteration we obtain stabilization.

6 Powerset Widenings Using Connectors

One obvious explanation for the divergence of the iteration sequence in Example 2 is that the base-level widening never comes into play. The second widening we propose is designed to avoid such situations.

To be more specific, when using the extrapolation heuristics H_{P}^{∇} of Definition 7, the reason for divergence is that elements of S_2 that do not cover an element in S_1 will be included, unchanged, in $H_{\mathrm{P}}^{\nabla}(S_1, S_2)$ without any involvement in a widening computation. To avoid this, one possibility is to replace S_2 with $S_1 \boxplus_{\mathrm{EM}} S_2$, where '$\boxplus_{\mathrm{EM}}$' is a connector for $\hat{D}_{\mathrm{P}}$, so that no such "brand new" elements can exist. Note that such a solution is not really specific for the extrapolation operator H_{P}^{∇}. Here we specify another subclass of extrapolation heuristics that, when combined with a connector operator as described above, can be used to define a proper widening operator.

Definition 12. (∇-connected heuristics.) The extrapolation heuristics h_{P}^{∇} is said to be ∇-*connected* if, for all $S_1, S_2 \in \wp_{\mathrm{fn}}^{\vdash}(D)$ where $S_1 \Vdash_{\mathrm{P}} S_2$, we have

$$\begin{aligned}
&\forall d \in h_{\mathrm{P}}^{\nabla}(S_1, S_2) \cap S_2 :\\
&\quad (\exists d_1 \in S_1 \,.\, d_1 \Vdash d) \implies (\exists d_1' \in S_1 \,.\, d_1' \Vdash_{\nabla} d). \qquad (9)
\end{aligned}$$

As already said, the above subclass includes the extrapolation heuristics H_{P}^{∇} of Definition 7.

Proposition 3. *The H_{P}^{∇} operator is a ∇-connected extrapolation heuristics for $\hat{D}_{\mathrm{P}}$.*

We therefore define a widening on the finite powerset domain that enforces convergence by using a connector with a ∇-connected extrapolation heuristics.

Definition 13. (The '${}_{\mathrm{EM}}\nabla_{\mathrm{P}}$' widening.) Let h_{P}^{∇} be a ∇-connected extrapolation heuristics and '$\boxplus_{\mathrm{EM}}$' be a connector for $\hat{D}_{\mathrm{P}}$. For all $S_1, S_2 \in \wp_{\mathrm{fn}}^{\vdash}(D)$ such that $S_1 \Vdash_{\mathrm{P}} S_2$, let $S_1 \; {}_{\mathrm{EM}}\nabla_{\mathrm{P}} \; S_2 := h_{\mathrm{P}}^{\nabla}(S_1, S_2')$ where

$$S_2' := \begin{cases} S_2, & \text{if } S_1 \vdash_{\mathrm{EM}} S_2;\\ S_1 \boxplus_{\mathrm{EM}} S_2, & \text{otherwise.} \end{cases}$$

Then '${}_{\mathrm{EM}}\nabla_{\mathrm{P}}$' is said to be a *connector-based widening.*

Theorem 2. *The '${}_{\mathrm{EM}}\nabla_{\mathrm{P}}$' operator is a widening on $\hat{D}_{\mathrm{P}}$.*

Clearly, the precision of this widening will depend on the chosen connector operator.

Example 6. To illustrate the widening operator '${}_{\mathrm{EM}}\nabla_{\mathrm{P}}$' we consider the powerset domain $(\widehat{\mathbb{CP}}_1)_{\mathrm{P}}$, with the standard widening '∇_s' on $\widehat{\mathbb{CP}}_1$ and the trivial connector '$\uplus_{\mathrm{EM}}$' returning the singleton poly-hull of its arguments. Consider the sequence $\mathcal{T}_0 \vdash_{\mathrm{P}} \mathcal{T}_1 \vdash_{\mathrm{P}} \cdots$ of Example 2 and the widened sequence $\mathcal{U}_0 \vdash_{\mathrm{P}} \mathcal{U}_1 \vdash_{\mathrm{P}} \cdots$ where $\mathcal{U}_0 = \mathcal{T}_0$ and $\mathcal{U}_i = \mathcal{U}_{i-1} \; {}_{\mathrm{EM}}\nabla_{\mathrm{P}} \; (\mathcal{U}_{i-1} \uplus_{\mathrm{P}} \mathcal{T}_i)$, for each $i > 0$. When computing $\mathcal{U}_1$, the second argument of the widening is $\mathcal{U}_0 \uplus_{\mathrm{P}} \mathcal{T}_1 = \mathcal{T}_1$. Note that $\mathcal{U}_0 \vdash_{\mathrm{EM}} \mathcal{T}_1$ does not hold so that the connector is needed. Thus, we obtain

$$\begin{aligned}
\mathcal{U}_1 &= H_{\mathrm{P}}^{\nabla}(\mathcal{U}_0, \mathcal{U}_0 \uplus_{\mathrm{EM}} \mathcal{T}_1)\\
&= H_{\mathrm{P}}^{\nabla}\Bigl(\mathcal{U}_0, \bigl\{\{0 \leq x \leq 1\}\bigr\}\Bigr)\\
&= \bigl\{\{0 \leq x\}\bigr\}.
\end{aligned}$$

In the next iteration we obtain stabilization.

Consider now the powerset domain $(\widehat{\mathbb{CP}}_2)_{\mathrm{P}}$ and the iteration sequences $\mathcal{T}_0 \vdash_{\mathrm{P}} \mathcal{T}_1 \vdash_{\mathrm{P}} \cdots$ and $\mathcal{U}_0 \vdash_{\mathrm{P}} \mathcal{U}_1 \vdash_{\mathrm{P}} \cdots$ of Example 3. Let $\mathcal{U}_0' \vdash_{\mathrm{P}} \mathcal{U}_1' \vdash_{\mathrm{P}} \cdots$ be the sequence computed by using the widening operator '${}_{\mathrm{EM}}\nabla_{\mathrm{P}}$' specified above. Clearly, $\mathcal{U}_0' = \mathcal{U}_0 = \mathcal{T}_0$. However, by letting

$$\begin{aligned}
\mathcal{P}_5 &:= \mathcal{P}_0 \uplus \mathcal{P}_1 \uplus \mathcal{P}_3\\
&= \{0 \leq x \leq 8, 0 \leq y \leq 8,\\
&\qquad x + y \leq 14, x - y \geq -6, 3x - y \geq -4\},
\end{aligned}$$

instead of obtaining $\mathcal{U}_1 = \{\mathcal{T}_1\}$ we obtain

$$\begin{aligned}
\mathcal{U}_1' &= H_{\mathrm{P}}^{\nabla}(\mathcal{U}_0', \mathcal{U}_0' \uplus_{\mathrm{EM}} \mathcal{T}_1)\\
&= H_{\mathrm{P}}^{\nabla}\bigl(\mathcal{U}_0', \{\mathcal{P}_0 \uplus \mathcal{P}_1 \uplus \mathcal{P}_3\}\bigr)\\
&= H_{\mathrm{P}}^{\nabla}\bigl(\mathcal{U}_0', \{\mathcal{P}_5\}\bigr)\\
&= \bigl\{\mathcal{P}_0 \nabla_s \mathcal{P}_5\bigr\}\\
&= \bigl\{\{x \geq 0, y \geq 0\}\bigr\}.
\end{aligned}$$

7 Powerset Widenings Using Certificates

The third widening technique we present in this paper requires a certificate $(\mathcal{O}, \preceq, \mu)$ for the base-level widening '∇'. Widenings obtained in this way will be termed *certificate-based*. As it is used in the computation of the new widening on the powerset domain, the certificate for the base-level widening must be *finitely computable*, i.e., such that the partial order '$\preceq$' and the level mapping μ are both finitely computable. This means that such a certificate for '∇' on $\hat{D}$ cannot be simply taken as $(D, \vdash_{\nabla}, \mathcal{I})$, where $\mathcal{I}$ is the identity map, since '$\vdash_{\nabla}$', in general, does not come with a computability guarantee.[7]

The certificate-based widening can be seen as a variant of the framework presented in [5,6], which provided a clean and formal development for an idea originally sketched in [8]. We will use the certificate μ to define a suitable relation '$\curvearrowright_{\mathrm{P}}$' on the finite powerset domain $\hat{D}_{\mathrm{P}}$. This relation will be shown to be a *limited growth ordering* (lgo) [5,6] on $\hat{D}_{\mathrm{P}}$, i.e., the strict version of a finitely computable preorder on $\wp_{\mathrm{fn}}^{\vdash}(D)$ that satisfies the ascending chain condition. Intuitively, the new widening '${}_{\mu}\nabla_{\mathrm{P}}$' on the finite powerset domain will be defined so that the relation '$\curvearrowright_{\mathrm{P}}$' will correspond to the relation '$\Vdash_{{}_{\mu}\nabla_{\mathrm{P}}}$'.

Definition 14. ('$\curvearrowright_{\mathrm{P}}$'.) Let $(\mathcal{O}, \preceq, \mu)$ be a finitely computable certificate for the widening operator '∇' on $\hat{D}$. The relation $\curvearrowright_{\mathrm{P}} \subseteq \wp_{\mathrm{fn}}^{\vdash}(D) \times \wp_{\mathrm{fn}}^{\vdash}(D)$ induced by μ is such that, for each $S_1, S_2 \in \wp_{\mathrm{fn}}^{\vdash}(D)$, $S_1 \curvearrowright_{\mathrm{P}} S_2$ holds if and only if, letting $d_1 := \bigoplus S_1$ and $d_2 := \bigoplus S_2$, one of the following holds:

$$\mu(d_1) \prec \mu(d_2); \tag{10}$$

$$\mu(d_1) = \mu(d_2) \wedge \# S_1 > 1 \wedge \# S_2 = 1; \tag{11}$$

$$\mu(d_1) = \mu(d_2) \wedge \# S_1 > 1 \wedge \# S_2 > 1 \wedge \tilde{\mu}(S_1) \mathrel{\prec\!\!\!\prec} \tilde{\mu}(S_2), \tag{12}$$

where the function $\tilde{\mu} \colon \wp_{\mathrm{fn}}^{\vdash}(D) \to \mathcal{M}(\mathcal{O})$ is defined, for each $S \in \wp_{\mathrm{fn}}^{\vdash}(D)$, so that $\tilde{\mu}(S)$ is the multiset over $\mathcal{O}$ obtained by applying μ to each element in S. Namely, by $\tilde{\mu}(S) := \bigl\{\, \mu(d) \bigm| d \in S \,\bigr\}$.

Proposition 4. *The '$\curvearrowright_{\mathrm{P}}$' relation is finitely computable and satisfies the ascending chain condition.*

Example 7. Consider the polyhedra in Figure 3 and let

$$\mathcal{T}_1 := \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4\},$$
$$\mathcal{T}_2 := \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3\},$$

so that $\mathcal{P} = \biguplus \mathcal{T}_1$ and $\mathcal{Q} = \biguplus \mathcal{T}_2$. By Definition 3, $\mu_s(\mathcal{P}) = (0,5) \prec_s (0,4) = \mu_s(\mathcal{Q})$ so that, by condition (10) of Definition 14, we obtain $\mathcal{T}_1 \curvearrowright_{\mathrm{P}} \mathcal{T}_2$.

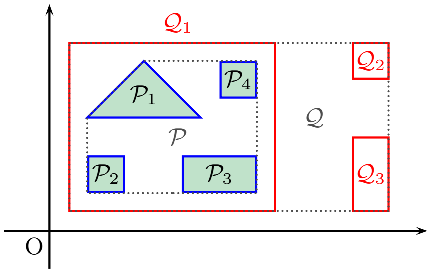

Fig. 3. The lgo relation on collections of polyhedra.

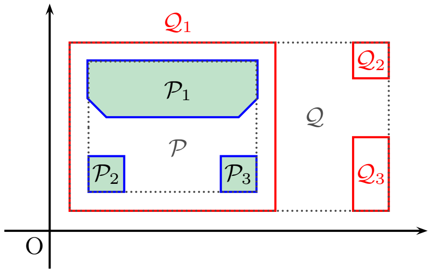

Fig. 4. The lgo relation on collections of polyhedra.

Now consider the polyhedra in Figure 4 and let

$$\mathcal{U}_1 := \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3\},$$
$$\mathcal{U}_2 := \{\mathcal{Q}_1\},$$
$$\mathcal{U}_3 := \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3\},$$

so that $\mathcal{U}_1 \Vdash_{\mathrm{P}} \mathcal{U}_2 \Vdash_{\mathrm{P}} \mathcal{U}_3$, $\mathcal{P} = \biguplus \mathcal{U}_1$, $\mathcal{Q}_1 = \biguplus \mathcal{U}_2$, $\mathcal{Q} = \biguplus \mathcal{U}_3$ and $\mu_s(\mathcal{P}) = \mu_s(\mathcal{Q}_1) = \mu_s(\mathcal{Q}) = (0,4)$. Since $\# \mathcal{U}_2 = 1$, none of the conditions of Definition 14 apply so that $\mathcal{U}_2 \not\curvearrowright_{\mathrm{P}} \mathcal{U}_3$. On the other hand, as $\# \mathcal{U}_1 = \# \mathcal{U}_3 = 3$, we have $\mathcal{U}_1 \curvearrowright_{\mathrm{P}} \mathcal{U}_2$ by condition (11) of Definition 14. Finally, since

$$\tilde{\mu}_s(\mathcal{U}_1) = \bigl\{(0,6), (0,4), (0,4)\bigr\}$$
$$\mathrel{\prec\!\!\!\prec_s} \bigl\{(0,4), (0,4), (0,4)\bigr\} = \tilde{\mu}_s(\mathcal{U}_3),$$

we obtain $\mathcal{U}_1 \curvearrowright_{\mathrm{P}} \mathcal{U}_3$ using condition (12) of Definition 14.

For the sake of precision, the specification of our certificate-based widening assumes the existence of a *subtract* operation for the base-level domain. It is expected that a specific subtraction would be provided for each domain; here we just indicate a minimal specification that can be trivially satisfied in case no better alternative is available.[8]

Definition 15. (Subtraction.) A *subtraction* for $\hat{D}$ is any partial operator $\ominus \colon D \times D \rightarrowtail D$ such that, for all $d_1, d_2 \in D$, $d_2 \vdash d_1$ implies that $d_1 \ominus d_2$ is defined and both $d_1 \ominus d_2 \vdash d_1$ and $d_1 = (d_1 \ominus d_2) \oplus d_2$ hold.

[7] Note that, for the domain $\mathbb{CP}_n$ of polyhedra and the standard widening '∇_s', the certificate $(\mathcal{O}_s, \preceq_s, \mu_s)$ provided in Definition 3 is finitely computable.

[8] The *subtraction operators* defined here are very similar to the *weakening operators* introduced in [2, Definition 40, page 146].

A trivial subtraction operator can always be defined as $d_1 \ominus d_2 := d_1$. In practice, when designing a widening, the actual subtraction operator would be expected to lose as little precision as possible.

Example 8. The operator $\mathrm{pdiff} \colon \mathbb{CP}_n \times \mathbb{CP}_n \to \mathbb{CP}_n$ is defined so that, for any $\mathcal{P}_1, \mathcal{P}_2 \in \mathbb{CP}_n$, $\mathrm{pdiff}(\mathcal{P}_1, \mathcal{P}_2)$ denotes the smallest closed and convex polyhedron containing the set difference $\mathcal{P}_1 \setminus \mathcal{P}_2$. Then, if $\mathcal{P}_2 \subseteq \mathcal{P}_1$, we have $\mathrm{pdiff}(\mathcal{P}_1, \mathcal{P}_2) \subseteq \mathcal{P}_1$ and

$$\begin{aligned} \mathcal{P}_1 &= (\mathcal{P}_1 \setminus \mathcal{P}_2) \cup \mathcal{P}_2 \\ &= \mathrm{pdiff}(\mathcal{P}_1, \mathcal{P}_2) \cup \mathcal{P}_2 \\ &= \mathrm{pdiff}(\mathcal{P}_1, \mathcal{P}_2) \uplus \mathcal{P}_2, \end{aligned}$$

so that 'pdiff' is a subtraction for $\widehat{\mathbb{CP}}_n$.

We now define a widening on the finite powerset domain that enforces convergence by means of a limited growth ordering induced by a certificate for the base-level widening.

Definition 16. (The '${}_\mu\nabla_{\mathrm{P}}$' widening.) Let '$\curvearrowright_{\mathrm{P}}$' be the limited growth ordering induced by the certificate μ for '∇', let '$\boxplus_{\mathrm{P}}$' be any upper bound operator on $\hat{D}_{\mathrm{P}}$ and '$\ominus$' be a subtraction for $\hat{D}$. For all $S_1, S_2 \in \wp_{\mathrm{fn}}^{\vdash}(D)$ such that $S_1 \Vdash_{\mathrm{P}} S_2$, let

$$S_1 \mathbin{{}_\mu\nabla_{\mathrm{P}}} S_2 := \begin{cases} S, & \text{if } S_1 \curvearrowright_{\mathrm{P}} S; \\ S \oplus_{\mathrm{P}} \{d\}, & \text{if } \bigoplus S_1 \Vdash \bigoplus S; \\ \{\bigoplus S\}, & \text{otherwise} \end{cases}$$

where $S := S_1 \boxplus_{\mathrm{P}} S_2$ and $d := \bigl(\bigoplus S_1 \mathbin{\nabla} \bigoplus S\bigr) \ominus \bigl(\bigoplus S\bigr)$. Then '${}_\mu\nabla_{\mathrm{P}}$' is said to be a *certificate-based widening.*

Theorem 3. *The '${}_\mu\nabla_{\mathrm{P}}$' operator is a widening on $\hat{D}_{\mathrm{P}}$.*

There are three cases in the specification of '${}_\mu\nabla_{\mathrm{P}}$' in Definition 16. In the first one, the widening simply returns the upper bound $S = S_1 \boxplus_{\mathrm{P}} S_2$, since this is enough to ensure a strict increase in the limited growth ordering relation. In the second case, the join of S_1 is strictly more precise than the join of S, so that we apply the base-level widening '∇' to them and then, using the subtraction operator, improve the obtained result, since $S_1 \curvearrowright_{\mathrm{P}} S \oplus_{\mathrm{P}} \{d\}$ holds. In the last case, since the join of S_1 is equivalent to the join of S, we return the singleton consisting of the join itself, as originally proposed in [16, Section 9].

Example 9. To illustrate the three cases of Definition 16, consider the finite powerset domain $(\widehat{\mathbb{CP}}_2)_{\mathrm{P}}$, with the standard widening '∇_s' for $\widehat{\mathbb{CP}}_2$, certified by the level mapping μ_s defined in Definition 3. Let also the subtraction '$\ominus$' be defined as 'pdiff' and the upper bound '$\boxplus_{\mathrm{P}}$' be defined as '$\oplus_{\mathrm{P}}$', so that, for all $\mathcal{S}_1, \mathcal{S}_2 \in (\widehat{\mathbb{CP}}_2)_{\mathrm{P}}$ such that $\mathcal{S}_1 \vdash_{\mathrm{P}} \mathcal{S}_2$, we will have $\mathcal{S}_1 \boxplus_{\mathrm{P}} \mathcal{S}_2 = \mathcal{S}_2$.

In particular, this means that in the first case of Definition 16, $\mathcal{S}_1 \mathbin{{}_\mu\nabla_{\mathrm{P}}} \mathcal{S}_2 = \mathcal{S}_2$. Applications of this case can be observed by considering the sets of polyhedra introduced in Example 7 (see Figure 3)

$$\{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4\} \mathbin{{}_\mu\nabla_{\mathrm{P}}} \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3\} = \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3\}$$

and (see Figure 4)

$$\begin{aligned} \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3\} \mathbin{{}_\mu\nabla_{\mathrm{P}}} \{\mathcal{Q}_1\} &= \{\mathcal{Q}_1\}, \\ \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3\} \mathbin{{}_\mu\nabla_{\mathrm{P}}} \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3\} &= \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3\}. \end{aligned}$$

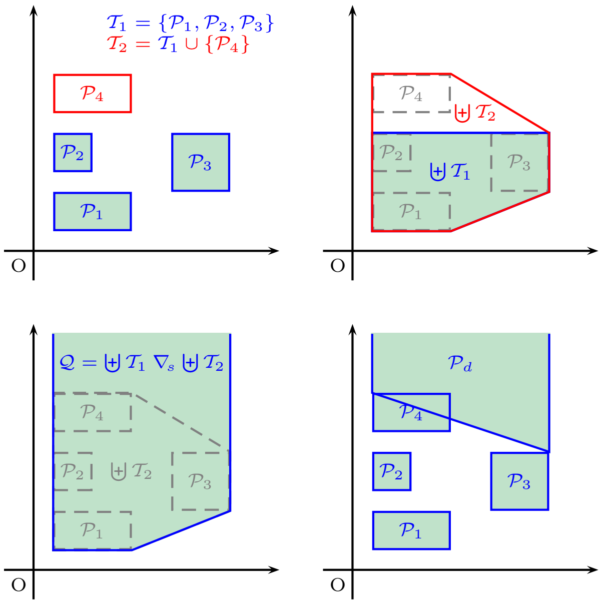
$\mathcal{T}_1 = \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3\}$
$\mathcal{T}_2 = \mathcal{T}_1 \cup \{\mathcal{P}_4\}$
$\mathcal{P}_4$
$\mathcal{P}_2$
$\mathcal{P}_3$
$\mathcal{P}_1$
O
$\mathcal{P}_4$
$\biguplus \mathcal{T}_2$
$\mathcal{P}_2$
$\biguplus \mathcal{T}_1$
$\mathcal{P}_3$
$\mathcal{P}_1$
O
$\mathcal{Q} = \biguplus \mathcal{T}_1 \nabla_s \biguplus \mathcal{T}_2$
$\mathcal{P}_4$
$\mathcal{P}_2$
$\biguplus \mathcal{T}_2$
$\mathcal{P}_3$
$\mathcal{P}_1$
O
$\mathcal{P}_d$
$\mathcal{P}_4$
$\mathcal{P}_2$
$\mathcal{P}_3$
$\mathcal{P}_1$
O

Fig. 5. The second case of the certificate-based widening.

Figure 5 illustrates an application of the second case of Definition 16. Consider the computation of $\mathcal{T}_1 \mathbin{{}_\mu\nabla_{\mathrm{P}}} \mathcal{T}_2$ where sets $\mathcal{T}_1$ and $\mathcal{T}_2$ are as shown in the top left diagram while their poly-hulls $\biguplus \mathcal{T}_1$ and $\biguplus \mathcal{T}_2$ are shown in the top right diagram. Since $\mu_s\bigl(\biguplus \mathcal{T}_1\bigr) = (0,5) \not\succ_s (0,6) = \mu_s\bigl(\biguplus \mathcal{T}_2\bigr)$, we obtain $\mathcal{T}_1 \not\curvearrowright_{\mathrm{P}} \mathcal{T}_2$, so that the first case of Definition 16 does not apply. As $\biguplus \mathcal{T}_1 \subset \biguplus \mathcal{T}_2$, the second case applies and we use the base-level widening to compute $\mathcal{Q} = \biguplus \mathcal{T}_1 \nabla_s \biguplus \mathcal{T}_2$, shown in the bottom left diagram. We then find the convex polyhedral difference $\mathcal{P}_d = \mathrm{pdiff}\bigl(\mathcal{Q}, \biguplus \mathcal{T}_2\bigr)$, to obtain the widened set $\mathcal{T}_1 \mathbin{{}_\mu\nabla_{\mathrm{P}}} \mathcal{T}_2 = \mathcal{T}_2 \cup \{\mathcal{P}_d\}$ as shown in the bottom right diagram. Note that, as far as the divergence problem is concerned, it would be enough to return the singleton set $\{\mathcal{Q}\}$, since $\mathcal{T}_1 \curvearrowright_{\mathrm{P}} \{\mathcal{Q}\}$ already holds; however, by exploiting the availability of a non-trivial subtraction operator, we can usually obtain, as in this case, a more precise result.

The polyhedra shown in the top left diagram of Figure 5 can also illustrate the application of the third and last case of Definition 16. To this end, consider the set $\mathcal{T}_1' = \{\mathcal{P}_1, \mathcal{P}_3, \mathcal{P}_4\}$ and the computation of $\mathcal{T}_1' \mathbin{{}_\mu\nabla_{\mathrm{P}}} \mathcal{T}_2$. Since $\biguplus \mathcal{T}_1' = \biguplus \mathcal{T}_2$, the first two cases of Definition 16 do not apply, so that we compute $\mathcal{T}_1' \mathbin{{}_\mu\nabla_{\mathrm{P}}} \mathcal{T}_2 = \bigl\{\biguplus \mathcal{T}_2\bigr\}$.

As shown in the example above, Definition 16 does not require that the upper bound operator '$\boxplus_{\mathrm{P}}$' is based on the base-level widening '∇'. Moreover, the scheme of Definition 16 can be easily extended to any finite set of heuristically-chosen upper bound operators on $\hat{D}_{\mathrm{P}}$, still obtaining a proper widening operator for the powerset domain. The simplest heuristics, already used in the example above, is the one taking $\boxplus_{\mathrm{P}} := \oplus_{\mathrm{P}}$. If this fails to ensure an increase in the level mapping, another possibility is the adoption of an extrapolation heuristics h^{∇}_{P} for $\hat{D}_{\mathrm{P}}$. Anyway, many variations could be defined, depending on the required precision/efficiency trade-off. In the following section, we investigate one of these possibilities, which originates as a generalization of an idea proposed in [11].

8 Merging Elements According to a Congruence Relation

For any powerset widening, it may be possible to *merge together* (i.e., join) some of the elements occurring in the second argument without compromising the finite convergence guarantee. This merging operation can be guided by a congruence relation on the finite powerset domain, the idea being that a well-chosen relation will benefit the precision/efficiency trade-off of the widening.

One option is to use semantics preserving congruence relations, i.e., refinements of the congruence relation '$\equiv_{\gamma_{\mathrm{P}}}$'. As the purpose of this paper is to provide generic widening procedures for powersets, here we only consider congruences that may be defined independently of any particular concrete domain and the intended widening. Two such relations are the *identity congruence* relation, where trivial equivalence is assumed, and the *$\oplus$-congruence* relation, where sets that have the same join are equivalent. However, the identity congruence allows for no merging at all and hence has no effect at all on the iteration sequence. On the other hand, the $\oplus$-congruence enables a direct application of the base-level widening on the joined elements, thereby providing a default heuristics that is likely to hasten convergence. We now define a new non-identity congruence relation for any powerset domain that refines the $\oplus$-congruence.

Definition 17. ('$\lhd$' and '$\bowtie$'.) The *content* relation $\lhd \subseteq \wp^{\vdash}_{\mathrm{fn}}(D) \times \wp^{\vdash}_{\mathrm{fn}}(D)$ is such that $S_1 \lhd S_2$ holds if and only if, for all $S_1' \in \wp^{\vdash}_{\mathrm{fn}}(D)$,

$$S_1' \vdash_{\mathrm{P}} S_1 \implies \exists S_1'' \in \wp^{\vdash}_{\mathrm{fn}}(D) \mathrel{.} \bigoplus S_1' = \bigoplus S_1'' \land S_1'' \vdash_{\mathrm{P}} S_2.$$

The *same-content* relation $\bowtie \subseteq \wp^{\vdash}_{\mathrm{fn}}(D) \times \wp^{\vdash}_{\mathrm{fn}}(D)$ is such that $S_1 \bowtie S_2$ holds if and only if $S_1 \lhd S_2$ and $S_2 \lhd S_1$.

Thus $S_1 \vdash_{\mathrm{P}} S_2$ implies that $S_1 \lhd S_2$ so that $S \lhd \bigl\{\bigoplus S\bigr\}$, since $S \vdash_{\mathrm{P}} \bigl\{\bigoplus S\bigr\}$. Moreover, if S_2 is a singleton, then $S_1 \vdash_{\mathrm{P}} S_2$ if and only if $S_1 \lhd S_2$. Hence '$\bowtie$' is a congruence relation on $\hat{D}_{\mathrm{P}}$ that refines the $\oplus$-congruence.

Observe that the identity congruence relation can be obtained by strengthening the conditions in the definition of '$\lhd$', replacing $\bigoplus S_1' = \bigoplus S_1''$ with $S_1' = S_1''$; and the $\oplus$-congruence can be obtained by weakening the conditions, replacing $S_1'' \vdash_{\mathrm{P}} S_2$ with $\bigoplus S_1'' \vdash \bigoplus S_2$. Thus the same-content relation is a compromise between keeping all the information provided by the explicit set structure, as done by the identity congruence, and losing all of this information, as occurs with the $\oplus$-congruence.

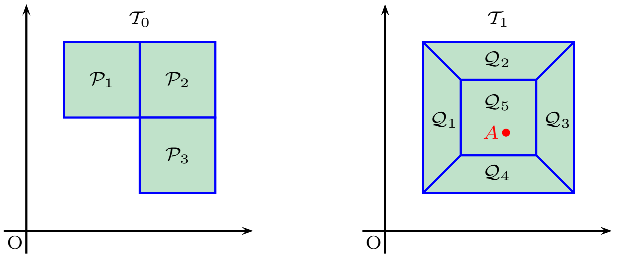

Fig. 6. Merging polyhedra according to '$\bowtie$'.

For the finite powerset domain of polyhedra $(\widehat{\mathbb{CP}_n})_{\mathrm{P}}$, the content relation '$\lhd$' corresponds to the condition that $\mathcal{S}_1 \lhd \mathcal{S}_2$ holds if and only if $\bigcup \mathcal{S}_1 \subseteq \bigcup \mathcal{S}_2$; and hence, the same-content relation '$\bowtie$' coincides with the induced congruence relation '$\equiv_{\gamma^{\mathbb{A}}_{\mathrm{P}}}$'.

Proposition 5. *For all $\mathcal{S}_1, \mathcal{S}_2 \in \wp^{\subseteq}_{\mathrm{fn}}(\mathbb{CP}_n)$, $\mathcal{S}_1 \bowtie \mathcal{S}_2$ if and only if $\mathcal{S}_1 \equiv_{\gamma^{\mathbb{A}}_{\mathrm{P}}} \mathcal{S}_2$.*

Example 10. Consider the polyhedra in Figure 6. Then $\mathcal{P}_1 \uplus \mathcal{P}_2 = \mathcal{P}_1 \cup \mathcal{P}_2$ and $\mathcal{P}_2 \uplus \mathcal{P}_3 = \mathcal{P}_2 \cup \mathcal{P}_3$ so that, as $\mathcal{T}_0 = \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3\}$,

$$\mathcal{T}_0 \bowtie \{\mathcal{P}_1 \cup \mathcal{P}_2, \mathcal{P}_3\} \bowtie \{\mathcal{P}_1, \mathcal{P}_2 \cup \mathcal{P}_3\}.$$

On the other hand, since $\mathcal{T}_1 = \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3, \mathcal{Q}_4, \mathcal{Q}_5\}$, letting $\mathcal{T}_2 := \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3, \mathcal{Q}_4\}$, although $\biguplus \mathcal{T}_1 = \biguplus \mathcal{T}_2$, $\mathcal{T}_1 \ntriangleleft \mathcal{T}_2$. To see this, let A be an inner point of polyhedron $\mathcal{Q}_5$ and let S_1, S_2, and S_1' in Definition 17 be $\mathcal{T}_1$, $\mathcal{T}_2$, and $\mathcal{T}_1' = \bigl\{\{A\}\bigr\}$, respectively. Then, the one and only collection of polyhedra having the same poly-hull of $\mathcal{T}_1'$ is $\mathcal{T}_1'$ itself. However, $\mathcal{T}_1' \nvdash_{\mathrm{P}} \mathcal{T}_2$, so that $\mathcal{T}_1 \ntriangleleft \mathcal{T}_2$.

We now define an operation *merger* for the powerset domain that replaces selected subsets by congruent singleton sets for any given congruence relation.

Definition 18. (Mergers.) Let '$\sim$' be a congruence relation on $\hat{D}_{\mathrm{P}}$. The relation $\mathrm{merge}_{\sim} \subseteq \wp^{\vdash}_{\mathrm{fn}}(D) \times \wp^{\vdash}_{\mathrm{fn}}(D)$ is such that $\mathrm{merge}_{\sim}(S_1, S_2)$ holds if and only if $S_1 \vdash_{\mathrm{P}} S_2$ and

$$\forall d_2 \in S_2 : \exists S_1' \subseteq S_1 \mathrel{.} \{d_2\} \sim S_1'.$$

A set $S \in \wp^{\vdash}_{\mathrm{fn}}(D)$ is *fully-merged for '$\sim$'*, if $\mathrm{merge}_{\sim}(S, S')$ implies $S = S'$; S is *pairwise-merged for '$\sim$'* if, for all $d_1, d_2 \in S$, we have that $\{d_1, d_2\}$ is fully-merged. A unary operator $\upharpoonright_{\sim} : \wp^{\vdash}_{\mathrm{fn}}(D) \to \wp^{\vdash}_{\mathrm{fn}}(D)$ is a *merger* for '$\sim$' if $\mathrm{merge}_{\sim}(S, \upharpoonright_{\sim} S)$ holds for all $S \in \wp^{\vdash}_{\mathrm{fn}}(D)$.

Observe that, for all $S \in \wp_{\mathrm{fn}}^{\vdash}(D)$, we have $S \vdash_{\mathrm{EM}} \upharpoonleft_{\sim} S$.

As '$\sim$' is a congruence relation on $\hat{D}_{\mathrm{P}}$, for any merger '$\upharpoonleft_{\sim}$' for '$\sim$' and $S \in \hat{D}_{\mathrm{P}}$, $S \sim (\upharpoonleft_{\sim} S)$ holds. Assuming that we use a congruence relation that refines the $\oplus$-congruence, we can always merge a set to obtain one that is fully- or pairwise-merged.

Proposition 6. *Let '$\sim$' be a congruence relation on $\hat{D}_{\mathrm{P}}$ that refines the $\oplus$-congruence relation. Then there exists a merger '$\upharpoonleft_{\sim}$' such that, for all $S \in \wp_{\mathrm{fn}}^{\vdash}(D)$, $\upharpoonleft_{\sim} S$ is fully-merged (resp., pairwise-merged).*

For the finite powerset domain over $\widehat{\mathbb{CP}}_n$, lines 1–9 of the algorithm specified in [11, Figure 8, page 773] define a merger operator '$\upharpoonleft_{\bowtie}$' such that, for each finite set $\mathcal{S}$ of polyhedra, $\upharpoonleft_{\bowtie} \mathcal{S}$ is pairwise-merged. Thus, the algorithm described in [11] is in fact the composition of three computational devices: the pairwise-merger for '$\bowtie$', the extrapolation heuristics H_{P}^{∇} of Definition 7 and a cardinality control mechanism (which is not formally specified). Since all the abstract elements considered in Example 3 are fully-merged, even the complete algorithm is still implementing an extrapolation operator which is not a proper widening.

Example 11. To illustrate the merge relation for the finite powerset domain of polyhedra $(\widehat{\mathbb{CP}}_n)_{\mathrm{P}}$ with the '$\bowtie$' congruence relation, we again use the two diagrams in Figure 6. The set of polyhedra $\mathcal{T}_0$ in the left-hand diagram is not pairwise-merged since

$$\begin{aligned} \{\mathcal{P}_1 \uplus \mathcal{P}_2\} &= \{\mathcal{P}_1 \cup \mathcal{P}_2\} \bowtie \{\mathcal{P}_1, \mathcal{P}_2\}, \\ \{\mathcal{P}_2 \uplus \mathcal{P}_3\} &= \{\mathcal{P}_2 \cup \mathcal{P}_3\} \bowtie \{\mathcal{P}_2, \mathcal{P}_3\}. \end{aligned}$$

The sets $\{\mathcal{P}_1 \cup \mathcal{P}_2, \mathcal{P}_3\}$ and $\{\mathcal{P}_1, \mathcal{P}_2 \cup \mathcal{P}_3\}$ are fully-merged and hence pairwise-merged, and both $\mathrm{merge}_{\bowtie}\bigl(\mathcal{T}_0, \{\mathcal{P}_1 \cup \mathcal{P}_2, \mathcal{P}_3\}\bigr)$ and $\mathrm{merge}_{\bowtie}\bigl(\mathcal{T}_0, \{\mathcal{P}_1, \mathcal{P}_2 \cup \mathcal{P}_3\}\bigr)$ hold. The set $\mathcal{T}_1$ in the right-hand diagram is pairwise-merged but not fully-merged. Since $\biguplus \mathcal{T}_1 = \bigcup \mathcal{T}_1$, the singleton set $\{\bigcup \mathcal{T}_1\}$ is fully-merged and hence pairwise-merged and $\mathrm{merge}_{\bowtie}\bigl(\mathcal{T}_1, \{\bigcup \mathcal{T}_1\}\bigr)$ holds.

9 Conclusion

We have studied the problem of endowing any abstract domain obtained by means of the finite powerset construction with a provably correct widening operator. We have proposed three generic widening operators that are parametric with respect to the specification of some simpler operators: these are summarized in Table 1, where we distinguish the operators that are actually required (even though defaults are available for some of them) from those that can be optionally used to tune the precision/efficiency trade-off of the widenings. These generic constructions are, on the one hand, strong enough to enforce the convergence of the abstract iteration sequence while, on the other hand, they provide the flexibility needed to attack the precision problem in a domain-dependent, problem-dependent way.

Widening	*Required*	*Optional*
${}_{k}\nabla_{\mathrm{P}}$	∇-covered h_{P}^{∇} (Ω_{D}^{∇}) k-collapsor $\Uparrow_{k}$	merger $\upharpoonleft_{\sim}$
${}_{\mathrm{EM}}\nabla_{\mathrm{P}}$	∇-connected h_{P}^{∇} $(H_{\mathrm{P}}^{\nabla})$ connector $\boxplus_{\mathrm{EM}}$	merger $\upharpoonleft_{\sim}$
${}_{\mu}\nabla_{\mathrm{P}}$	certificate μ upper bound $\boxplus_{\mathrm{P}}$ subtraction $\ominus$ (π_1)	merger $\upharpoonleft_{\sim}$ other upper bounds

Table 1. The required and optional operators for the three widenings (defaults defined in the present paper in parentheses).

Given three alternative methodologies for the specification of a widening, it is natural to question whether or not one is better than the others with respect to some "measure," be it the precision of the obtained results, the expected efficiency of the overall analysis, or even the ease of implementation. Not surprisingly, the very generality of the finite powerset construction itself makes it impossible to answer such a question in general. It should also be stressed that, when comparing different widening operators, an "immediate" precision improvement does not automatically imply a corresponding "overall" precision improvement, because accurate widening operators are usually not monotonic in their arguments. Nonetheless, the following observations provide useful guidelines for the prospective widening designer.

A rather obvious remark is that all the widenings presented in this paper directly depend on the base-level widening used for the extrapolation heuristics or the convergence certificate: there is little hope that any of the methodologies we propose will behave well if the base-level widening is inadequate for the considered application. This observation applies not only to the precision of the widening, but also (to some extent) to its efficiency.

If the finite powerset construction is applied to a reasonably precise base-level domain, such as the domain of convex polyhedra, then a certificate-based widening may be a good starting point. In fact, this methodology provides the designer with a high degree of flexibility: not only does it provide a choice between several, possibly fine-grained certificates, but it also allows the designer to experiment with a potentially wide range of upper bound operators. In such a context, the cardinalities of the collections of abstract elements are relatively small, since most of the precision of the approximation is already encoded in the base-level elements (e.g., the polyhedra). In other words, the finite powerset construction is meant to avoid the precision losses caused by a few, mainly irregular, growth patterns.

If, in contrast, the base-level domain and its widening are somehow weak, as might be the case for the abstract domain of intervals [14], then a certificate-based widening is probably not a good choice. In this case, the simplicity and efficiency of the base-level domain are better exploited by increasing the cardinality of the abstract collections of elements, so that a cardinality-based widening is likely to provide better results.

The connector-based widening can be seen as an intermediate alternative: on the one hand, it shares with the cardinality-based widening the simplicity (and ease of implementation) of the approach; on the other hand, in order to obtain reasonable precision, it requires that an adequate connector operator is identified for the application at hand.

As already stressed in the introduction, in all cases the resulting widening operator will have to undergo a thorough experimentation in order to ascertain to what extent it actually adapts to the specific class of problem instances that are of interest. For the finite powerset domain of convex polyhedra, this work has just started. We have extended the *Parma Polyhedra Library* (PPL) [4,7], a modern C++ library for the manipulation of convex polyhedra, with a prototype implementation of the certificate-based widening and its variant employing the 'widening up to' technique [26,27]. The current prototype, which is based on the new widening operator defined in [5,6], tries two upper bound operators for the powerset domain: the first one is the least upper bound; the second one is the extrapolation heuristics H_{P}^{∇} (but instantiated using the widening in [5,6]). The initial results obtained are very encouraging as our new widening compares favorably, both in terms of precision and efficiency, with the extrapolation operator of [11]. By exploiting the genericity of the finite powerset construction, it was also possible to experiment with the certificate-based widening on other base-level domains provided by the PPL, including the domain of bounded difference shapes [3] and a domain of *rational grids* (i.e., sets defined by systems of relational congruences in the style of [24]).

In the examples previously shown and in the experiments conducted so far, the finite powerset construction has been applied only to those base-level abstract domains that encode numerical information. This was mainly for ease of presentation and also because of the availability of implementations for the chosen domains; there is nothing in the considered generic constructions that prevents one from applying them to domains that encode symbolic information. However, the reader should remember that the finite powerset construction is meant to provide a restricted form of disjunctive information, where different disjuncts are explicitly maintained as base-level abstract elements. As a consequence, it is unlikely that the finite powerset construction could really compete with *ad-hoc* disjunctive constructions, as provided by, e.g., BDD structures [12] or type graphs [34].

References

1. S. Abramsky and A. Jung. Domain theory. In S. Abramsky, D. M. Gabbay, and T. S. E. Maibaum, editors, *Handbook of Logic in Computer Science*, volume 3, chapter 1, pages 1–168. Clarendon Press, Oxford, UK, 1994.
2. R. Bagnara. A hierarchy of constraint systems for data-flow analysis of constraint logic-based languages. *Science of Computer Programming*, 30(1–2):119–155, 1998.
3. R. Bagnara, P. M. Hill, E. Mazzi, and E. Zaffanella. Widening operators for weakly-relational numeric abstractions. In C. Hankin and I. Silveroni, editors, *Static Analysis: Proceedings of the 12th International Symposium*, volume 3672 of *Lecture Notes in Computer Science*, pages 3–18, London, UK, 2005. Springer-Verlag, Berlin.
4. R. Bagnara, P. M. Hill, E. Ricci, and E. Zaffanella. *The Parma Polyhedra Library User's Manual*. Department of Mathematics, University of Parma, Parma, Italy, release 0.5 edition, April 2003. Available at `http://www.cs.unipr.it/ppl/`.
5. R. Bagnara, P. M. Hill, E. Ricci, and E. Zaffanella. Precise widening operators for convex polyhedra. In R. Cousot, editor, *Static Analysis: Proceedings of the 10th International Symposium*, volume 2694 of *Lecture Notes in Computer Science*, pages 337–354, San Diego, California, USA, 2003. Springer-Verlag, Berlin.
6. R. Bagnara, P. M. Hill, E. Ricci, and E. Zaffanella. Precise widening operators for convex polyhedra. *Science of Computer Programming*, 58(1–2):28–56, 2005.
7. R. Bagnara, E. Ricci, E. Zaffanella, and P. M. Hill. Possibly not closed convex polyhedra and the Parma Polyhedra Library. In M. V. Hermenegildo and G. Puebla, editors, *Static Analysis: Proceedings of the 9th International Symposium*, volume 2477 of *Lecture Notes in Computer Science*, pages 213–229, Madrid, Spain, 2002. Springer-Verlag, Berlin.
8. F. Besson, T. P. Jensen, and J.-P. Talpin. Polyhedral analysis for synchronous languages. In A. Cortesi and G. Filé, editors, *Static Analysis: Proceedings of the 6th International Symposium*, volume 1694 of *Lecture Notes in Computer Science*, pages 51–68, Venice, Italy, 1999. Springer-Verlag, Berlin.
9. G. Birkhoff. *Lattice Theory*, volume XXV of *Colloquium Publications*. American Mathematical Society, Providence, Rhode Island, USA, third edition, 1967.
10. F. Bourdoncle. Abstract interpretation by dynamic partitioning. *Journal of Functional Programming*, 2(4):407–435, 1992.
11. T. Bultan, R. Gerber, and 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):747–789, 1999.
12. A. Cortesi, G. Filé, and W. Winsborough. *Prop* revisited: Propositional formula as abstract domain for groundness analysis. In *Proceedings, Sixth Annual IEEE Symposium on Logic in Computer Science*, pages 322–327, Amsterdam, The Netherlands, 1991. IEEE Computer Society Press.
13. A. Cortesi, B. Le Charlier, and P. Van Hentenryck. Combinations of abstract domains for logic programming: Open product and generic pattern construction. *Science of Computer Programming*, 38(1–3):27–71, 2000.

14. P. Cousot and R. Cousot. Static determination of dynamic properties of programs. In B. Robinet, editor, *Proceedings of the Second International Symposium on Programming*, pages 106–130, Paris, France, 1976. Dunod, Paris, France.
15. P. Cousot and R. Cousot. Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints. In *Proceedings of the Fourth Annual ACM Symposium on Principles of Programming Languages*, pages 238–252, New York, 1977. ACM Press.
16. P. Cousot and R. Cousot. Systematic design of program analysis frameworks. In *Proceedings of the Sixth Annual ACM Symposium on Principles of Programming Languages*, pages 269–282, New York, 1979. ACM Press.
17. P. Cousot and R. Cousot. Abstract interpretation and applications to logic programs. *Journal of Logic Programming*, 13(2&3):103–179, 1992.
18. P. Cousot and R. Cousot. Abstract interpretation frameworks. *Journal of Logic and Computation*, 2(4):511–547, 1992.
19. P. Cousot and R. Cousot. Comparing the Galois connection and widening/narrowing approaches to abstract interpretation. In M. Bruynooghe and M. Wirsing, editors, *Proceedings of the 4th International Symposium on Programming Language Implementation and Logic Programming*, volume 631 of *Lecture Notes in Computer Science*, pages 269–295, Leuven, Belgium, 1992. Springer-Verlag, Berlin.
20. P. Cousot and N. Halbwachs. Automatic discovery of linear restraints among variables of a program. In *Conference Record of the Fifth Annual ACM Symposium on Principles of Programming Languages*, pages 84–96, Tucson, Arizona, 1978. ACM Press.
21. G. Delzanno and A. Podelski. Model checking in CLP. In R. Cleaveland, editor, *Tools and Algorithms for Construction and Analysis of Systems, 5th International Conference, TACAS '99*, volume 1579 of *Lecture Notes in Computer Science*, pages 223–239, Amsterdam, The Netherlands, 1999. Springer-Verlag, Berlin.
22. N. Dershowitz and Z. Manna. Proving termination with multiset orderings. *Communications of the ACM*, 22(8):465–476, 1979.
23. G. Filé and F. Ranzato. The powerset operator on abstract interpretations. *Theoretical Computer Science*, 222:77–111, 1999.
24. P. Granger. Static analyses of congruence properties on rational numbers (extended abstract). In P. Van Hentenryck, editor, *Static Analysis: Proceedings of the 4th International Symposium*, volume 1302 of *Lecture Notes in Computer Science*, pages 278–292, Paris, France, 1997. Springer-Verlag, Berlin.
25. N. Halbwachs. *Détermination Automatique de Relations Linéaires Vérifiées par les Variables d'un Programme.* Thèse de $3^{\text{ème}}$ cycle d'informatique, Université scientifique et médicale de Grenoble, Grenoble, France, March 1979.
26. N. Halbwachs. Delay analysis in synchronous programs. In C. Courcoubetis, editor, *Computer Aided Verification: Proceedings of the 5th International Conference*, volume 697 of *Lecture Notes in Computer Science*, pages 333–346, Elounda, Greece, 1993. Springer-Verlag, Berlin.
27. N. Halbwachs, Y.-E. Proy, and P. Roumanoff. Verification of real-time systems using linear relation analysis. *Formal Methods in System Design*, 11(2):157–185, 1997.
28. C. Holzbaur. OFAI clp(q,r) manual, edition 1.3.3. Technical Report TR-95-09, Austrian Research Institute for Artificial Intelligence, Vienna, 1995.
29. W. Kelly, V Maslov, W. Pugh, E. Rosser, T. Shpeisman, and D. Wonnacott. The Omega library interface guide. Technical Report CS-TR-3445, Department of Computer Science, University of Maryland, College Park, MD, USA, 1995.
30. H. Le Verge. A note on Chernikova's algorithm. *Publication interne* 635, IRISA, Campus de Beaulieu, Rennes, France, 1992.
31. V. Loechner. *PolyLib*: A library for manipulating parameterized polyhedra. Available at `http://icps.u-strasbg.fr/~loechner/polylib/`, March 1999. Declares itself to be a continuation of [35].
32. W. Pugh. A practical algorithm for exact array dependence analysis. *Communications of the ACM*, 35(8):102–114, 1992.
33. D. Srivastava. Subsumption and indexing in constraint query languages with linear arithmetic constraints. *Annals of Mathematics and Artificial Intelligence*, 8(3–4):315–343, 1993.
34. P. Van Hentenryck, A. Cortesi, and B. Le Charlier. Type analysis of Prolog using type graphs. *Journal of Logic Programming*, 22(3):179–209, 1995.
35. D. K. Wilde. A library for doing polyhedral operations. Master's thesis, Oregon State University, Corvallis, Oregon, December 1993. Also published as IRISA *Publication interne* 785, Rennes, France, 1993.

A Proofs

Proof (of Proposition 1). Using the notation of Definition 6, we show that $H^{\nabla}_{\mathrm{P}}(S_1, S_2) = S_2 \oplus_{\mathrm{P}} \Omega^{\vdash}_{D}(S)$, where

$$S := \{\, d_1 \nabla d_2 \in D \mid d_1 \in S_1, d_2 \in S_2, d_1 \Vdash d_2 \,\},$$

satisfies properties (6) and (7) of Definition 6. Note that, by Definition 6, we have $S_1 \Vdash_{\mathrm{P}} S_2$, so that $S_2 \neq \varnothing$.

Let $S' := S_2 \cup \Omega^{\vdash}_{D}(S)$; then, by Definitions 5 and 7, we obtain

$$\begin{aligned} H^{\nabla}_{\mathrm{P}}(S_1, S_2) &= S_2 \oplus_{\mathrm{P}} \Omega^{\vdash}_{D}(S) \\ &= \Omega^{\vdash}_{D}\bigl(S_2 \cup \Omega^{\vdash}_{D}(S)\bigr) \\ &= \Omega^{\vdash}_{D}(S'). \end{aligned} \tag{13}$$

We will prove that

$$S_2 \vdash_{\mathrm{P}} \Omega^{\vdash}_{D}(S'); \tag{14}$$

$$\forall d \in \Omega^{\vdash}_{D}(S') : \exists d_2 \in S_2 \,.\, d_2 \vdash d; \tag{15}$$

$$\forall d \in \Omega^{\vdash}_{D}(S') \setminus S_2 : \exists d_1 \in S_1 \,.\, d_1 \Vdash_{\nabla} d. \tag{16}$$

Then, using equation (13), property (6) of Definition 6 follows from properties (14) and (15) and property (7) of Definition 6 follows from property (16).

To prove property (14), let $d_2 \in S_2$; then, since $S_2 \subseteq S'$, we also have $d_2 \in S'$; hence, by Definition 4, there exists $d_2' \in \Omega^{\vdash}_{D}(S')$ such that $d_2 \vdash d_2'$. Therefore property (14) holds.

To prove properties (15) and (16), let $d \in \Omega^{\vdash}_{D}(S') \setminus S_2$; then, by Definition 4, $d \in S'$ and, as $d \notin S_2$, again by Definition 4, $d \in S$. Thus there exist $d_1 \in S_1$ and $d_2 \in S_2$ such that $d = d_1 \nabla d_2$ and hence $d_1 \Vdash_{\nabla} d$. Therefore property (16) holds. Moreover, as '∇' is a widening on $\hat{D}$, $d_2 \vdash d$. Therefore property (15) holds. □

Proof (of Proposition 2). Assuming the notation and hypothesis introduced in Definition 9, we show that $\Omega^{\nabla}_{D}(S_1, S_2)$ satisfies properties (6) and (7) of Definition 6 and property (8) of Definition 8. Let $S = S_1 \cup S_2$, so that $\Omega^{\nabla}_{D}(S_1, S_2) = \Omega^{\nabla}_{D}(S)$. By Definition 9, there exist $m \in \mathbb{N}$ and a sequence $T_0, \ldots, T_m$ in $\wp_{\mathrm{f}}(D)$ where $T_0 = S$, $T_m = \Omega^{\nabla}_{D}(S)$ and, for each $0 < i \leq m$, there exist $d, d' \in T_{i-1}$ such that $d \Vdash d'$ and $T_i = \bigl(T_{i-1} \setminus \{d, d'\}\bigr) \cup \{d \nabla d'\}$. Thus, for all $d \in T_{i-1}$ there exists $d' \in T_i$ such that $d \vdash_{\nabla} d'$. We prove, for all $0 \leq i \leq m$, the following properties hold:

$$\forall d \in S_2 : \exists d_i \in T_i \;.\; d \vdash d_i; \tag{17}$$

$$\forall d_i \in T_i : \exists d \in S_2 \;.\; d \vdash d_i; \tag{18}$$

$$\forall d_i \in T_i \setminus S : \exists d \in S_1 \;.\; d \Vdash_{\nabla} d_i; \tag{19}$$

$$\forall d \in S_1 : \exists d_i \in T_i \;.\; d \vdash_{\nabla} d_i. \tag{20}$$

Letting $i = m$ in properties (17) and (18) we obtain $S_2 \vdash_{\mathrm{EM}} \Omega^{\nabla}_{D}(S)$, so that property (6) in Definition 6 holds. Since $S_1 \vdash_{\mathrm{P}} S_2$ and $T_m \in \wp^{\vdash}_{\mathrm{fn}}(D)$, we have $T_m \setminus S = T_m \setminus S_2$; thus, letting $i = m$ in property (19), we obtain that property (7) in Definition 6 holds. Thus Ω^{∇}_{D} is an extrapolation heuristics for $\hat{D}_{\mathrm{P}}$. Moreover, letting $i = m$ in property (20), we obtain that property (8) in Definition 8 holds, so that Ω^{∇}_{D} is ∇-covered.

We now prove the four properties by induction on i. For the base case, we have $i = 0$ and $T_0 = S = S_1 \cup S_2$, so that all the properties hold trivially. For the inductive case, we have $m > 0$ and assume that $i > 0$. By Definition 9, if $d_{i-1} \in T_{i-1}$, then there exists $d_i \in T_i$ such that either $d_{i-1} = d_i$ or there exists $d_{i-1}' \in T_{i-1}$ such that $d_{i-1} \Vdash d_{i-1}' \neq d_i$ and $d_i = d_{i-1} \nabla d_{i-1}'$; in both cases, $d_{i-1} \vdash_{\nabla} d_i$. Thus, assuming properties (17), (18), (19) and (20) hold for $i - 1$, they also hold for i. □

Proof (of Theorem 1). We first prove condition (1) of Definition 1, i.e., $S_2 \vdash_{\mathrm{P}} S_1 \mathbin{{}_{k}\nabla_{\mathrm{P}}} S_2$. Assume the notation and the hypotheses introduced in Definition 11 and let $T := S_1 \mathbin{{}_{k}\nabla_{\mathrm{P}}} S_2 = h^{\nabla}_{\mathrm{P}}(S_1, \Uparrow_k S_2)$. By Definition 10, $S_2 \vdash_{\mathrm{P}} \Uparrow_k S_2$ and, by Definition 6, $\Uparrow_k S_2 \vdash_{\mathrm{P}} T$ so that, by transitivity of '$\vdash_{\mathrm{P}}$', $S_2 \vdash_{\mathrm{P}} T$.

We now prove condition (2) holds in Definition 1. Suppose $T_0 \vdash_{\mathrm{P}} T_1 \vdash_{\mathrm{P}} \cdots$ is an increasing chain of elements in $\wp^{\vdash}_{\mathrm{fn}}(D)$ and consider the sequence defined by $U_0 := T_0$ and, for each $i > 0$,

$$U_i := U_{i-1} \mathbin{{}_{k}\nabla_{\mathrm{P}}} T_i',$$

where $T_i' = U_{i-1} \oplus_{\mathrm{P}} T_i$. Suppose that $i > 0$. By Definition 5, $U_{i-1} \vdash_{\mathrm{P}} T_i'$. By the first part of the proof, $T_i' \vdash_{\mathrm{P}} U_i$ so that, by transitivity of '$\vdash_{\mathrm{P}}$', $U_{i-1} \vdash_{\mathrm{P}} U_i$. Thus $U_0 \vdash_{\mathrm{P}} U_1 \vdash_{\mathrm{P}} \cdots$ is another increasing chain in $\wp^{\vdash}_{\mathrm{fn}}(D)$.

Consider any $j \geq 0$ such that $U_j \neq \varnothing$ and suppose $d_j \in U_j$. By Definition 11, we have $U_{j+1} = h^{\nabla}_{\mathrm{P}}(U_j, \Uparrow_k T_j')$. By condition (8) in Definition 8, there exists $d_{j+1} \in U_{j+1}$ such that $d_j \vdash_{\nabla} d_{j+1}$. By transitivity, for all $i > j$, there exists $d_i \in U_i$ such that $d_j \vdash_{\nabla} d_i$. As the '$\Vdash_{\nabla}$' relation satisfies the ascending chain condition, there exist $m \in \mathbb{N}$ and $d_m \in U_m$ such that $d_j \vdash_{\nabla} d_m$ and, for all $i \geq m$, $d_m \in U_i$.

Reasoning towards a contradiction, suppose that the widened sequence does not converge in a finite number of steps. Then, by the point above, there exists $\ell \in \mathbb{N}$ and $U \in \wp^{\vdash}_{\mathrm{fn}}(D)$ where $\# U > k$ and $U \subseteq U_i$ for all $i \geq \ell$. In particular, we have $U \subseteq U_\ell \cap U_{\ell+1}$. By condition (7) in Definition 6, $U \subseteq \Uparrow_k T_{\ell+1}'$ so that $\#(\Uparrow_k T_{\ell+1}') > k$ contradicting Definition 10. Thus the widened sequence converges in a finite number of steps. □

Proof (of Proposition 3). By Proposition 1, H^{∇}_{P} is an extrapolation heuristics for $\hat{D}_{\mathrm{P}}$. Therefore, using the notation of Definition 6, it remains to show that $H^{\nabla}_{\mathrm{P}}(S_1, S_2) = S_2 \oplus_{\mathrm{P}} \Omega^{\vdash}_{D}(S)$ where

$$S := \{\, d_1 \nabla d_2 \in D \mid d_1 \in S_1, d_2 \in S_2, d_1 \Vdash d_2 \,\}$$

satisfies property (9) of Definition 12. In Proposition 1, we have shown that equation (13) holds.

Assume that $d \in H^{\nabla}_{\mathrm{P}}(S_1, S_2) \cap S_2$ and $d_1 \in S_1$ are such that $d_1 \Vdash d$. Then $d_1 \nabla d$ is defined and is in S. By Definition 4, there exists $d' \in \Omega^{\vdash}_{D}(S)$ such that $d_1 \nabla d \vdash d'$; also, as '∇' is a widening, $d \vdash d_1 \nabla d$. Again by Definition 4, there exists $d'' \in \Omega^{\vdash}_{D}\bigl(S_2 \cup \Omega^{\vdash}_{D}(S)\bigr) = H^{\nabla}_{\mathrm{P}}(S_1, S_2)$ such that $d' \vdash d''$. To summarize,

$$d \vdash d_1 \nabla d \vdash d' \vdash d''.$$

Since we observed that both $d, d'' \in H^{\nabla}_{\mathrm{P}}(S_1, S_2) \in \wp^{\vdash}_{\mathrm{fn}}(D)$, non-redundancy implies that $d = d''$, so that

$$d = d_1 \nabla d = d' = d''.$$

Since $d' \in S$, we also have $d \in S$. Thus, there exist $d_1' \in S_1$ and $d_2 \in S_2$ such that $d = d_1' \nabla d_2$ and hence $d_1' \Vdash_{\nabla} d$, proving property (9) of Definition 12. □

Proof (of Theorem 2). We first prove condition (1) in Definition 1, i.e., $S_2 \vdash_{\mathrm{P}} S_1 \mathbin{{}_{\mathrm{EM}}\nabla_{\mathrm{P}}} S_2$. Assume the notation and the hypotheses introduced in Definition 13 and let $T := S_1 \mathbin{{}_{\mathrm{EM}}\nabla_{\mathrm{P}}} S_2 = h^{\nabla}_{\mathrm{P}}(S_1, S_2')$. As '$\boxplus_{\mathrm{EM}}$' is a connector, we have $S_2 \vdash_{\mathrm{EM}} S_1 \boxplus_{\mathrm{EM}} S_2$. Thus, in both the cases of the definition of S_2', $S_2 \vdash_{\mathrm{EM}} S_2'$, which implies $S_2 \vdash_{\mathrm{P}} S_2'$. Moreover, by Definition 6, $S_2' \vdash_{\mathrm{P}} T$ so that, by transitivity of '$\vdash_{\mathrm{P}}$', $S_2 \vdash_{\mathrm{P}} T$.

We now prove condition (2) holds in Definition 1. Suppose $T_0 \vdash_{\mathrm{P}} T_1 \vdash_{\mathrm{P}} \cdots$ is an increasing chain of elements in $\wp^{\vdash}_{\mathrm{fn}}(D)$ and consider the sequence defined by $U_0 := T_0$ and $U_i := U_{i-1} \mathbin{{}_{\mathrm{EM}}\nabla_{\mathrm{P}}} (U_{i-1} \oplus_{\mathrm{P}} T_i)$, for each $i > 0$.

As we have already shown that condition (1) in Definition 1 holds, $(U_{i-1} \oplus_{\mathrm{P}} T_i) \vdash_{\mathrm{P}} U_i$ so that, by transitivity of '$\vdash_{\mathrm{P}}$', $U_{i-1} \vdash_{\mathrm{P}} U_i$. Thus $U_0 \vdash_{\mathrm{P}} U_1 \vdash_{\mathrm{P}} \cdots$ is another increasing chain in $\wp_{\mathrm{fn}}^{\vdash}(D)$. We need to show that the widened sequence converges in a finite number of steps.

For each $i > 0$, consider the successive widened iterates U_{i-1} and U_i, so that, according to Definition 13, we can write $U_i = h_{\mathrm{P}}^{\nabla}(U_{i-1}, S_2')$, where in both the cases for the definition of S_2' we have $U_{i-1} \vdash_{\mathrm{EM}} S_2'$. Since h_{P}^{∇} is a ∇-connected extrapolation heuristics, by property (6) of Definition 6, we have $S_2' \vdash_{\mathrm{EM}} U_i$ and, by transitivity of '$\vdash_{\mathrm{EM}}$', $U_{i-1} \vdash_{\mathrm{EM}} U_i$. Moreover, in the above context, the properties (7) of Definition 6 and (9) of Definition 12 can be rewritten to the simpler property:

$$\forall d' \in U_i : \exists d \in U_{i-1} \,.\, d \vdash_{\nabla} d'. \tag{21}$$

Let $W_i \subseteq D \times D$ be defined so that $(d, d') \in W_i$ holds if and only if $d \in U_{i-1}$, $d' \in U_i$ and $d \Vdash_{\nabla} d'$. Thus, by property (21), we have

$$\forall d' \in U_i \setminus U_{i-1} : \exists d \in U_{i-1} \,.\, (d, d') \in W_i. \tag{22}$$

For each $i \in \mathbb{N}$, consider the finite directed graph $G_i = (V_i, E_i)$, where

- the set of vertices $V_i \subseteq D$ is $V_i := \bigcup \{\, U_j \mid 0 \leq j \leq i \,\}$;
- the set of edges $E_i \subseteq V_i \times V_i$ is
$$E_i := \bigcup \{\, W_j \mid 0 < j \leq i \,\}.$$

Furthermore, consider the (*a priori*, possibly infinite) graph $G = (V, E)$ such that

$$V = \bigcup_{i \geq 0} V_i; \qquad E = \bigcup_{i \geq 0} E_i = \bigcup_{i \geq 1} W_i.$$

We will now show that G is a finite and acyclic graph, so that, by property (22), '${}_{\mathrm{EM}}\nabla_{\mathrm{P}}$' is a widening. Namely, we will prove the following properties for the graph G, which combined together imply that G is a finite and acyclic graph:

1. G has no infinite paths;
2. G has a finite number of connected components;
3. G is finitely branching, i.e., each vertex has finite outdegree.

To prove G has no infinite paths, suppose $p := d_0 \to d_1 \to \cdots \to d_i \to \cdots$ is a (possibly infinite) path in G. By the definition of G, if (d_{k-1}, d_k) is an edge in p for some $k > 0$, then there exists an index $j > 0$ such that $(d_{k-1}, d_k) \in W_j$. By definition of W_j, we know that $d_{k-1} \Vdash_{\nabla} d_k$. Thus, we have a strictly increasing sequence $d_0 \Vdash_{\nabla} d_1 \Vdash_{\nabla} \cdots \Vdash_{\nabla} d_i \Vdash_{\nabla} \cdots$ and hence, as '$\Vdash_{\nabla}$' satisfies the ascending chain condition, the path p must be finite.

We now prove that the graph G has a finite number of connected components. Consider, for any $i > 0$ the graph $G_i = (V_i, E_i)$. Then, by property (22), for each vertex d_i in V_i either $d_i \in V_{i-1}$ or there is an edge (d_{i-1}, d_i) in E_i where $d_{i-1} \in U_{i-1} \subseteq V_{i-1}$. Thus, for all $i \in \mathbb{N}$, the number of components of G_i is no more than the number of components of G_{i-1}. As the number of components of G_0 is $\# U_0$, the number of components of G is no more than $\# U_0$.

Finally, to prove that G is finitely branching, consider any vertex $d \in V$. Suppose that, for some index $i > 0$, there exists $(d, d') \in E_i \setminus E_{i-1}$ (note that $(d, d') \notin E_0$ because $E_0 = \varnothing$). Then $(d, d') \in W_i$. Thus, by definition of W_i, $d \in U_{i-1}$ and $d' \in U_i \setminus U_{i-1}$ and $d \Vdash d'$. However, for all indices $j \geq i$, as $U_i \vdash_{\mathrm{P}} U_j$, there exists $d_j \in U_j$ such that $d' \vdash d_j$ so that $d \Vdash d_j$; as U_j is a non-redundant set (in the sense of Definition 4), $d \notin U_j$. Thus all outgoing edges from d are in E_i. As the set E_i is finite, d has a finite number of outgoing edges. □

In order to prove Proposition 4, we first define a minor variant (a coarsening) of the '$\curvearrowright_{\mathrm{P}}$' relation and show that it satisfies the ascending chain condition.

Definition 19. ('$\curvearrowright_{\mathrm{L}}$'.) Let $(\mathcal{O}, \preceq, \mu)$ be a finite convergence certificate for the widening operator '∇' on $\hat{D}$. The relation $\curvearrowright_{\mathrm{L}} \subseteq \wp_{\mathrm{fn}}^{\vdash}(D) \times \wp_{\mathrm{fn}}^{\vdash}(D)$ induced by μ is such that, for each $S_1, S_2 \in \wp_{\mathrm{fn}}^{\vdash}(D)$, $S_1 \curvearrowright_{\mathrm{L}} S_2$ holds if and only if either one of the following conditions holds:

$$\mu\bigl(\textstyle\bigoplus S_1\bigr) \prec \mu\bigl(\textstyle\bigoplus S_2\bigr);$$
$$\mu\bigl(\textstyle\bigoplus S_1\bigr) = \mu\bigl(\textstyle\bigoplus S_2\bigr) \land \tilde{\mu}(S_1) \mathrel{\prec\!\!\prec} \tilde{\mu}(S_2).$$

Lemma 1. *The '$\curvearrowright_{\mathrm{L}}$' relation on $\hat{D}_{\mathrm{P}}$ satisfies the ascending chain condition.*

Proof. By assumption, $(\mathcal{O}, \preceq, \mu)$ is a finite convergence certificate for the base-level widening operator '∇', so that $\langle \mathcal{O}, \preceq \rangle$ satisfies the ascending chain condition. As noted in Section 2, the induced poset $\bigl\langle \mathcal{M}(\mathcal{O}), \mathrel{\preceq\!\!\preceq} \bigr\rangle$ also satisfies the ascending chain condition. As a consequence, the lexicographic product of '$\preceq$' and '$\mathrel{\preceq\!\!\preceq}$' satisfies the ascending chain condition on the product $\mathcal{O} \times \mathcal{M}(\mathcal{O})$. Note that, by Definition 19, $S_1 \curvearrowright_{\mathrm{L}} S_2$ holds if and only if there is a strict increase in this lexicographic product ordering, so that '$\curvearrowright_{\mathrm{L}}$' satisfies the ascending chain condition. □

Proof (of Proposition 4). By hypotheses, the certificate $(\mathcal{O}, \preceq, \mu)$ is such that both '$\preceq$' and μ are finitely computable. Thus, the finite computability of the relation '$\curvearrowright_{\mathrm{P}}$' is an easy consequence of the way it is defined and the fact that we only consider finite sets and multisets.

We first show that, for all $S_i, S_{i+1}, S_{i+2} \in \hat{D}_{\mathrm{P}}$,

$$S_i \curvearrowright_{\mathrm{P}} S_{i+1} \curvearrowright_{\mathrm{P}} S_{i+2} \implies (S_i \curvearrowright_{\mathrm{L}} S_{i+1}) \lor (S_i \curvearrowright_{\mathrm{L}} S_{i+2}). \tag{23}$$

There are three cases corresponding to the three conditions in Definition 14. If $S_i \curvearrowright_{\mathrm{P}} S_{i+1}$ holds by virtue of condition (10), then $\mu\bigl(\bigoplus S_i\bigr) \prec \mu\bigl(\bigoplus S_{i+1}\bigr)$, which implies $S_i \curvearrowright_{\mathrm{L}} S_{i+1}$. Similarly, if $S_i \curvearrowright_{\mathrm{P}} S_{i+1}$ holds by virtue of condition (12), then both $\mu\bigl(\bigoplus S_i\bigr) = \mu\bigl(\bigoplus S_{i+1}\bigr)$ and $\tilde{\mu}(S_i) \mathrel{\prec\mkern-5mu\prec} \tilde{\mu}(S_{i+1})$, which again implies $S_i \curvearrowright_{\mathrm{L}} S_{i+1}$. Otherwise, $S_i \curvearrowright_{\mathrm{P}} S_{i+1}$ must hold by virtue of condition (11) so that $\mu\bigl(\bigoplus S_i\bigr) = \mu\bigl(\bigoplus S_{i+1}\bigr)$ and $\# S_{i+1} = 1$. However, $S_{i+1} \curvearrowright_{\mathrm{P}} S_{i+2}$ also holds and, as $\# S_{i+1} = 1$, this may only happen by virtue of condition (10) so that $\mu\bigl(\bigoplus S_{i+1}\bigr) \prec \mu\bigl(\bigoplus S_{i+2}\bigr)$. Thus,

$$\mu\bigl(\textstyle\bigoplus S_i\bigr) = \mu\bigl(\textstyle\bigoplus S_{i+1}\bigr) \prec \mu\bigl(\textstyle\bigoplus S_{i+2}\bigr)$$

and $S_i \curvearrowright_{\mathrm{L}} S_{i+2}$.

We prove that '$\curvearrowright_{\mathrm{P}}$' satisfies the ascending chain condition by contraposition; thus we suppose that there is an infinite chain $S_0 \curvearrowright_{\mathrm{P}} S_1 \curvearrowright_{\mathrm{P}} \cdots \curvearrowright_{\mathrm{P}} S_i \curvearrowright_{\mathrm{P}} \cdots$ of abstract elements in the finite powerset domain $\hat{D}_{\mathrm{P}}$. Then by (23), there exists a sequence $0 \leq j_1 \leq \cdots \leq j_i \leq \cdots$ such that $S_0 \curvearrowright_{\mathrm{L}} S_{j_1} \curvearrowright_{\mathrm{L}} \cdots \curvearrowright_{\mathrm{L}} S_{j_i} \curvearrowright_{\mathrm{L}} \cdots$ and, for all $i \in \mathbb{N}, j_i \geq i$. Thus this sequence is also infinite which contradicts the result of Lemma 1 that '$\curvearrowright_{\mathrm{L}}$' satisfies the ascending chain condition. □

Proof (of Theorem 3). Let $S_1, S_2 \in \wp_{\mathrm{fn}}^{\vdash}(D)$, where $S_1 \Vdash_{\mathrm{P}} S_2$ and let $T := S_1 \; {}_{\mu}\nabla_{\mathrm{P}} \; S_2$. Let $S := S_1 \boxplus_{\mathrm{P}} S_2$. Then, as '$\boxplus_{\mathrm{P}}$' is an upper bound operator on $\hat{D}_{\mathrm{P}}$ we have

$$S_1 \Vdash_{\mathrm{P}} S_2 \vdash_{\mathrm{P}} S. \tag{24}$$

We first prove that condition (1) in Definition 1 holds, i.e., $S_2 \vdash_{\mathrm{P}} T$. Consider each of the three cases in Definition 16 separately. If the first case applies, then $T = S$ and the result holds by (24). If the second case applies, then $T = S \oplus_{\mathrm{P}} \{d\}$ so that as '$\oplus_{\mathrm{P}}$' is the least upper bound operator, the result follows again by (24). If the third and last case applies, then $T = \bigl\{\bigoplus S_2\bigr\}$, so that the result holds trivially by definition of '$\vdash_{\mathrm{P}}$'.

We now prove that condition (2) in Definition 1 holds. By Proposition 4, '$\curvearrowright_{\mathrm{P}}$' satisfies the ascending chain condition; hence, to complete the proof it is sufficient to show that $S_1 \curvearrowright_{\mathrm{P}} T$. Consider each of the three cases in Definition 16. If the first case is applied, then the applicability condition trivially ensures that $S_1 \curvearrowright_{\mathrm{P}} T$.

If the second case is applied, then $T = S \oplus_{\mathrm{P}} \{d\}$, where

$$\begin{aligned} d &:= d_1 \ominus d_2; \\ d_1 &:= \textstyle\bigoplus S_1 \nabla \bigoplus S; \\ d_2 &:= \textstyle\bigoplus S. \end{aligned}$$

Note that the applicability condition $\bigoplus S_1 \Vdash \bigoplus S$ for this case ensures that the base-level widening application in the computation of the abstract element $d_1 \in D$ is well defined. Moreover, since '∇' is an upper bound operator on $\hat{D}$, we have $d_2 \vdash d_1$, so that also the subtraction application in the computation of the abstract element $d \in D$ is well defined. By Definition 15, we know that $d_1 = (d_1 \ominus d_2) \oplus d_2$. As a consequence, we obtain

$$\begin{aligned} \textstyle\bigoplus T &= \textstyle\bigoplus\bigl(S \oplus_{\mathrm{P}} \{d\}\bigr) \\ &= d \oplus \bigl(\textstyle\bigoplus S\bigr) \\ &= d \oplus d_2 \\ &= (d_1 \ominus d_2) \oplus d_2 \\ &= d_1 \\ &= \textstyle\bigoplus S_1 \nabla \bigoplus S. \end{aligned}$$

Since μ is a certificate for the base-level widening '∇', we obtain

$$\mu\bigl(\textstyle\bigoplus S_1\bigr) \prec \mu\bigl(\textstyle\bigoplus S_1 \nabla \bigoplus S\bigr) = \mu\bigl(\textstyle\bigoplus T\bigr),$$

so that by condition (10) of Definition 14, $S_1 \curvearrowright_{\mathrm{P}} T$.

Finally, suppose that the last case is applied. Then $T = \bigl\{\bigoplus S_2\bigr\}$ so that $\bigoplus T = \bigoplus S_2$. It follows from (24) that $\bigoplus S_1 \vdash \bigoplus T \vdash \bigoplus S$. As the condition for the second case of Definition 16 does not hold, $\bigoplus S_1 = \bigoplus S$, which implies $\bigoplus S_1 = \bigoplus T$ and $\mu\bigl(\bigoplus S_1\bigr) = \mu\bigl(\bigoplus T\bigr)$. By (24), $S_1 \Vdash_{\mathrm{P}} T$ so that $\# S_1 > 1$. Since $\# T = 1$, condition (11) of Definition 14 is satisfied and $S_1 \curvearrowright_{\mathrm{P}} T$. □

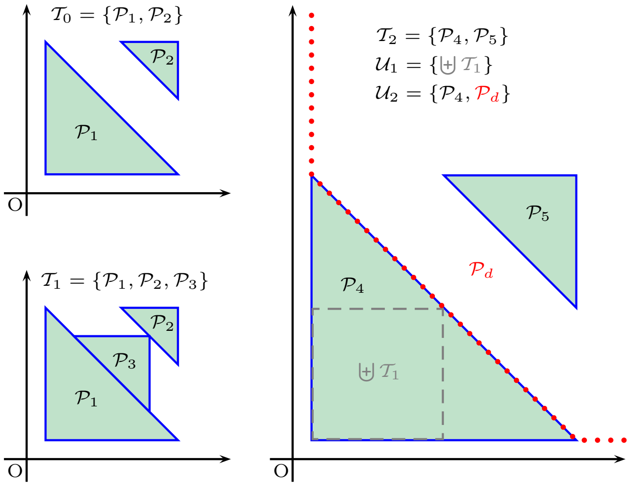
$\mathcal{T}_0 = \{\mathcal{P}_1, \mathcal{P}_2\}$
$\mathcal{P}_2$
$\mathcal{P}_1$
O
$\mathcal{T}_1 = \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3\}$
$\mathcal{P}_2$
$\mathcal{P}_3$
$\mathcal{P}_1$
O
$\mathcal{T}_2 = \{\mathcal{P}_4, \mathcal{P}_5\}$
$\mathcal{U}_1 = \{\biguplus \mathcal{T}_1\}$
$\mathcal{U}_2 = \{\mathcal{P}_4, \mathcal{P}_d\}$
$\mathcal{P}_5$
$\mathcal{P}_d$
$\mathcal{P}_4$
$\biguplus \mathcal{T}_1$
O

Fig. 7. The condition $\# \mathcal{U}_1 > 1$ is needed to obtain a proper widening.

It should be noted that case (11) of Definition 14 has been introduced so as to ensure that $S_1 \curvearrowright_{\mathrm{P}} \bigl\{\bigoplus S_2\bigr\}$ holds in the last case of the specification of '${}_{\mu}\nabla_{\mathrm{P}}$', therefore inducing a strict decrease in the corresponding level mapping. This also made necessary the addition of the extra conditions on the cardinalities of S_1 and S_2 in case (12) of Definition 14, since without these the relation would have violated the ascending chain condition. This is illustrated by the following example.

Example 12. Consider the finite powerset domain $\bigl(\widehat{\mathbb{CP}_2}\bigr)_{\mathrm{P}}$, with the standard widening '∇_s' on the base-level domain $\mathbb{CP}_2$, certified by the level mapping μ_s defined in Definition 3 and the upper bound '$\boxplus_{\mathrm{P}}$' defined as '$\oplus_{\mathrm{P}}$', so that we will always have $S_1 \boxplus_{\mathrm{P}} S_2 = S_2$. Consider an iteration sequence $\mathcal{T}_0 \vdash_{\mathrm{P}} \mathcal{T}_1 \vdash_{\mathrm{P}} \mathcal{T}_2 \vdash_{\mathrm{P}} \cdots$ starting with elements

$$\begin{aligned} \mathcal{T}_0 &= \{\mathcal{P}_1, \mathcal{P}_2\}, \\ \mathcal{T}_1 &= \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3\}, \\ \mathcal{T}_2 &= \{\mathcal{P}_4, \mathcal{P}_5\}, \end{aligned}$$

as in the three diagrams in Figure 7. Then, the first three elements of widened sequence $\mathcal{U}_0 \vdash_{\mathrm{P}} \mathcal{U}_1 \vdash_{\mathrm{P}} \mathcal{U}_2 \vdash_{\mathrm{P}} \cdots$ can be computed as follows. First $\mathcal{U}_0 := \mathcal{T}_0$. Then, since $\mathcal{U}_0 \vdash_{\mathrm{P}} \mathcal{T}_1$, we have $\mathcal{U}_1 := \mathcal{U}_0 \;{}_{\mu}\nabla_{\mathrm{P}}\; \mathcal{T}_1 = \mathcal{T}_0 \;{}_{\mu}\nabla_{\mathrm{P}}\; \mathcal{T}_1$. As $\biguplus \mathcal{T}_0 = \biguplus \mathcal{T}_1$, $\# \mathcal{T}_1 > 1$ and

$$\begin{aligned} \tilde{\mu}_s(\mathcal{T}_0) &= \bigl\{(0,3),(0,3)\bigr\} \\ &\qquad \not\lll_s \bigl\{(0,3),(0,3),(0,3)\bigr\} = \tilde{\mu}_s(\mathcal{T}_1), \end{aligned}$$

the last case in Definition 16 applies so that $\mathcal{U}_1 = \bigl\{ \biguplus \mathcal{T}_1 \bigr\}$, as is indicated in the lower square in the right-hand diagram of Figure 7. Thus $\mathcal{U}_1 \vdash_{\mathrm{P}} \mathcal{T}_2$, so that the widened iterate $\mathcal{U}_2 := \mathcal{U}_1 \;{}_{\mu}\nabla_{\mathrm{P}}\; \mathcal{T}_2$ is defined. Since

$$\mu_s\bigl(\biguplus \mathcal{U}_1\bigr) = (0,4) = \mu_s\bigl(\biguplus \mathcal{T}_2\bigr)$$

but

$$\tilde{\mu}_s(\mathcal{U}_1) = \bigl\{(0,4)\bigr\} \lll_s \bigl\{(0,3),(0,3)\bigr\} = \tilde{\mu}_s(\mathcal{T}_2),$$

without the extra condition $\# \mathcal{U}_1 > 1$ in case (12) of Definition 14, we would apply the first case in Definition 16, obtaining $\mathcal{T}_2$ as the next element of the widened sequence. However, it can be seen that $\mathcal{T}_2$ has the same structure as $\mathcal{U}_0 = \mathcal{T}_0$ (the former being obtained from the latter by a suitable affine transformation) so that the sequence $\mathcal{T}_0, \mathcal{T}_1, \mathcal{T}_2$ and the corresponding "widened" sequence $\mathcal{U}_0, \mathcal{U}_1, \mathcal{T}_2$ can be extended indefinitely without obtaining convergence (in a finite number of steps). In contrast, since we do require the condition $\# \mathcal{U}_1 > 1$, the second case of Definition 16 applies and we compute $\mathcal{U}_2 = \mathcal{T}_2 \uplus_{\mathrm{P}} \{\mathcal{P}_d\} = \{\mathcal{P}_4, \mathcal{P}_d\}$, where $\mathcal{P}_d$ is the (unbounded) polyhedron indicated by the dotted lines in the right-hand diagram.

Proof (of Proposition 5). The finite powerset domain $(\widehat{\mathbb{CP}_n})_{\mathrm{P}}$ is related to the concrete domain $\hat{\mathsf{A}}_n$ defined in Section 2.2 by the concretization function $\gamma^{\mathsf{A}}_{\mathrm{P}}$ induced from γ^{A}, where $\gamma^{\mathsf{A}}(\mathcal{P}) = \mathcal{P}$ for each $\mathcal{P} \in \mathbb{CP}_n$. Namely, for each $\mathcal{S} \in \wp^{\subseteq}_{\mathrm{fn}}(\mathbb{CP}_n)$, we have $\gamma^{\mathsf{A}}_{\mathrm{P}}(\mathcal{S}) = \bigcup \mathcal{S}$. Therefore, we have to show that, for all $\mathcal{S}_1, \mathcal{S}_2 \in \wp^{\subseteq}_{\mathrm{fn}}(\mathbb{CP}_n)$, $\mathcal{S}_1 \bowtie \mathcal{S}_2$ if and only if $\bigcup \mathcal{S}_1 = \bigcup \mathcal{S}_2$.

First we assume that $\bigcup \mathcal{S}_1 \subseteq \bigcup \mathcal{S}_2$ and show that $\mathcal{S}_1 \triangleleft \mathcal{S}_2$. Consider an arbitrary element $\mathcal{S}_1' \in \wp^{\subseteq}_{\mathrm{fn}}(\mathbb{CP}_n)$ such that $\mathcal{S}_1' \vdash_{\mathrm{P}} \mathcal{S}_1$. By Definition 5, this implies $\bigcup \mathcal{S}_1' \subseteq \bigcup \mathcal{S}_1$ so that $\bigcup \mathcal{S}_1' \subseteq \bigcup \mathcal{S}_2$. Let

$$\mathcal{S}_1'' = \Omega^{\subseteq}_{\mathbb{CP}_n}\bigl(\{\, \mathcal{P}_1' \cap \mathcal{P}_2 \in \mathbb{CP}_n \mid \mathcal{P}_1' \in \mathcal{S}_1', \mathcal{P}_2 \in \mathcal{S}_2 \,\}\bigr)$$

so that, by Definitions 4 and 5, $\mathcal{S}_1'' \vdash_{\mathrm{P}} \mathcal{S}_2$. Moreover, by Definition 4, $\bigcup \mathcal{S}_1'' = \bigcup \mathcal{S}_1' \cap \bigcup \mathcal{S}_2$ which implies that $\bigcup \mathcal{S}_1' = \bigcup \mathcal{S}_1''$ so that $\biguplus \mathcal{S}_1' = \biguplus \mathcal{S}_1''$. Thus, by Definition 17, $\mathcal{S}_1 \triangleleft \mathcal{S}_2$. By a symmetric argument, we can prove that $\bigcup \mathcal{S}_2 \subseteq \bigcup \mathcal{S}_1$ implies $\mathcal{S}_2 \triangleleft \mathcal{S}_1$. Thus, again by Definition 17, we obtain that $\mathcal{S}_1 \equiv_{\gamma^{\mathsf{A}}_{\mathrm{P}}} \mathcal{S}_2$ implies $\mathcal{S}_1 \bowtie \mathcal{S}_2$.

Second we assume that $\bigcup \mathcal{S}_1 \not\subseteq \bigcup \mathcal{S}_2$ and show that $\mathcal{S}_1 \ntriangleleft \mathcal{S}_2$. By assumption, there exist a point $\mathbf{p} \in \mathbb{R}^n$ such that $\mathbf{p} \in \bigl(\bigcup \mathcal{S}_1\bigr) \setminus \bigl(\bigcup \mathcal{S}_2\bigr)$. As a consequence, there must exist a polyhedron $\mathcal{P}_1 \in \mathcal{S}_1$ such that $\mathbf{p} \in \mathcal{P}_1$. Consider now the polyhedron $\mathcal{P}_1' := \{\mathbf{p}\}$ and the corresponding singleton $\mathcal{S}_1' = \{\mathcal{P}_1'\}$. Note that $\mathcal{S}_1' \vdash_{\mathrm{P}} \mathcal{S}_1$ and $\mathcal{S}_1' \nvdash_{\mathrm{P}} \mathcal{S}_2$. Moreover, if $\mathcal{S}_1'' \in \wp^{\subseteq}_{\mathrm{fn}}(\mathbb{CP}_n)$ is such that $\biguplus \mathcal{S}_1'' = \biguplus \mathcal{S}_1'$, then we must have $\mathcal{S}_1'' = \mathcal{S}_1'$ so that $\mathcal{S}_1'' \nvdash_{\mathrm{P}} \mathcal{S}_2$. Hence, by Definition 17, $\mathcal{S}_1 \ntriangleleft \mathcal{S}_2$. By a symmetric argument, we can prove that $\bigcup \mathcal{S}_2 \not\subseteq \bigcup \mathcal{S}_1$ implies $\mathcal{S}_2 \ntriangleleft \mathcal{S}_1$. Thus, reasoning by contraposition, we obtain that $\mathcal{S}_1 \bowtie \mathcal{S}_2$ implies $\mathcal{S}_1 \equiv_{\gamma^{\mathsf{A}}_{\mathrm{P}}} \mathcal{S}_2$. □

To prove Proposition 6, it is convenient to consider *non-redundant merges*, where each of the elements in the original abstract collection participates to just one join operation.

Definition 20. Let '$\sim$' be a congruence relation on $\hat{D}_{\mathrm{P}}$. Let $S_1, S_2 \in \wp^{\vdash}_{\mathrm{fn}}(D)$, where $S_2 = \{d_1, \ldots, d_m\}$ and $\{S_{1i}\}_{i=1}^{m}$ is a partition of S_1 such that, for each $1 \leq i \leq m$, $\mathrm{merge}_{\sim}\bigl(S_{1i}, \{d_i\}\bigr)$ holds. Then we write $\mathrm{merge\_n}_{\sim}(S_1, S_2)$ and say that S_2 is a *non-redundant merge* of S_1.

Lemma 2. *Let '$\sim$' be a congruence relation on $\hat{D}_{\mathrm{P}}$ that refines the $\oplus$-congruence relation. Let $S_1 \neq S_2 \in \wp^{\vdash}_{\mathrm{fn}}(D)$ where* $\mathrm{merge}_{\sim}(S_1, S_2)$. *Then there exists $S_2' \in \wp^{\vdash}_{\mathrm{fn}}(D)$ such that* $\mathrm{merge\_n}_{\sim}(S_1, S_2')$ *and $\# S_2' < \# S_1$.*

Proof. As $S_1 \neq S_2$ and $\mathrm{merge}_{\sim}(S_1, S_2)$ holds, by Definition 18, $S_1 \Vdash_{\mathrm{P}} S_2$ so that there exists $d_2 \in S_2 \setminus S_1$. Therefore, also by Definition 18, there exists $S_1' \subseteq S_1$ (so that $d_2 \notin S_1'$) where $\{d_2\} \sim S_1'$. Since '$\sim$' refines the $\oplus$-congruence relation, we obtain $d_2 = \bigoplus S_1'$ so that $S_1' \neq \{d_2\}$ and $\# S_1' > 1$. Let $S_2' = (S_1 \setminus S_1') \oplus_{\mathrm{P}} \{d_2\}$. Then, by Definition 5, $\# S_2' < \# S_1$ and, by Definition 18, $\mathrm{merge}_{\sim}(S_1, S_2')$ holds. If $S_1 \setminus S_1' \neq \varnothing$, then $\{S_1 \setminus S_1', S_1'\}$ is a partition of S_1; otherwise, $\{S_1'\}$ is such a partition. In both cases, by Definition 20, $\mathrm{merge\_n}_{\sim}(S_1, S_2')$. □

Proof (of Proposition 6). We first prove, by induction on $\# S$, that there exists $S' \in \wp^{\vdash}_{\mathrm{fn}}(D)$ such that $\mathrm{merge}_{\sim}(S, S')$, S' is fully-merged and, if $S' \neq S$, then $\# S' < \# S$. As $\mathrm{merge}_{\sim}$ is reflexive, the result holds trivially if S is fully-merged. Note that, for the base cases when $\# S \leq 1$, S is fully-merged and the result holds. Suppose therefore that S is not fully-merged (so that $\# S > 1$). Then there exists $S'' \in \wp^{\vdash}_{\mathrm{fn}}(D) \setminus \{S\}$ such that $\mathrm{merge}_{\sim}(S, S'')$. By Lemma 2, we can assume that S'' is chosen so that $\mathrm{merge\_n}_{\sim}(S, S'')$ and $\# S'' < \# S$. Therefore we can apply the inductive hypothesis to S''; there exists $S' \in \wp^{\vdash}_{\mathrm{fn}}(D)$ which is fully-merged, $\mathrm{merge}_{\sim}(S'', S')$ and $\# S' \leq \# S''$. As $\mathrm{merge}_{\sim}$ is transitive, we obtain $\mathrm{merge}_{\sim}(S, S')$ and $\# S' < \# S$.

Therefore, the merger '$\upharpoonright_{\sim}$' can be defined, for each $S \in \wp^{\vdash}_{\mathrm{fn}}(D)$, as $\upharpoonright_{\sim} S = S$, when S is already fully-merged, and $\upharpoonright_{\sim} S = S'$ as defined above, otherwise. The proof for a pairwise-merger is similar. □