A Formal Framework to Measure the Incompleteness of Abstract Interpretations

No Author Given

No Institute Given

Abstract. In program analysis by abstract interpretation, completeness represents the best possible scenario where no imprecision is generated by the abstract interpreter. However, as for all approximation methods, the presence of false-alarms is unavoidable and therefore we need to deal somehow with imprecision, i.e., incompleteness. To this end, the partial completeness property has been recently formalized in order to tolerate a limited amount of imprecision when the abstract and concrete domains are complete lattices and form a Galois Connection. Partial completeness measures the amount of imprecision by using quasi-metrics compatible with the underlying abstract domain. In this paper we generalize the partial completeness property by considering also abstractions that admit a concretization map only and are not necessarily complete lattices. This leads us to formalize a weaker form of quasi-metric, called pre-metric, which can be defined on all domains equipped with a pre-order relation. Pre-metrics allow us to design distance functions that well fit in the considered abstract interpretation, according to the information and the corresponding level of approximation that we want to measure. We show how this newly defined notion of pre-metric allows us to derive other pre-metrics on other domains by exploiting the concretization and, when available, the abstraction maps.

Keywords: Abstract Interpretation · Partial Completeness · Completeness · Distances · Program Analysis

1 Introduction

The theory of Abstract Interpretation introduced by Cousot and Cousot [18,19,20], is a general theory for the approximation of formal program semantics based on a simple but striking idea that extracting properties of programs' execution means over-approximating their semantics. It is an invaluable framework that helps programmers design sound-by-construction program analysis tools as it makes possible to express mathematically the link between the output of a practical, approximated analysis, also called abstract semantics, and the original, uncomputable program semantics, also called concrete semantics.

In the standard abstract interpretation framework [18,19], the abstract interpretation of a program P consists of an abstract domain of properties of interest $\mathcal{A}$ ordered by a partial-order $\leq_{\mathcal{A}}$, a concretization map γ and an abstract interpreter $[\![\cdot]\!]_{\mathcal{A}}$, designed for the language used to specify P and on the abstract

domain $\mathcal{A}$. If $[\![\cdot]\!]$ is the concrete (collecting) program semantics, then *soundness* means that $[\![P]\!]\gamma(S^\sharp) \subseteq \gamma([\![P]\!]_{\mathcal{A}} S^\sharp)$ for all inputs $S^\sharp \in \mathcal{A}$. Furthermore, when $[\![P]\!]_{\mathcal{A}}$ satisfies $[\![P]\!]\gamma(S^\sharp) = \gamma([\![P]\!]_{\mathcal{A}} S^\sharp)$ for the input $S^\sharp$ then $[\![P]\!]_{\mathcal{A}}$ is said to be exact or *forward-complete* for $S^\sharp$. Forward-completeness in abstract interpretation intuitively encodes the greatest achievable precision for an abstract interpreter $[\![\cdot]\!]_{\mathcal{A}}$ applied on a program P with input $S^\sharp \in \mathcal{A}$, meaning that $[\![P]\!]_{\mathcal{A}} S^\sharp$ exactly matches the concrete result of the concrete counterpart $[\![P]\!]\gamma(S^\sharp)$. When the abstraction $\mathcal{A}$ also admits a Galois Connection (GC) through an abstraction map α, then $[\![P]\!]_{\mathcal{A}}$ is said to be *backward-complete* for the concrete input S when $\alpha([\![P]\!]S) = [\![P]\!]_{\mathcal{A}}\alpha(S)$ holds. The backward-completeness property encodes an optimal behavior of the abstract interpreter $[\![\cdot]\!]_{\mathcal{A}}$ with respect to the abstraction in $\mathcal{A}$ of the concrete behavior $[\![P]\!]S$. Backward and forward completeness are both highly desirable properties in program analysis for verifying safety properties of programs (also called trace properties) [19,30,26,36]. Unfortunately, it is well known that whenever a non-trivial abstract domain is used, the analysis will be necessarily incomplete, meaning that false alarms or spurious counterexamples will arise also for correct programs [26]. In fact, (forward/backward)-completeness in program analysis is extremely hard, if not even impossible, to achieve [30]. For this reason, instead of trying to reach completeness, we need to deal with incompleteness and therefore with imprecision [24].

To this end, the notion of *partial completeness* has been introduced in [10] in order to weaken the equality requirement of the backward-completeness property. Partial completeness allows a limited amount of (backward-)incompleteness and this amount is measured by quasi-metrics $\delta_{\mathcal{A}} : \mathcal{A} \times \mathcal{A} \to \mathbb{R}^{\infty}_{\geq 0} \cup \{\bot\}$ (where the symbol $\bot$ means undefined) compatible with the underlying abstract domain ordering, where $\mathcal{A}$ is assumed to be a complete lattice. More specifically, for a distance function $\delta_{\mathcal{A}}$ being a quasi-metric $\mathcal{A}$-compatible means satisfying for all $a_1, a_2, a_3 \in \mathcal{A}$: (i) the ordering $\leq_{\mathcal{A}}$: $a_1 \leq_{\mathcal{A}} a_2 \Leftrightarrow \delta_{\mathcal{A}}(a_1, a_2) \neq \bot$; (ii) identity of indiscernibles: $a_1 = a_2 \Leftrightarrow \delta(a_1, a_2) = 0$; and (iii) the weak triangle inequality, namely, the triangle inequality only along chains $a_1 \leq_{\mathcal{A}} a_2 \leq_{\mathcal{A}} a_3$. So for instance, consider the intervals abstract domain [17] $\mathsf{Int} \stackrel{def}{=} \{[a, b] \mid a, b \in \mathbb{Z}^*,\ a \leq b\} \cup \{\bot_{\mathsf{Int}}\}$, where $\mathbb{Z}^* \stackrel{def}{=} \mathbb{Z} \cup \{-\infty, +\infty\}$, endowed with the standard ordering $\leq_{\mathsf{Int}}$ induced by the interval containment. We can consider as quasi-metric Int-compatible the distance $\delta^{\mathfrak{w}}_{\mathsf{Int}}$ that counts how many more integer values has one interval with respect to another comparable interval. For instance, $\delta^{\mathfrak{w}}_{\mathsf{Int}}([0, 0], [0, 5]) = 5$, $\delta^{\mathfrak{w}}_{\mathsf{Int}}([0, +\infty], [-2, +\infty]) = 2$, while $\delta^{\mathfrak{w}}_{\mathsf{Int}}([0, 5], [0, 0]) = \bot$ as $[0, 5] \not\leq_{\mathsf{Int}} [0, 0]$. The analysis $[\![P]\!]_{\mathcal{A}}$ on a program P with input S is said to be ε-partial complete for an amount $\varepsilon \in \mathbb{R}^{\infty}_{\geq 0}$ whenever $\delta_{\mathcal{A}}(\alpha([\![P]\!]S), [\![P]\!]_{\mathcal{A}}\alpha(S)) \leq \varepsilon$ holds, namely the distance between the abstraction of the concrete execution and the result of the abstract interpreter is at maximum ε. In this setting, requiring 0-partial completeness corresponds to require backward-completeness.

Main Contribution. In this paper we generalize the partial completeness property in order to be able to weaken both the backward-completeness property in presence of a GC, *and* the forward-completeness property in case only the concretization function γ is available. In this last scenario, as we may need to define

distances on concrete domains, a weakening of the definition of quasi-metrics $\mathcal{A}$-compatible is necessary. This is because, in the original formalization [10], the definition of quasi-metric $\mathcal{A}$-compatible is specifically tailored for the structure of abstract domains and their relative partial-ordering: the axiom (i) forces the quasi-metric to return a value different from $\bot$ only if the two elements are comparable according to $\leq_{\mathcal{A}}$, namely $\delta_{\mathcal{A}}$ induces the partial-order $\leq_{\mathcal{A}}$. This is in fact not necessary: as our aim is to measure the incompleteness of an abstract interpreter with respect to the concrete execution, these two results are guaranteed to be comparable by soundness, therefore the distance may even return values on non-comparable elements as long as it is defined on all comparable ones. Moreover, the identity of indiscernibles axiom (ii) requires the quasi-metric to be precise enough to recognize equal elements since it constraints the distance to return zero whenever the two elements are equal. This is a too strong requirement especially in the scenario where only γ is available and we have to define a distance on the concrete domain where elements contain more information than what we are interested in for measuring the incompleteness. For instance, by considering the concrete domain $\wp(\mathbb{Z}^n)$ where n is the number of variables used in a program and elements in $S \in \wp(\mathbb{Z}^n)$ are program states, we might need a distance function $\delta_{\wp(\mathbb{Z}^n)}$ that measures the imprecision of certain variables only, say x and y. A possible estimate of this imprecision could be done by calculating the volume of their abstraction into the intervals abstract domain, namely, the area of the rectangle abstracting the values of x and y.

To this end, we formalize a new framework for defining a distance able to reason on incompleteness: in Section 3 we reason on the *weakest* axioms a distance function should meet. We just require a relaxed version of the identity of indiscernibles axiom (only the left-implication) and a condition on chains. As domains may not be complete lattices or partially-ordered, e.g. the convex polyhedra abstract domain [22], we only require one of the weakest form of ordering relation: a pre-order. The formed distance function will be called pre-metric $\preceq_D$-compatible where D is a pre-ordered set according to the pre-order $\preceq_D$. We will show many useful examples of pre-metrics compatible to generic pre-ordered sets, as well as well-known numerical abstract domains, that can be used in practice. In Section 4 we show how this newly defined notion of pre-metric $\preceq_D$-compatible allows us to derive other pre-metrics from one domain to another by exploiting the concretization γ or, when available, the abstraction map α. Finally, in Section 5 we define the new generalized notion of partial completeness using pre-metrics compatible with the underlying domain ordering and show that, when a certain condition on the precision of the pre-metric is met, then we can characterize the forward/backward-completeness as the 0-partial completeness. The proposed framework is general enough to be instantiated by most known metrics for abstract interpretation [23,33,38,11,10]. Since imprecision, i.e., incompleteness, is unavoidable in program analysis, our ambition is to help abstract interpretation designers in defining distances able to measure the imprecision they *want to track* regardless of the domain on which they want to define the distance, hence providing the appropriate tools to fully control the imprecision propagation.

2 Background

Orderings. Given two sets S and T, $\wp(S)$ denotes the powerset of S, $\varnothing$ is the empty set, $S \subseteq T$ denotes sets inclusion, $|S|$ denotes the cardinality where S is finite if $|S| < \omega$, countably infinite if $|S| = \omega$, countable if $|S| \leq \omega$. A binary relation $\sim$ over a set S is a subset of the Cartesian product $\sim \subseteq S \times S$. We will emphasize the set S on which a binary relation $\sim$ is defined by the subscript $\sim_S$ except for the straightforward equivalence relation $=$ unless it has a different definition. We denote with $\mathbb{Z}$ and $\mathbb{R}$ the sets of all, respectively, integer and real numbers. We will use subscripts in order to limit their range, while the superscript symbol ∞ denotes the inclusion of the infinite symbol. For example, $\mathbb{R}^{\infty}_{\geq 0}$ denotes the set of all non-negative real numbers with the infinity element.

A binary relation $\preceq_L \in \wp(L \times L)$ is a pre-order iff it is reflexive ($\forall l \in L.\ l \preceq_L l$) and transitive ($\forall l_1, l_2, l_3 \in L.\ l_1 \preceq_L l_2 \wedge l_2 \preceq_L l_3 \ \Rightarrow\ l_1 \preceq_L l_3$). A set L endowed with a pre-order relation $\preceq_L$ is called a pre-ordered set, and it is denoted by $(L, \preceq_L)$. Furthermore, if $\preceq_L$ is anti-symmetric ($\forall l_1, l_2 \in L.\ l_1 \preceq_L l_2 \wedge l_2 \preceq_L l_1 \ \Rightarrow\ l_1 = l_2$) then it is a partial-order and the couple $(L, \preceq_L)$ is called partially-ordered set. Clearly, every partially-ordered set is also a pre-ordered set. A subset $Y \subseteq L$ of a pre-ordered set $(L, \preceq_L)$ is a chain iff for all $y_1, y_2 \in Y$, $y_1 \preceq_L y_2$ or $y_2 \preceq_L y_1$.

Measures and Distances. A σ-algebra on a set X is a collection of subsets of X that includes X itself, is closed under complement and is closed under countable unions. The definition implies that it also includes the empty set $\varnothing$ and that it is closed under countable intersections. Consider a σ-algebra A over X. The tuple (X, A) is called a *measurable space*.

Definition 1 (Measure). *A function $\mu : A \to \mathbb{R}^{\infty}_{\geq 0}$ is called measure iff it satisfies the following properties:*

(1) *non-negativity:* $\forall S \in A.\ \mu(S) \geq 0$*;*
(2) *null empty set:* $\mu(\varnothing) = 0$*;*
(3) *countable additivity: if $S_i \in A$ is a countable sequence of disjoint sets, then $\mu(\bigcup_i S_i) = \sum_i \mu(S_i)$.*

The triple (X, A, μ) is called measure space. ■

A *metric* is a function that defines a distance between pairs of elements of a set S. Formally:

Definition 2 (Metric). *A metric on a non-empty set S is a map $\delta_S : S \times S \to \mathbb{R}_{\geq 0}$ that $\forall x, y, z \in S$ satisfies:*

(1) *identity of indiscernibles:* $x = y \Leftrightarrow \delta_S(x, y) = 0$*;*
(2) *symmetry:* $\delta_S(x, y) = \delta_S(y, x)$*;*
(3) *triangle inequality:* $\delta_S(x, z) \leq \delta_S(x, y) + \delta_S(y, z)$.

A set provided with a metric is called metric space. ■

A function $\delta_S : S \times S \to \mathbb{R}_{\geq 0}$ satisfying all axioms of Definition 2 except for symmetry is called *quasi-metric*, while if δ_S does not satisfy the $\Leftarrow$ direction of the

identity of indiscernibles axiom then it is called *pseudo-metric*. A pseudoquasi-metric relaxes both the indiscernibility axiom and the symmetry axiom of a metric. δ_S is said to be a *pre-metric* if it satisfies only the $\Leftarrow$ implication of the identity of indiscernibility axiom.

Abstract Interpretation. We consider here the standard abstract interpretation framework as defined in [20] and based on the correspondence between a domain of concrete or exact properties $\mathcal{C}$ and a domain of abstract or approximate properties $\mathcal{A}$. Concrete and abstract domains are assumed to be pre-ordered sets, respectively $(\mathcal{C}, \preceq_{\mathcal{C}})$ and $(\mathcal{A}, \preceq_{\mathcal{A}})$, and be related by a monotone *concretization* function $\gamma : \mathcal{A} \to \mathcal{C}$. Furthermore, when they enjoy a GC $(\mathcal{C}, \preceq_{\mathcal{C}}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\mathcal{A}, \preceq_{\mathcal{A}})$ through a monotone *abstraction* function $\alpha : \mathcal{C} \to \mathcal{A}$, then for all $a \in \mathcal{A}$ and $c \in \mathcal{C}$: $\alpha(c) \preceq_{\mathcal{A}} a \;\Leftrightarrow\; c \preceq_{\mathcal{C}} \gamma(a)$. A GC is a *Galois Insertion* (GI) when $\alpha \circ \gamma = id$, where $\circ$ denotes functions composition and id is the identity function.

Soundness, Completeness and Partial Completeness. Let $f_{\mathcal{C}} : \mathcal{C} \to \mathcal{C}$ be a concrete monotone operator (to keep notation simple we consider unary functions) and let $f_{\mathcal{A}} : \mathcal{A} \to \mathcal{A}$ be a corresponding monotone abstract operator defined on some abstraction $\mathcal{A}$. Then, $f_{\mathcal{A}}$ is a *sound* (or *correct*) approximation of $f_{\mathcal{C}}$ on $\mathcal{A}$ when for all $a \in \mathcal{A}$: $f_{\mathcal{C}}(\gamma(a)) \preceq_{\mathcal{C}} \gamma(f_{\mathcal{A}}(a))$ holds. If $f_{\mathcal{A}}$ is correct for $f_{\mathcal{C}}$ then least fixpoint correctness holds, that is, $\mathsf{lfp}(f_{\mathcal{C}}) \preceq_{\mathcal{C}} \gamma(\mathsf{lfp}(f_{\mathcal{A}}))$ holds. When dealing with GCs, between all abstract functions that approximate a concrete one $f_{\mathcal{C}}$ we can define the most precise one called *best correct approximation* (bca for short): $f^{\alpha}_{\mathcal{A}} \stackrel{def}{=} \alpha \circ f_{\mathcal{C}} \circ \gamma$. It turns out that any abstract function $f_{\mathcal{A}}$ is a correct approximation of $f_{\mathcal{C}}$ if and only if $f^{\alpha}_{\mathcal{A}} \preceq_{\mathcal{A}} f_{\mathcal{A}}$ [18].

When the concretization of the result of $f_{\mathcal{A}}(a)$ matches the concrete counterpart $f_{\mathcal{C}}(\gamma(a))$ then $f_{\mathcal{A}}$ is said to be *forward-complete*[1] at input $a \in \mathcal{A}$.

Definition 3 (Forward-completeness). *Let $f_{\mathcal{A}} : \mathcal{A} \to \mathcal{A}$ be a sound approximation of $f_{\mathcal{C}} : \mathcal{C} \to \mathcal{C}$. Given an input $a \in \mathcal{A}$, $f_{\mathcal{A}}$ is said to be forward-complete at the input a, when $f_{\mathcal{C}}(\gamma(a)) = \gamma(f_{\mathcal{A}}(a))$ holds.* ■

When $\mathcal{C}$ and $\mathcal{A}$ admit a GC, then we can also define the property of backward-completeness [19,30,5].

Definition 4 (Backward-completeness). *Let $(\mathcal{C}, \preceq_{\mathcal{C}}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\mathcal{A}, \preceq_{\mathcal{A}})$ and $f_{\mathcal{A}}$ be a sound approximation of $f_{\mathcal{C}}$. Given an input $c \in \mathcal{C}$, $f_{\mathcal{A}}$ is said to be backward-complete at the input c, when $\alpha(f_{\mathcal{C}}(c)) = f_{\mathcal{A}}(\alpha(c))$ holds.* ■

This definition of backward-completeness is also called local completeness [5] as it is required to hold on a specified input. As a remark, if $f_{\mathcal{A}}$ is backward-complete on an input $c^{\alpha} \in \mathcal{C}$ representable in $\mathcal{A}$, namely $c^{\alpha} = \gamma(\alpha(c^{\alpha}))$, then this implies that $f_{\mathcal{A}}(\alpha(c^{\alpha}))$ corresponds to the bca $f^{\alpha}_{\mathcal{A}}(\alpha(c^{\alpha}))$.

[1] The term "forward-complete" was introduced in [29] and it is also known in the literature as *exactness* [36]

A weakening of the backward-completeness property has been introduced in [10] where quasi-metrics comparable with the underlying abstract domain are considered for measuring the imprecision of $f_{\mathcal{A}}$ compared to $f_{\mathcal{C}}$.

Definition 5 (ε-Partial completeness). *Let $(\mathcal{C}, \preceq_{\mathcal{C}}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\mathcal{A}, \preceq_{\mathcal{A}})$, $f_{\mathcal{A}}$ be a sound approximation of $f_{\mathcal{C}}$, $\delta_{\mathcal{A}}$ be a quasi-metric $\mathcal{A}$-compatible, and $\varepsilon \in \mathbb{R}_{\geq 0}$. $f_{\mathcal{A}}$ is an ε-partial complete approximation of $f_{\mathcal{C}}$ on input $c \in \mathcal{C}$ when the following inequality holds: $\delta_{\mathcal{A}}(\alpha(f_{\mathcal{C}}(c)), f_{\mathcal{A}}(\alpha_{\mathcal{A}}(c))) \leq \varepsilon$.* ■

3 Distances on Orderings

The goal of this section is to set the minimum requirements that a distance function must meet so that it can be used to measure the incompleteness generated by a sound abstract function $f_{\mathcal{A}} : \mathcal{A} \to \mathcal{A}$, operating on a set of approximated properties $\mathcal{A}$, with respect to the concrete function $f_{\mathcal{C}} : \mathcal{C} \to \mathcal{C}$, operating on a set of properties $\mathcal{C}$ some of which may be undecidable. The target distance function could be defined either on the concrete domain $\mathcal{C}$ in order to calculate the distance between $f_{\mathcal{C}}(\gamma(a))$ and $\gamma(f_{\mathcal{A}}(a))$ for an input $a \in \mathcal{A}$, that is the forward-(in)completeness, or, e.g. when they enjoy a GC, directly on the abstract domain $\mathcal{A}$ for measuring the distance between $\alpha(f_{\mathcal{C}}(c))$ and $f_{\mathcal{A}}(\alpha(c))$ for $c \in \mathcal{C}$, that is the backward-(in)completeness (as, e.g., formalized in [10] through the use of quasi-metrics). In abstract interpretation both $\mathcal{C}$ and $\mathcal{A}$ are often based on a qualitative notion of precision in order to distinguish which property is more precise. More generally, given an unordered set D, a basic relation able to accomplish this task is a pre-order relation $\preceq_D \in \wp(D \times D)$ where $x \preceq_D y$ for $x, y \in D$ intuitively means that y *approximates* x. Therefore, we need to define a general notion of distance able to exploit *any* pre-ordered structure. Let us informally analyze each property that we may expect on a distance measuring the incompleteness of abstract interpretations.

When comparing two elements $x, y \in D$ in a pre-ordered set $(D, \preceq_D)$, the distance function $\delta_D(x, y)$ must return a non-negative real value for all $x, y \in D$. We also give δ_D the possibility to return the symbol ∞ meaning an infinite distance between two elements where $\varepsilon < \infty$ for all $\varepsilon \in \mathbb{R}_{\geq 0}$. Thus, δ_D is formally defined as:

$$\delta_D : D \times D \to \mathbb{R}^{\infty}_{\geq 0} \tag{0}$$

If we are calculating the distance between two identical elements, then we expect δ_D to output zero:

$$x = y \;\Rightarrow\; \delta_D(x, y) = 0 \tag{1}$$

However, we do not require the converse implication: we allow $x \neq y$ even if $\delta_D(x, y) = 0$. This gives us the freedom to say that, e.g., the distance between two distinct elements is zero because the distance itself is considering the information represented by x and y *up to some abstraction of interest*. That is, the distance itself can be considered as *another* layer of approximation between

the elements of D and, thus, it may output zero even if they are represented differently in D. For example, consider the poset $(\wp(\mathbb{Z}), \subseteq)$ corresponding to the powerset of integers together with the subset inclusion relation (i.e., a partial-order). Given two sets $X, Y \in \wp(\mathbb{Z})$ such that $X \subseteq Y$ (e.g., $X = \{2, 9, 19\}$ and $Y = \{2, 9, 15, 19\}$), we might be interested in a function $\delta_{\wp(\mathbb{Z})}$ that calculates the distance of an approximated representation of both X and Y, for instance, by taking their interval abstraction. In this case, it might happen that X and Y are mapped to the same interval (i.e., the interval $[2, 19]$ for the chosen X and Y) and therefore $\delta_{\wp(\mathbb{Z})}(X, Y) = 0$ even though $X \neq Y$. As another example, when considering the convex polyhedra domain $(\mathsf{Poly}, \preceq_{\mathsf{Poly}})$ [22] over $\mathbb{R}^n$ we might want that the distance between two polyhedra $p_1, p_2 \in \mathsf{Poly}$ is zero when they represent the same set of vectors in $\mathbb{R}^n$. That is, if $\gamma(p_1) = \gamma(p_2)$ then $\delta_{\mathsf{Poly}}(p_1, p_2) = 0$ even if p_1 and p_2 are represented by different inequalities in Poly, i.e., $p_1 \neq p_2$.

The requirements (0)-(1) define δ_D to be a generalization of a metric: by relaxing the identity of indiscernibles axiom and dropping the symmetry and triangle inequality axioms of metrics, we get a *pre-metric*[2]. Similarly to the relation between pre-orders and other stronger orderings (e.g., partial-orders and equivalence relations), pre-metrics are more general than pseudoquasi-metrics, quasi-metrics and metrics (see Section 2): a pre-metric satisfying the triangle inequality axiom is a pseudoquasi-metric, furthermore if it also satisfies the identity of indiscernibles then it is a quasi-metric, while a symmetric quasi-metric is a metric. Pre-metrics can be considered as one of the weakest forms of distance functions from which we can build on top of pre-ordered sets.

Until now the definition of pre-metric does not consider the pre-order between elements of D. Recall that we are interested in computing a distance between the result of a concrete operator $f_{\mathcal{C}}$ working on $(\mathcal{C}, \preceq_{\mathcal{C}})$ and a *sound* abstract operator $f_{\mathcal{A}}$ working on $(\mathcal{A}, \preceq_{\mathcal{A}})$. Therefore, we already know that for any $a \in \mathcal{A}$ the two results $f_{\mathcal{C}}(\gamma(a))$ and $\gamma(f_{\mathcal{A}}(a))$ are comparable according to $\preceq_{\mathcal{C}}$, namely $f_{\mathcal{C}}(\gamma(a)) \preceq_{\mathcal{C}} \gamma(f_{\mathcal{A}}(a))$ thanks to the soundness assumption of $f_{\mathcal{A}}$. This means that our definition of distance should have a meaning when used to calculate distances between elements being part of the same chain, i.e., comparable according to $\preceq_D$, while we do not care about the result of $\delta_D(x, y)$ when $x \not\preceq_D y$. That said, suppose $x, y, z \in D$ are related by $x \preceq_D y \preceq_D z$, i.e., z is an approximation of y and y approximates x. If we ascend the chain from x to y, then we would expect that the remaining distance from y to z to be less than or equal the entire distance from x to z. Similarly, if we descend the chain from z to y then we would expect the remaining distance from x and y to be less than or equal the whole distance from x to z. Formally:

$$x \preceq_D y \preceq_D z \;\Rightarrow\; \delta_D(x, y) \leq \delta_D(x, z) \wedge \delta_D(y, z) \leq \delta_D(x, z) \tag{2}$$

[2] This is not a standard term in the literature: sometimes it is used to refer to other generalizations of metrics such as pseudosemi-metrics [7] or pseudo-metrics [31]; it sometimes appears as pra-metric [3]. This definition is taken from Wikipedia [1].

This axiom gives us the possibility to reason on distance results between elements on the same chain. For example, suppose that the concrete and abstract domains are related by a GC $(\mathcal{C}, \preceq_{\mathcal{C}}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\mathcal{A}, \preceq_{\mathcal{A}})$ and that we have defined a distance $\delta_{\mathcal{A}}$ on the elements of $\mathcal{A}$. Given an input $c \in \mathcal{C}$, if the abstraction of the result of the concrete operator is $\alpha(f_{\mathcal{C}}(c)) = x$, and the result of our chosen sound abstract operator is $f_{\mathcal{A}}(\alpha(c)) = z$, then the result of the bca of $f_{\mathcal{C}}$ on $\mathcal{A}$ is in the middle between them, namely, if $f^{\alpha}_{\mathcal{A}}(\alpha(c)) = y$ then $x \preceq_{\mathcal{A}} y \preceq_{\mathcal{A}} z$. In this case, we would expect that the distance from the concrete result x to the result of the best possible approximation of $f_{\mathcal{C}}$, i.e., $\delta_{\mathcal{A}}(x, y)$ to be less than or equal to the distance between the concrete and the chosen abstract operator $f_{\mathcal{A}}$, namely, $\delta_{\mathcal{A}}(x, z)$, and the same for $\delta_{\mathcal{A}}(y, z)$. Note that the triangle inequality axiom required by metrics and some of their weakening, like pseudo-metrics and quasi-metrics, does not imply axiom (2), and (2) does not imply the triangle inequality. For example, if $D = \{x, y, z\}$ with $x \preceq_D y \preceq_D z$ and $\delta_D(x, y) = 2$, $\delta_D(y, z) = 1$, $\delta_D(x, z) = 1$, then $\delta_D(x, z) = 1 < 3 = \delta_D(x, y) + \delta_D(y, z)$ but $\delta_D(x, y) = 2 > \delta_D(x, z) = 1$. Instead, if $\delta_D(x, y) = 1$, $\delta_D(y, z) = 1$, $\delta_D(x, z) = 3$ then (2) holds while $\delta_D(x, z) = 3 > 2 = \delta_D(x, y) + \delta_D(y, z)$. In fact, we do not require the triangle inequality axiom (neither its weaker form on chains as formalized, e.g., in [23,33,10]): as we are focusing on incompleteness results and, therefore, elements on chains according to the ordering $\preceq_D$, the distance $\delta_{\mathcal{A}}(x, z)$ could be greater or lower than the sum between $\delta_{\mathcal{A}}(x, y)$ and $\delta_{\mathcal{A}}(y, z)$ as long as it respects (2).

We have now all the ingredients to formalize the distance matching our purposes: it must be a pre-metric (axioms (0)-(1)) compatible with the underlying pre-order (axiom (2)). Functions that meet these requirements over a pre-ordered set $(D, \preceq_D)$ are called *pre-metrics $\preceq_D$-compatible*.

Definition 6 (Pre-metric $\preceq_D$-compatible). *Let $(D, \preceq_D)$ be a pre-ordered set. The function $\delta_D : D \times D \to \mathbb{R}^{\infty}_{\geq 0}$ is a pre-metric $\preceq_D$-compatible if and only if the following axioms are satisfied for all $x, y, z \in D$:*

(1) $x = y \;\Rightarrow\; \delta_D(x, y) = 0$;

(2) $x \preceq_D y \preceq_D z \;\Rightarrow\; \delta_D(x, y) \leq \delta_D(x, z) \wedge \delta_D(y, z) \leq \delta_D(x, z)$. ■

Pre-ordered sets equipped with a compatible pre-metric are called *pre-metric $\preceq_D$-compatible spaces*.

Definition 7 (Pre-metric $\preceq_D$-compatible space). *Given a pre-ordered set $(D, \preceq_D)$ and a pre-metric $\preceq_D$-compatible δ_D, the triple $(D, \preceq_D, \delta_D)$ is a pre-metric $\preceq_D$-compatible space. We use $Pre((D, \preceq_D))$ to refer to the the set of all pre-metric $\preceq_D$-compatible spaces: $(D, \preceq_D, \delta_D) \in Pre((D, \preceq_D))$.* ■

The following are a list of pre-metrics compatible with a generic pre-ordered set $(D, \preceq_D)$ or shaped for specific domains.

Example 1 (Zero-distance). One of the most trivial pre-metric $\preceq_D$-compatible definable on any pre-ordered set is the distance that always returns the value

zero for all $x, y \in D$:

$$\delta^0_D(x, y) \stackrel{def}{=} 0$$

Although it respects all axioms from Definition 6, it does not give any information on the distance between elements in D since it treats them as they are all close. ⧫

Example 2 (Ordering-distance). The following distance

$$\delta^{\preceq_D}_D(x, y) \stackrel{def}{=} \begin{cases} 0 & \text{if } x = y, \\ 1 & \text{if } x \neq y \wedge x \preceq_D y, \\ \infty & \text{otherwise} \end{cases}$$

is clearly a pre-metric $\preceq_D$-compatible. In fact, it extends the pre-order relation $\preceq_D$ with the function $\delta^{\preceq_D}_D$ having three output values: 0 for equal elements, 1 for not equal but comparable elements and ∞ for non-comparable elements. ⧫

Example 3 (Measure-distance). Let (Z, D, μ) be a measure space, i.e., D be a domain that forms a σ-algebra over a set Z and $\mu : D \to \mathbb{R}^{\infty}_{\geq 0}$ be a measure function. We define the function δ^{μ}_D for every $X, Y \in D$ as follows:

$$\delta^{\mu}_D(X, Y) \stackrel{def}{=} Av(\mu(Y) - \mu_D(X))$$

where Av is the absolute value function. Note that, because D is composed by measurable properties, δ^{μ}_D can exploit the measure function μ in order to quantify the distance between elements of D. However, depending on how $\preceq_D$ is defined, it still may not be a pre-metric $\preceq_D$-compatible as axiom (2) may be violated. Let us show two examples where δ^{μ}_D is compatible with $\preceq_D$.

Consider the measure space $(D, \wp(D), \mu^c)$, where $(\wp(D), \subseteq)$ and μ^c is the *counting measure*, namely, for all $X \in \wp(D)$, $\mu^c(X) \stackrel{def}{=} |X|$ if $|X|$ is finite, ∞ otherwise. Intuitively, $\delta^{\mu^c}_{\wp(D)}(X, Y)$ counts the elements in X and Y and returns the absolute value of their difference. Note that: axioms (0)-(1) are satisfied since $\delta^{\mu^c}_{\wp(D)}(X, Y)$ is either non-negative or ∞, and if $X = Y$ then they have the same number of elements which implies[3] $\delta^{\mu^c}_{\wp(D)}(X, Y) = 0$. Furthermore, axiom (2) holds as $X \subseteq Y \subseteq Z$ implies that Z has more elements than Y and Y has more elements than X, thus ascending (resp. descending) a chain implies that the distance will increase (resp. decrease). The function $\delta^{\mu^c}_{\wp(D)}$ fulfills all axioms (0)-(2) and, therefore, it is a pre-metric $\subseteq$-compatible. Dually, the same reasoning holds with $\wp(D)$ being partially-ordered by $\supseteq$. This is one of the most common distance used for evaluating the outcome of a program analysis: you simply count the elements generated by the abstract analysis and the elements generated by the concrete execution and then the absolute value of the difference tells you the quality of the analysis result. The bigger this difference is, the worse the result will be. ⧫

[3] We assume the following results when the ∞ symbol is involved: $Av(k - \infty) = Av(\infty - k) = \infty$ with $k \in \mathbb{R}$, while $\infty - \infty = 0$.

Example 4 (Volume-distance). Let us consider the pre-ordered domain of convex polyhedra $(\mathsf{Poly}, \preceq_{\mathsf{Poly}})$. We define the pre-metric

$$\delta^{\mathit{Vol}}_{\mathsf{Poly}}(p_1, p_2) \stackrel{\mathit{def}}{=} \mathit{Av}(\mathit{Vol}(p_1) - \mathit{Vol}(p_2))$$

calculating the absolute value of the difference between the volume of two polyhedra $p_1, p_2 \in \mathsf{Poly}$. The volume function $\mathit{Vol} : \mathsf{Poly} \to \mathbb{R}^{\infty}_{\geq 0}$ could be an (over-)approximation of the exact volume computation (see, e.g., [13,32]). This means that *Vol* may not be a measure according to Definition 1 as the countable-additivity axiom may be violated. However, $\delta^{\mathit{Vol}}_{\mathsf{Poly}}$ satisfies the two axioms of Definition 6 and therefore it is $\preceq_{\mathsf{Poly}}$-compatible. ⧫

Example 5 (Trace-Length distance). Let Σ be a set of program states and let $\Sigma^{+\infty} \stackrel{\mathit{def}}{=} \Sigma^{+} \cup \Sigma^{\infty}$ be the set of all non-empty finite (Σ^{+}) and infinite (Σ^{∞}) sequences of program states. We consider the domain of sets of program traces ordered by set inclusion, i.e., $(\wp(\Sigma^{+\infty}), \subseteq)$, and define the following function function $\mathit{Len}\colon \wp(\Sigma^{+\infty}) \to \mathbb{R}^{\infty}_{\geq 0}$:

$$\mathit{Len}(T) \stackrel{\mathit{def}}{=} \begin{cases} 0 & \text{if } T = \varnothing, \\ \max\{|\sigma| \mid \sigma \in T\} & \text{if } T \cap \Sigma^{\infty} = \varnothing, \\ \infty & \text{otherwise} \end{cases}$$

where $|\sigma|$ applied on a trace denotes its length. *Len* computes the length of the longest program trace in a set of traces T. The following pre-metric

$$\delta^{\mathit{Len}}_{\wp(\Sigma^{+\infty})}(T_1, T_2) \stackrel{\mathit{def}}{=} \mathit{Av}(\mathit{Len}(T_1) - \mathit{Len}(T_2))$$

looking at the absolute value of the difference between the lengths of the longest traces in two sets $T_1, T_2 \in \wp(\Sigma^{+\infty})$ is a pre-metric $\subseteq$-compatible. Note that $\mathit{Len}(T)$ does not form a measure as the countable-additivity axiom does not hold. ⧫

Example 6 (Weighted path-length distance). We consider the weighted path-length distance $\delta^{\mathfrak{w}}_{D}$ defined in [10] for posets. We propose a slightly modified version able to work with any pre-ordered structures $(D, \preceq_D)$. Intuitively, $\delta^{\mathfrak{w}}_{D}$ considers a pre-ordered set as a directed weighted graph where the set of edges $E_D \subseteq D \times D$ is defined as $E_D \stackrel{\mathit{def}}{=} \{(x, y) \mid x \prec_D y\}$, and $\mathfrak{w} : E_D \to \mathbb{R}_{\geq 0}$ is the weight function which assigns a non-negative real value to each edge. The relation $x \prec_D y$ is true whenever $x \prec_D y$ and there is no element $z \in D$ such that $x \prec_D z \prec_D y$. Clearly, if $\preceq_D$ is a partial-order then the graph is acyclic. Given $x, y \in D$ such that $x \neq y$, let $\mathfrak{C}^{y}_{x}$ denotes the set of all possible chains $\mathbf{c} \subseteq E_D$ such that if $(z, u) \in \mathbf{c}$ then $x \preceq_D z \prec_D u \preceq_D y$. It is clear that if $x \not\preceq_D y$ then $\mathfrak{C}^{y}_{x} = \varnothing$. The weighted path-length distance $\delta^{\mathfrak{w}}_{D} : D \times D \to \mathbb{R}^{\mathfrak{w}}_{\geq 0}$ is defined as follows:

$$\delta^{\mathfrak{w}}_{D}(x, y) \stackrel{\mathit{def}}{=} \begin{cases} 0 & \text{if } x = y, \\ \infty & \text{if } \forall \mathbf{c} \in \mathfrak{C}^{y}_{x}.\ |\mathbf{c}| = \omega, \\ \min\left\{ \sum_{e \in \mathbf{c}} \mathfrak{w}(e) \;\middle|\; \begin{array}{l} \mathbf{c} \in \mathfrak{C}^{y}_{x} \\ |\mathbf{c}| < \omega \end{array} \right\} & \text{if } \exists \mathbf{c} \in \mathfrak{C}^{y}_{x}.\ |\mathbf{c}| < \omega. \end{cases}$$

Intuitively, when $\delta_D^{\mathfrak{w}}$ is used to calculate the distance between x and y such that $x \prec_D y$ then it outputs the minimum weighted path w.r.t. $\mathfrak{w}$ between x and y, while if $x \not\preceq_D y$ then it outputs ∞. Note that $\delta_D^{\mathfrak{w}}$ is a pre-metric that does not satisfy symmetry, while it satisfies the triangle inequality axiom only on chains. However, it may not be compatible with the underlying ordering $\preceq_D$ as axiom (2) may turn false. For instance, consider the $\mathsf{Sign} \stackrel{def}{=} \{\mathbb{Z}, -, 0, +, \varnothing\}$ domain for sign analysis of integer variables [17]. Sign is ordered by the following partial-order: $\varnothing \preceq_{\mathsf{Sign}} 0 \preceq_{\mathsf{Sign}} - \preceq_{\mathsf{Sign}} \mathbb{Z}$ and $\varnothing \preceq_{\mathsf{Sign}} 0 \preceq_{\mathsf{Sign}} + \preceq_{\mathsf{Sign}} \mathbb{Z}$. Suppose the weight function $\mathfrak{w}$ assigns $\mathfrak{w}((0,-)) = 5$ while for the others couple $(a,b) \in E_{\mathsf{Sign}}$, $\mathfrak{w}((a,b)) = 1$. Then, the weighted path-length $\delta_{\mathsf{Sign}}^{\mathfrak{w}}$ is not a pre-metric $\preceq_{\mathsf{Sign}}$-compatible as $\delta_{\mathsf{Sign}}^{\mathfrak{w}}(0,-) = 5 > 2 = \delta_{\mathsf{Sign}}^{\mathfrak{w}}(0,\mathbb{Z})$ thus violating (2). On the other hand, if we set $\forall (a,b) \in E_{\mathsf{Sign}}$, $\mathfrak{w}((a,b)) = 1$ then we get a pre-metric $\preceq_{\mathsf{Sign}}$-compatible.

As a final case of application of $\delta_D^{\mathfrak{w}}$, consider the domain of integer intervals Int also known as the box domain. Given any two intervals $i_1, i_2 \in \mathsf{Int}$ such that $(i_1, i_2) \in E_{\mathsf{Int}}$, if we define $\mathfrak{w}((i_1, i_2)) = 1$, then $\delta_{\mathsf{Int}}^{\mathfrak{w}}$ is a pre-metric $\preceq_{\mathsf{Int}}$-compatible. Intuitively, $\delta_{\mathsf{Int}}^{\mathfrak{w}}(i_1, i_2)$ for $i_1 \preceq_{\mathsf{Int}} i_2$ counts how many more elements one interval has w.r.t. another one: if $\delta_{\mathsf{Int}}^{\mathfrak{w}}(i_1, i_2) = k$ for some $k \in \mathbb{N}$, then the interval i_2 contains exactly k more elements than i_1. For instance, $\delta_{\mathsf{Int}}^{\mathfrak{w}}([0,0],[-1,2]) = 3$ as the interval $[-1,2]$ has 3 more elements than the singleton $[0,0]$, namely: $-1, 1, 2$; $\delta_{\mathsf{Int}}^{\mathfrak{w}}([0,10],[0,+\infty]) = \infty$ as $[0,+\infty]$ has an infinite number of more elements than $[0,10]$, while $\delta_{\mathsf{Int}}^{\mathfrak{w}}([0,+\infty],[-5,+\infty]) = 5$. ♦

When a pre-metric $\preceq_D$-compatible is precise enough to be able to assign zero only when two comparable elements are identical, namely, when it satisfies the identity of indiscernibles axiom on chains, it will be called *strong*.

Definition 8 (Strong pre-metric $\preceq_D$-compatible). *Consider the pre-ordered space $(D, \preceq_D, \delta_D)$. The pre-metric $\preceq_D$-compatible δ_D is said to be strong if and only if the following implication holds for every $x, y \in D$:*

$$x \preceq_D y \;\Rightarrow\; (\delta_D(x,y) = 0 \;\Rightarrow\; x = y) \qquad \blacksquare$$

For instance, the ordering-distance $\delta_D^{\preceq_D}$ of Example 2 and the weighted path-length $\delta_{\mathsf{Int}}^{\mathfrak{w}}$ defined on intervals in Example 6 are strong, whereas the zero-distance δ_D^0, the counting measure-distance $\delta_{\wp(D)}^{\mu^c}$ (except for some D, e.g., $\delta_{\wp(\mathbb{Z})}^{\mu^c}$ is strong), the volume-distance $\delta_{\mathsf{Poly}}^{Vol}$ on polyhedra and the trace-length distance $\delta_{\wp(\Sigma^{+\infty})}^{Len}$ defined, respectively, in Examples 1, 3, 4 and 5, are not. We will see in Section 5 that strong pre-metrics $\preceq_D$-compatible play an important rule when measuring the incompleteness of abstract interpretations.

As a last note, it is worth noting that Definition 6 is general enough to be instantiated with other definitions of metrics specifically shaped in the context of abstract interpretation. For instance, if a pre-metric $\preceq_D$-compatible δ_D is also symmetric and it satisfies the weak triangle inequality then it is a pseudo-metric $\preceq_D$-compatible according to [33], whereas if δ_D both induces the underlying order relation, it is strong and it satisfies the weak triangle inequality then it is a quasi-metric $\preceq_D$-compatible [10,23].

4 Deriving Pre-metrics from Domains

Concrete $\mathcal{C}$ and abstract $\mathcal{A}$ domains of properties in abstract interpretation are often related by a monotonic concretization function $\gamma : \mathcal{A} \to \mathcal{C}$ and sometimes additionally by a monotonic abstraction function $\alpha : \mathcal{C} \to \mathcal{A}$ that associates an abstract element to a concrete one such that $(\mathcal{C}, \preceq_{\mathcal{C}}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\mathcal{A}, \preceq_{\mathcal{A}})$ forms a GC. By exploiting these structures we can *derive* pre-metrics from one domain to another.

Given a pre-metric compatible with the concrete domain $(\mathcal{C}, \preceq_{\mathcal{C}})$, we can exploit the concretization function γ to derive a pre-metric compatible with the underlying abstract domain ordering. Here a GC between $\mathcal{C}$ and $\mathcal{A}$ is not necessary.

Definition 9 (Induced distance from the concrete domain). *Consider $(\mathcal{C}, \preceq_{\mathcal{C}}, \delta_{\mathcal{C}}) \in Pre((\mathcal{C}, \preceq_{\mathcal{C}}))$. For all $a_1, a_2 \in \mathcal{A}$, we define:*

$$\overrightarrow{\delta}_{\mathcal{A}}(a_1, a_2) \stackrel{def}{=} \delta_{\mathcal{C}}(\gamma(a_1), \gamma(a_2))$$

as the pre-metric induced on $\mathcal{A}$ from the concrete pre-metric $\preceq_{\mathcal{C}}$-compatible space $(\mathcal{C}, \preceq_{\mathcal{C}}, \delta_{\mathcal{C}})$. ■

Proposition 1. *The following statements hold:*

(i) $(\mathcal{A}, \preceq_{\mathcal{A}}, \overrightarrow{\delta}_{\mathcal{A}})$ is a pre-metric $\preceq_{\mathcal{A}}$-compatible space;

(ii) if $\delta_{\mathcal{C}}$ is strong and γ is injective then $\overrightarrow{\delta}_{\mathcal{A}}$ is strong. □

Furthermore, when the concrete and abstract domains admit a GC through an abstraction function $\alpha : \mathcal{C} \to \mathcal{A}$, we can define a pre-metric $\preceq_{\mathcal{A}}$-compatible directly on the abstract properties, yet leading to the pre-metric $\preceq_{\mathcal{A}}$-compatible space $(\mathcal{A}, \preceq_{\mathcal{A}}, \delta_{\mathcal{A}})$, and then derive the pre-metric $\preceq_{\mathcal{C}}$- compatible on the concrete properties. This distance will be called the induced distance from the abstract pre-metric $\preceq_{\mathcal{A}}$-compatible space.

Definition 10 (Induced distance from the abstract domain). *Let the concrete and abstract domains be correlated by a GC $(\mathcal{C}, \preceq_{\mathcal{C}}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\mathcal{A}, \preceq_{\mathcal{A}})$. Moreover, let $(\mathcal{A}, \preceq_{\mathcal{A}}, \delta_{\mathcal{A}}) \in Pre((\mathcal{A}, \preceq_{\mathcal{A}}))$ be a pre-metric $\preceq_{\mathcal{A}}$-compatible space. For all $c_1, c_2 \in \mathcal{C}$, we define:*

$$\overleftarrow{\delta}_{\mathcal{C}}(c_1, c_2) \stackrel{def}{=} \delta_{\mathcal{A}}(\alpha(c_1), \alpha(c_2))$$

as the pre-metric induced on $\mathcal{C}$ from the abstract pre-metric $\preceq_{\mathcal{A}}$-compatible space $(\mathcal{A}, \preceq_{\mathcal{A}}, \delta_{\mathcal{A}})$. ■

Proposition 2. *$(\mathcal{C}, \preceq_{\mathcal{C}}, \overleftarrow{\delta}_{\mathcal{C}})$ is a pre-metric $\preceq_{\mathcal{C}}$-compatible space.* □

The derived pre-metric on the concrete properties is compatible with $\preceq_{\mathcal{C}}$ as it measures the distance between two concrete elements by throwing away non-relevant information according to the abstraction α.

Note how Definition 9 and Definition 10 define a way to build pre-metrics on domains correlated by a concretization function and/or an abstraction function. This means that the distance itself δ_D defined on a pre-ordered domain D, can view properties of D on different levels of precision: δ_D can exploit a more approximated pre-metric $\delta_{\mathcal{A}}$ defined on an abstraction of properties $\mathcal{A}$ of D, e.g. we can use $\overleftarrow{\delta}_D$ when $(D, \preceq_D) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\mathcal{A}, \preceq_{\mathcal{A}})$. Alternatively, δ_D can exploit a more precise distance $\delta_{\mathcal{C}}$, for instance when $\delta_{\mathcal{C}}$ is defined on a more precise domain $\mathcal{C}$ related with D through the concretization $\gamma : D \to \mathcal{C}$, then we can use $\overrightarrow{\delta}_D$. We can also combine distances in a way similar to combining abstractions.

Example 7. Consider the volume-distance $\delta^{\mathit{Vol}}_{\mathsf{Poly}}$ defined on convex polyhedra in Example 4. We can systematically derive other volume-distances on domains which can be represented by Poly such as intervals Int, Zones Zone [34] and Octagons Oct [35]. For instance, given $\gamma_{\mathsf{Oct}} : \mathsf{Oct} \to \mathsf{Poly}$, $\gamma_{\mathsf{Zone}} : \mathsf{Zone} \to \mathsf{Poly}$, $\gamma_{\mathsf{Int}} : \mathsf{Int} \to \mathsf{Poly}$, for all $o_1, o_2 \in \mathsf{Oct}$, $z_1, z_2 \in \mathsf{Zone}$, $i_1, i_2 \in \mathsf{Int}$ we get

$$\overrightarrow{\delta}^{\mathit{Vol}}_{\mathsf{Oct}}(o_1, o_2) = \delta^{\mathit{Vol}}_{\mathsf{Poly}}(\gamma_{\mathsf{Oct}}(o_1), \gamma_{\mathsf{Oct}}(o_2))$$
$$\overrightarrow{\delta}^{\mathit{Vol}}_{\mathsf{Zone}}(z_1, z_2) = \delta^{\mathit{Vol}}_{\mathsf{Poly}}(\gamma_{\mathsf{Zone}}(z_1), \gamma_{\mathsf{Zone}}(z_2))$$
$$\overrightarrow{\delta}^{\mathit{Vol}}_{\mathsf{Int}}(i_1, i_2) = \delta^{\mathit{Vol}}_{\mathsf{Poly}}(\gamma_{\mathsf{Int}}(i_1), \gamma_{\mathsf{Int}}(i_2)) \qquad \blacklozenge$$

Depending on a number of factors such as the imprecision we want to track, the quantity of information represented by a domain, and/or the computational complexity needed to implement δ_D, we may switch from one domain to another. This procedure is also common in program analysis by abstract interpretation where it can be useful to convert between one abstract domain and another, for instance to switch abstract domains dynamically during the analysis or benefit from abstract operators available in other more abstract domains (see, e.g., [16,36]).

Example 8. Let us consider the concrete set $(\wp(\mathbb{Z}^n), \subseteq)$ and the abstract pre-metric $\preceq_{\mathsf{Int}}$-compatible space $(\mathsf{Int}, \preceq_{\mathsf{Int}}, \delta^{\mathfrak{w}}_{\mathsf{Int}})$ of intervals together with the weighted path-length defined in Example 6. We can derive the pre-metric

$$\overleftarrow{\delta}_{\wp(\mathbb{Z}^n)}(S_1, S_2) = \delta^{\mathfrak{w}}_{\mathsf{Int}}(\alpha_i(S_1), \alpha_i(S_2))$$

where for all $S_1, S_2 \in \wp(\mathbb{Z}^n)$, $\alpha_i : \wp(\mathbb{Z}^n) \to \mathsf{Int}$ calculates the interval of the i-th component only, with $1 \leq i \leq n$, of set of vectors S_1 and S_2. For instance, if $n = 3$ and $S_1 = \{\langle 1, 9, 9\rangle, \langle 1, 0, 10\rangle\}$, $S_2 = \{\langle 1, 5, 0\rangle, \langle -1, 0, 10\rangle, \langle 5, 0, 0\rangle\}$ then $\alpha_1(S_1) = [1, 1]$, $\alpha_1(S_2) = [-1, 5]$, and their distance is $\overleftarrow{\delta}_{\wp(\mathbb{Z}^n)}(S_1, S_2) = \delta^{\mathfrak{w}}_{\mathsf{Int}}([1, 1], [-1, 5]) = 6$. This can be useful, e.g., when $\sigma \in S$ represents a program state and the i-th component of σ corresponds to the value of a program

variable, thus, $\overleftarrow{\delta}_{\wp(\mathbb{Z}^n)}(S_1, S_2)$ is interested in calculating the imprecision of that variable only. ⧫

5 Generalized Partial Completeness Property

In this section we generalize the notion of partial completeness defined in [10] through the use of pre-metrics defined in Section 3. Unlike in [10] where the concrete $\mathcal{C}$ and abstract $\mathcal{A}$ domains of properties are required to be complete lattices and related by a GC, we ask $\mathcal{C}$ and $\mathcal{A}$ to have fewer structures: they must be pre-ordered sets and be correlated by a monotone concretization function $\gamma : \mathcal{A} \to \mathcal{C}$. In particular, given a pre-metric $\delta_{\mathcal{C}} : \mathcal{C} \times \mathcal{C} \to \mathbb{R}^{\infty}_{\geq 0}$ encoding what imprecision we want to measure on the concrete pre-ordered set $(\mathcal{C}, \preceq_{\mathcal{C}})$, we require $(\mathcal{C}, \preceq_{\mathcal{C}}, \delta_{\mathcal{C}})$ to be a pre-metric $\preceq_{\mathcal{C}}$-compatible space. The new generalized notion of partial completeness is defined as follows.

Definition 11 (ε-Partial completeness). *Let $(\mathcal{C}, \preceq_{\mathcal{C}}, \delta_{\mathcal{C}}) \in Pre((\mathcal{C}, \preceq_{\mathcal{C}}))$ and $f_{\mathcal{A}} : \mathcal{A} \to \mathcal{A}$ be a sound approximation of $f_{\mathcal{C}} : \mathcal{C} \to \mathcal{C}$. Given $\varepsilon \in \mathbb{R}^{\infty}_{\geq 0}$, we say that $f_{\mathcal{A}}$ is an ε-partial complete approximation of $f_{\mathcal{C}}$ on input $a \in \mathcal{A}$ if and only if the following predicate holds:*

$$\delta_{\mathcal{C}}(f_{\mathcal{C}}(\gamma(a)), \gamma(f_{\mathcal{A}}(a))) \leq \varepsilon$$ ■

The value of the distance δ_D between the result of the concrete operator $f_{\mathcal{C}}(\gamma(a))$ and the concretization of the abstract operator $\gamma(f_{\mathcal{A}}(a)))$ can be interpreted as the measure of the approximation introduced by $f_{\mathcal{A}}$ with respect to $f_{\mathcal{C}}$. Therefore, this distance encodes a quantitative level of imprecision introduced by $f_{\mathcal{A}}$, more precisely, *the imprecision that we want to measure* according to how we have defined the pre-metric $\preceq_{\mathcal{C}}$-compatible $\delta_{\mathcal{C}}$.

Proposition 3. *Let $(\mathcal{C}, \preceq_{\mathcal{C}}, \delta_{\mathcal{C}}) \in Pre((\mathcal{C}, \preceq_{\mathcal{C}}))$ and $f_{\mathcal{A}} : \mathcal{A} \to \mathcal{A}$ be a correct approximation of $f_{\mathcal{C}} : \mathcal{C} \to \mathcal{C}$. The following hold for every $a \in \mathcal{A}$:*

(i) $f_{\mathcal{A}}$ ε-partial complete $\Rightarrow$ $\forall \xi \geq \varepsilon$. $f_{\mathcal{A}}$ ξ-partial complete;

(ii) $f_{\mathcal{A}}$ ∞-partial complete. □

If $f_{\mathcal{A}}$ is ε-partial complete then admitting a larger imprecision ξ according to $\delta_{\mathcal{C}}$ results in the property ξ-partial completeness which is always satisfied by $f_{\mathcal{A}}$. This implies that, if we define the class of all ε-partial complete (sound) abstract operators with respect to $f_{\mathcal{C}}$ and input $a \in \mathcal{A}$, namely

$$\mathbb{C}^{\varepsilon}_{f_{\mathcal{C}},a} \stackrel{def}{=} \{f_{\mathcal{A}} \mid \delta_{\mathcal{C}}(f_{\mathcal{C}}(\gamma(a)), \gamma(f_{\mathcal{A}}(a))) \leq \varepsilon\}$$

then for all $\xi \geq \varepsilon$: $\mathbb{C}^{\varepsilon}_{f_{\mathcal{C}},a} \subseteq \mathbb{C}^{\xi}_{f_{\mathcal{C}},a}$. The second point of Proposition 3 simply states that any sound approximation $f_{\mathcal{A}}$ of $f_{\mathcal{C}}$ is partial complete when we admit an infinite level of imprecision.

```
var x : int, y : int;
begin
  x = 0; y = 0;
  while (x <= 9) and (y >= 0) do
    if x <= 4 then
      x = x + 1; y = y + 1;
    else
      x = x + 1; y = y − 1;
    endif;
  done;
end
```

Fig. 1: The Program P

```
var x : int;
begin
  while x > 0 do
    x = x − 1;
  done;
end
```

Fig. 2: The Program Q

5.1 An Example on Static Analysis of Numeric Invariants

We want to analyze the partial completeness of the Interproc[4] [2] static analyzer when used to infer the numerical invariant of the while-loop of program P defined in Fig. 1 using the abstract domains $\mathcal{A} \in \{\mathsf{Oct}, \mathsf{Poly}\}$. The imprecision generated by the abstract execution $[\![P]\!]_{\mathcal{A}}$ with respect to the concrete (collecting) execution $[\![P]\!]$, is measured by using the following pre-metric $\subseteq$-compatible on $(\wp(\mathbb{Z}^n), \subseteq)$:

$$\%Vol(S_1, S_2) \stackrel{def}{=} \frac{(Vol(\alpha_{\mathsf{Int}^n}(S_2)) - Vol(\alpha_{\mathsf{Int}^n}(S_1))) \cdot 100}{Vol(\alpha_{\mathsf{Int}^n}(S_1))}$$

Intuitively, the value returned by $\%Vol(S_1, S_2)$ is to be interpreted as the percentage of more volume that the abstraction $(\alpha_{\mathsf{Int}^n})$ of S_2 into Int^n has compared to the volume of the abstraction of S_1 into Int^n, namely, $Vol(\alpha_{\mathsf{Int}^n}(S_1))$ and $Vol(\alpha_{\mathsf{Int}^n}(S_2))$ are the volumes of the two hyperrectangles representing respectively S_1 and S_2. Calculating the exact volume of hyperrectangles is in general much less computational expensive than computing volumes of octagons and polyhedra, so this choice can be a good trade-off. In our case example, $n = 2$ since P has two variables so that Int^2 represents rectangles and $Vol(\alpha_{\mathsf{Int}^2}(S))$ is the area of the rectangle $\alpha_{\mathsf{Int}^2}(S)$. Note that, since $[\![P]\!] \subseteq \gamma_{\wp(\mathbb{Z}^2)}([\![P]\!]_{\mathsf{Poly}}) \subseteq \gamma_{\wp(\mathbb{Z}^2)}([\![P]\!]_{\mathsf{Oct}})$ where $\gamma_{\wp(\mathbb{Z}^2)}$ is the concretization of Oct and Poly into $\wp(\mathbb{Z}^2)$, then, thanks to axiom 2, we are sure that

$$\begin{aligned} \%Vol([\![P]\!], \gamma_{\wp(\mathbb{Z}^2)}([\![P]\!]_{\mathsf{Poly}})) &\leq \%Vol([\![P]\!], \gamma_{\wp(\mathbb{Z}^2)}([\![P]\!]_{\mathsf{Oct}})) \\ \%Vol(\gamma_{\wp(\mathbb{Z}^2)}([\![P]\!]_{\mathsf{Poly}}), \gamma_{\wp(\mathbb{Z}^2)}([\![P]\!]_{\mathsf{Oct}})) &\leq \%Vol([\![P]\!], \gamma_{\wp(\mathbb{Z}^2)}([\![P]\!]_{\mathsf{Oct}})) \end{aligned}$$

[4] Interproc is freely available at http://pop-art.inrialpes.fr/interproc/interprocweb.cgi

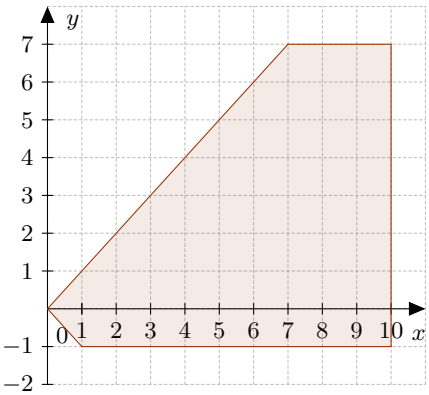

Fig. 3: Invariant generated by $[\![P]\!]_{\mathsf{Oct}}$

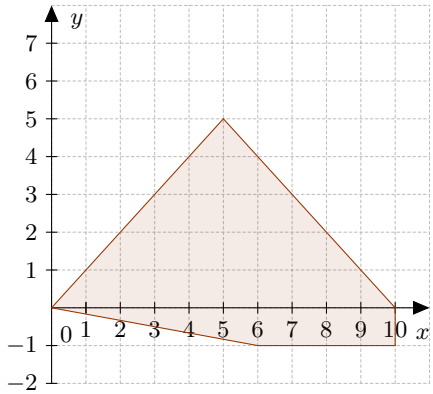

Fig. 4: Invariant generated by $[\![P]\!]_{\mathsf{Poly}}$

always hold regardless of program P. This means that $\%Vol$ can estimate *how more inaccurate* is $[\![P]\!]_{\mathsf{Oct}}$ compared to $[\![P]\!]_{\mathsf{Poly}}$, $[\![P]\!]_{\mathsf{Poly}}$ compared to $[\![P]\!]$, and $[\![P]\!]_{\mathsf{Oct}}$ compared to $[\![P]\!]$.

Suppose our imprecision tolerance measured by $\%Vol$ is 20%. We want to check when $\%Vol([\![P]\!], \gamma_{\wp(\mathbb{Z}^2)}([\![P]\!]_{\mathcal{A}})) \leq 20$ holds, i.e., whether $[\![P]\!]_{\mathsf{Oct}}$ and $[\![P]\!]_{\mathsf{Poly}}$ are 20-partial complete. By running Interproc using Oct and Poly[5] we get the following inequalities representing the inferred while-loop invariants:

$$\begin{aligned}[\![P]\!]_{\mathsf{Oct}} &= \{x >= 0; -x + 10 >= 0; -x + y + 11 >= 0; x + y >= 0;\\ &\quad y + 1 >= 0; -x - y + 17 >= 0; x - y >= 0; -y + 7 >= 0\}\\ [\![P]\!]_{\mathsf{Poly}} &= \{-x - y + 10 >= 0; -x + 10 >= 0; y + 1 >= 0;\\ &\quad x - y >= 0; x + 6y >= 0\}\end{aligned}$$

Fig. 3 and Fig. 4 depict, respectively, $[\![P]\!]_{\mathsf{Oct}}$ and $[\![P]\!]_{\mathsf{Poly}}$. The pre-metric $\%Vol$ outputs:

$$\begin{aligned}\%Vol(\gamma_{\wp(\mathbb{Z}^2)}([\![P]\!]_{\mathsf{Poly}}), \gamma_{\wp(\mathbb{Z}^2)}([\![P]\!]_{\mathsf{Oct}})) &= 33.33\\ \%Vol([\![P]\!], \gamma_{\wp(\mathbb{Z}^2)}([\![P]\!]_{\mathsf{Poly}})) &= 20\\ \%Vol([\![P]\!], \gamma_{\wp(\mathbb{Z}^2)}([\![P]\!]_{\mathsf{Oct}})) &= 60\end{aligned}$$

These numbers validate the better accuracy of $[\![P]\!]_{\mathsf{Poly}}$ compared to $[\![P]\!]_{\mathsf{Oct}}$ by providing us a quantitative estimation: the rectangle representing $[\![P]\!]_{\mathsf{Oct}}$ has 33.33% more volume than $[\![P]\!]_{\mathsf{Poly}}$, the one representing $[\![P]\!]_{\mathsf{Poly}}$ has 20% more volume than the concrete execution $[\![P]\!]$, while $[\![P]\!]_{\mathsf{Oct}}$ has 60% more volume than $[\![P]\!]$. We can conclude that $[\![P]\!]_{\mathsf{Poly}}$ is 20-partial complete whereas $[\![P]\!]_{\mathsf{Oct}}$ is not.

It is worth noting that, the same results can be drawn by defining a similar (computationally more efficient) pre-metric compatible with the Oct domain

[5] For the convex polyhedra analysis, we activated the option of 2 descending steps.

$\%Vol(o_1, o_2)$ with $o_1, o_2 \in \mathsf{Oct}$ which abstracts octagons into boxes, thus calculating for instance $\%Vol(\alpha_{\mathsf{Oct}}([\![P]\!]), [\![P]\!]_{\mathsf{Oct}})$ without passing through the concrete domain $\wp(\mathbb{Z}^2)$.

5.2 Characterizing Forward/Backward-Completeness

The original definition of partial completeness given in [10] makes use of quasi-metrics satisfying the identity of indiscernibles axiom, thus, able to recognize identical elements as the value 0 is assigned if and only if the two elements are equal. By using quasi-metrics, asking for the 0-partial completeness is equivalent to require the backward-completeness [10].

In our weaker framework where pre-metrics are involved, requiring 0-partial completeness may not coincide with requiring the forward/backward-completeness property: given two concrete elements $c_1, c_2 \in \mathcal{C}$ such that $c_1 \preceq_{\mathcal{C}} c_2$, the pre-metric $\preceq_{\mathcal{C}}$-compatible chosen may assign $\delta_{\mathcal{C}}(c_1, c_2) = 0$ even if $c_1 \neq c_2$, namely, $\delta_{\mathcal{C}}$ may not be precise enough to distinguish that two elements are not equal.

Example 9. Given $(\wp(\mathbb{Z}^2), \subseteq)$, consider the pre-metric $\subseteq$-compatible $\%Vol$ defined in Section 5.1. The $\%Vol$ of the following two sets $S_1 = \{\langle 0, 2\rangle, \langle 3, 5\rangle\}$, $S_2 = \{\langle 0, 2\rangle, \langle 1, 1\rangle, \langle 3, 5\rangle\}$, is $\%Vol(S_1, S_2) = 0$ even if $S_1 \neq S_2$. This is because of the approximation made by $\%Vol$, namely, $\alpha_{\mathsf{Int}^2}(S_1) = \langle [0, 3], [1, 5]\rangle = \alpha_{\mathsf{Int}^2}(S_2)$. Therefore, if S_1 and S_2 are the results of, respectively, a concrete operator $f_{\wp(\mathbb{Z}^2)}$ and and abstract operator f_{Int^2}, then f_{Int^2} is 0-partial complete but not forward-complete. ♦

However, it turns out that the 0-partial completeness property coincides with the forward-completeness property when the pre-metric $\preceq_D$-compatible is strong.

Theorem 1. *If $\delta_{\mathcal{C}}$ is strong then the following equivalence holds for all $a \in \mathcal{A}$:*

$$f_{\mathcal{A}} \text{ 0-partial complete} \Leftrightarrow f_{\mathcal{A}} \text{ forward-complete} \qquad \square$$

Example 10. If we define the weighted path-length directly on $(\wp(\mathbb{Z}), \subseteq)$, namely, $\delta^{\mathfrak{w}}_{\wp(\mathbb{Z})}$ where $\mathfrak{w}(S_1, S_2) = 1$ for all $(S_1, S_2) \in E_{\wp(\mathbb{Z})}$, then $\delta^{\mathfrak{w}}_{\wp(\mathbb{Z})}$ is strong. Consider the program Q defined in Fig. 2. We analyze the value of variable x at the end of the program having input the interval $[10, 10]$ using Interproc on the interval abstract domain Int. The weighted path-length outputs

$$\delta^{\mathfrak{w}}_{\wp(\mathbb{Z})}([\![Q]\!]\gamma([10, 10]), \gamma([\![Q]\!]_{\mathsf{Int}}[10, 10])) = 0$$

i.e., $[\![Q]\!]_{\mathsf{Int}}$ is 0-partial complete on input $[10, 10]$ using $\delta^{\mathfrak{w}}_{\wp(\mathbb{Z})}$. Since $\delta^{\mathfrak{w}}_{\wp(\mathbb{Z})}$ is strong, this implies $[\![Q]\!]\gamma([10, 10]) = \gamma([\![Q]\!]_{\mathsf{Int}}[10, 10])$, i.e., $[\![Q]\!]_{\mathsf{Int}}$ is forward-complete. ♦

Furthermore, when the concrete and abstract domains admit a GI through an abstraction function $\alpha : \mathcal{C} \to \mathcal{A}$ and $\mathcal{A}$ is equipped with a pre-metric $\preceq_{\mathcal{A}}$-compatible $\delta_{\mathcal{A}}$, we can also characterize the backward-completeness property over representable elements of $\mathcal{C}$ as an instance of the partial completeness property by exploiting the induced pre-metric $\overleftarrow{\delta_{\mathcal{C}}}$ from the abstract pre-metric $\preceq_{\mathcal{A}}$-compatible space .

Theorem 2. *Let* $(\mathcal{C}, \preceq_{\mathcal{C}}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\mathcal{A}, \preceq_{\mathcal{A}})$ *be a GI,* $(\mathcal{A}, \preceq_{\mathcal{A}}, \delta_{\mathcal{A}}) \in Pre((\mathcal{A}, \preceq_{\mathcal{A}}))$ *and* $(\mathcal{C}, \preceq_{\mathcal{C}}, \overleftarrow{\delta}_{\mathcal{C}}) \in Pre((\mathcal{C}, \preceq_{\mathcal{C}}))$*. If* $\delta_{\mathcal{A}}$ *is strong then the following equivalence holds for every* $a \in \mathcal{A}$*:*

$$f_{\mathcal{A}} \text{ 0-partial complete} \Leftrightarrow f_{\mathcal{A}} \text{ backward-complete.} \qquad \square$$

Example 11. Let us use the weighted path-length $\delta^{\mathfrak{w}}_{\mathsf{Int}}$ on intervals for reasoning on the backward-completeness of program Q defined in Fig. 2. Recall that $\delta^{\mathfrak{w}}_{\mathsf{Int}}$ is a strong pre-metric $\preceq_{\mathsf{Int}}$-compatible. We get

$$\overleftarrow{\delta}_{\wp(\mathbb{Z})}(\llbracket Q \rrbracket \gamma([10,10]), \gamma(\llbracket Q \rrbracket_{\mathsf{Int}}[10,10])) = \delta^{\mathfrak{w}}_{\mathsf{Int}}(\alpha(\llbracket Q \rrbracket \gamma([10,10])), \llbracket Q \rrbracket_{\mathsf{Int}}[10,10]) = 0$$

namely $\llbracket Q \rrbracket_{\mathsf{Int}}$ is 0-partial complete on input $[10,10]$ using $\overleftarrow{\delta}_{\wp(\mathbb{Z})}$. Since $\delta^{\mathfrak{w}}_{\mathsf{Int}}$ is strong, this implies that $\alpha(\llbracket Q \rrbracket \gamma([10,10])) = \llbracket Q \rrbracket_{\mathsf{Int}}[10,10]$, i.e., $\llbracket Q \rrbracket_{\mathsf{Int}}$ is backward-complete. ⧫

6 Related Work

Forward and backward completeness are well known notions in abstract interpretation, especially in static program analysis for verifying safety program properties [18,30,26]. The first attempt to weaken the notion of backward-completeness in abstract interpretation has been defined in [5]. Here the authors introduced the notion of local completeness which corresponds to our definition of backward-completeness (Definition 3). Partial completeness has been recently introduced [10,9] as a further weakening of the local completeness property by admitting a limited amount of imprecision measured by a quasi-metric compatible with the underlying abstract domain.

Besides the partial completeness property, the problem of measuring the imprecision of abstract interpretations is not new. Sotin [38] defines a metric to quantify the result of numerical invariants by calculating the size of the concretization into $\mathbb{R}^n$. This metric can be considered as an instance of pre-metric $\subseteq$-compatible, thus it can be used for our new generalized partial completeness property.

Crazzolara [23] proposes to substitute partial-orders with quasi-metrics, namely, the concrete and abstract partially-ordered set turn into quasi-metric spaces. Our approach, instead, preserves the standard abstract interpretation framework and considers the distances as external tools for measuring the incompleteness of abstract operators. A similar idea is proposed by Di Pierro and Wiklicky [37] where partially-ordered domains are replaced by vector spaces lifting abstract interpretation to a probabilistic version where it is possible to apply some well-known distances in linear spaces.

Logozzo et al. [33] adapt the notion of pseudo-metric to be compatible with partially-ordered sets in order to measure the distance between two elements. Their definition of pseudo-metric requires the weak triangle inequality axiom and symmetry, while our definition of pre-metric relaxes those axioms. Moreover,

axiom 2 may not be satisfied by pseudo-metrics, therefore their distances may not fit well in the partial completeness property.

Casso et al. [11] proposes a list of observations about distance functions when used to measure distances between elements of abstract domains in the context of logic programming. They show that it is possible to induce other distances from one domain to another through the concretization and abstraction functions in a similar way we did in Section 4. However, their notion of distance requires more compatibility with the underlying lattice than our approach as they focus on abstract domains commonly used for analyzing logic programs. For instance, they assume abstract domains to be complete lattices related by a GI with the concrete domains, they require another type of triangle inequality called diamond inequality, and consider distances between comparable elements only. As our notion of pre-metric is weaker than what they require for distances, our framework can be instantiated with their distances.

7 Conclusion

We generalized the partial completeness property with the aim of weakening both the backward-completeness, in presence of a GC, and the forward-completeness properties in case only a concretization function is available, e.g., the case of convex polyhedra or the domain of formal languages [8]. The definition of pre-metrics is general enough to be instantiated by distance functions having different "levels of view". For instance, a distance may be precise enough to satisfies the identity of indiscernibles axiom on the concrete domain, so that it can be used to reason on the forward-completeness. Different levels of approximation can be obtain by inducing pre-metrics from one domain to another by the use of the concretization or the abstraction maps. Our framework could help program analysis designers to control the incompleteness propagation, e.g., by checking how an invariant generated by the analysis grows with respect to the concrete execution or another comparable analysis. This checking could be combined also with other repairing techniques aiming to enrich the expressiveness of abstract domains [30,6]. Although the generalized partial completeness is still an undecidable property similar to the other completeness properties [26,4,10,9], as a future work we plan to extend the proof system proposed in [10] in order to be able to overestimate, according to δ_D, a bound of imprecision generated by the abstract interpreter without actually executing the program. This, in fact, can be considered as another abstract interpretation analyzing the abstract interpreter [21]. Studying the incompleteness propagation through pre-metrics has a strong connection to code obfuscation [14,25,27,28]. Being able to quantify the amount of incompleteness induced in the abstract interpretation by a code obfuscating program transformation could allow us to measure the potency of these transformations. This is still one of the main open challenges in software protection [39,12,15].

References

1. https://en.wikipedia.org/wiki/Metric_space#Premetrics
2. http://pop-art.inrialpes.fr/people/bjeannet/bjeannet-forge/interproc/index.html
3. Arkhangel'Skii, A., Fedorchuk, V.: General topology I: basic concepts and constructions dimension theory, vol. 17. Springer (2012)
4. Bruni, R., Giacobazzi, R., Gori, R., Garcia-Contreras, I., Pavlovic, D.: Abstract extensionality: on the properties of incomplete abstract interpretations. Proc. ACM Program. Lang. **4**(POPL), 28:1–28:28 (2020). https://doi.org/10.1145/3371096
5. Bruni, R., Giacobazzi, R., Gori, R., Ranzato, F.: A logic for locally complete abstract interpretations. In: 36th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2021, Rome, Italy, June 29 - July 2, 2021. pp. 1–13. IEEE (2021). https://doi.org/10.1109/LICS52264.2021.9470608
6. Bruni, R., Giacobazzi, R., Gori, R., Ranzato, F.: Abstract interpretation repair. In: Jhala, R., Dillig, I. (eds.) PLDI '22: 43rd ACM SIGPLAN International Conference on Programming Language Design and Implementation, San Diego, CA, USA, June 13 - 17, 2022. pp. 426–441. ACM (2022). https://doi.org/10.1145/3519939.3523453
7. Buldygin, V.V., Kozachenko, I.V.: Metric characterization of random variables and random processes, vol. 188. American Mathematical Soc. (2000). https://doi.org/10.1090/mmono/188
8. Campion, M., Dalla Preda, M., Giacobazzi, R.: Abstract interpretation of indexed grammars. In: International Static Analysis Symposium. pp. 121–139. Springer (2019). https://doi.org/10.1007/978-3-030-32304-2_7
9. Campion, M., Preda, M.D., Giacobazzi, R.: On the properties of partial completeness in abstract interpretation. In: Lago, U.D., Gorla, D. (eds.) Proceedings of the 23rd Italian Conference on Theoretical Computer Science, ICTCS 2022, Rome, Italy, September 7-9, 2022. CEUR Workshop Proceedings, vol. 3284, pp. 79–85. CEUR-WS.org (2022), http://ceur-ws.org/Vol-3284/8665.pdf
10. Campion, M., Preda, M.D., Giacobazzi, R.: Partial (in)completeness in abstract interpretation: limiting the imprecision in program analysis. Proc. ACM Program. Lang. **6**(POPL), 1–31 (2022). https://doi.org/10.1145/3498721
11. Casso, I., Morales, J.F., López-García, P., Giacobazzi, R., Hermenegildo, M.V.: Computing abstract distances in logic programs. In: International Symposium on Logic-Based Program Synthesis and Transformation. pp. 57–72. Springer (2019). https://doi.org/10.1007/978-3-030-45260-5_4
12. Ceccato, M., Tonella, P., Basile, C., Falcarin, P., Torchiano, M., Coppens, B., Sutter, B.D.: Understanding the behaviour of hackers while performing attack tasks in a professional setting and in a public challenge. Empir. Softw. Eng. **24**(1), 240–286 (2019). https://doi.org/10.1007/s10664-018-9625-6
13. Cohen, J., Hickey, T.J.: Two algorithms for determining volumes of convex polyhedra. J. ACM **26**(3), 401–414 (1979). https://doi.org/10.1145/322139.322141
14. Collberg, C., Nagra, J.: Surreptitious Software: Obfuscation, Watermarking, and Tamperproofing for Software Protection. Addison-Wesley Professional (2009)
15. Collberg, C.S., Davidson, J.W., Giacobazzi, R., Gu, Y.X., Herzberg, A., Wang, F.: Toward digital asset protection. IEEE Intelligent Systems **26**(6), 8–13 (2011). https://doi.org/10.1109/MIS.2011.106
16. Cousot, P.: Principles of Abstract Interpretation. The MIT Press, Cambridge, Mass. (2021)

17. Cousot, P., Cousot, R.: Static determination of dynamic properties of programs. In: Proceedings of the 2nd International Symposium on Programming. pp. 106–130. Dunod, Paris (1976). https://doi.org/10.1145/390019.808314
18. Cousot, P., Cousot, R.: Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints. In: Graham, R.M., Harrison, M.A., Sethi, R. (eds.) Proceedings of the 4th ACM Symposium on Principles of Programming Languages, Los Angeles, California, USA, January 1977. pp. 238–252. ACM (1977). https://doi.org/10.1145/512950.512973
19. Cousot, P., Cousot, R.: Systematic design of program analysis frameworks. In: Aho, A.V., Zilles, S.N., Rosen, B.K. (eds.) Proceedings of the 6th ACM Symposium on Principles of Programming Languages, San Antonio, Texas, USA, January 1979. pp. 269–282. ACM Press (1979). https://doi.org/10.1145/567752.567778
20. Cousot, P., Cousot, R.: Abstract interpretation frameworks. J. Log. Comput. **2**(4), 511–547 (1992). https://doi.org/10.1093/logcom/2.4.511
21. Cousot, P., Giacobazzi, R., Ranzato, F.: A^2i: Abstract^2 interpretation. Proc. ACM Program. Lang. **3**(POPL) (Jan 2019). https://doi.org/10.1145/3290355
22. Cousot, P., Halbwachs, N.: Automatic discovery of linear restraints among variables of a program. In: Aho, A.V., Zilles, S.N., Szymanski, T.G. (eds.) Conference Record of the Fifth Annual ACM Symposium on Principles of Programming Languages, Tucson, Arizona, USA, January 1978. pp. 84–96. ACM Press (1978). https://doi.org/10.1145/512760.512770
23. Crazzolara, F.: Quasi-metric spaces as domains for abstract interpretation. In: Falaschi, M., Navarro, M., Policriti, A. (eds.) 1997 Joint Conf. on Declarative Programming, APPIA-GULP-PRODE'97, Grado, Italy, June 16-19, 1997. pp. 45–56 (1997)
24. Distefano, D., Fähndrich, M., Logozzo, F., O'Hearn, P.W.: Scaling static analyses at facebook. Commun. ACM **62**(8), 62–70 (2019). https://doi.org/10.1145/3338112
25. Giacobazzi, R.: Hiding information in completeness holes: New perspectives in code obfuscation and watermarking. In: Cerone, A., Gruner, S. (eds.) Sixth IEEE International Conference on Software Engineering and Formal Methods, SEFM 2008, Cape Town, South Africa, 10-14 November 2008. pp. 7–18. IEEE Computer Society (2008). https://doi.org/10.1109/SEFM.2008.41
26. Giacobazzi, R., Logozzo, F., Ranzato, F.: Analyzing program analyses. In: Rajamani, S.K., Walker, D. (eds.) Proceedings of the 42nd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2015, Mumbai, India, January 15-17, 2015. pp. 261–273. ACM (2015). https://doi.org/10.1145/2676726.2676987
27. Giacobazzi, R., Mastroeni, I.: Making abstract interpretation incomplete: Modeling the potency of obfuscation. In: Miné, A., Schmidt, D. (eds.) Static Analysis - 19th International Symposium, SAS 2012, Deauville, France, September 11-13, 2012. Proceedings. Lecture Notes in Computer Science, vol. 7460, pp. 129–145. Springer (2012). https://doi.org/10.1007/978-3-642-33125-1_11
28. Giacobazzi, R., Mastroeni, I., Preda, M.D.: Maximal incompleteness as obfuscation potency. Formal Aspects Comput. **29**(1), 3–31 (2017). https://doi.org/10.1007/s00165-016-0374-2
29. Giacobazzi, R., Quintarelli, E.: Incompleteness, counterexamples, and refinements in abstract model-checking. In: Cousot, P. (ed.) Static Analysis, 8th International Symposium, SAS 2001, Paris, France, July 16-18, 2001, Proceedings. Lecture Notes in Computer Science, vol. 2126, pp. 356–373. Springer (2001). https://doi.org/10.1007/3-540-47764-0_20

30. Giacobazzi, R., Ranzato, F., Scozzari, F.: Making abstract interpretations complete. J. ACM **47**(2), 361–416 (2000). https://doi.org/10.1145/333979.333989
31. Helemskii, A.Y.: Lectures and Exercises on Functional Analysis, vol. 233. American Mathematical Soc. (2006). https://doi.org/10.1090/mmono/233
32. Lawrence, J.: Polytope volume computation. Mathematics of Computation **57**(195), 259–271 (Jul 1991). https://doi.org/10.1090/S0025-5718-1991-1079024-2
33. Logozzo, F.: Towards a quantitative estimation of abstract interpretations. In: Workshop on Quantitative Analysis of Software. Microsoft (June 2009), https://www.microsoft.com/en-us/research/publication/towards-a-quantitative-estimation-of-abstract-interpretations/
34. Miné, A.: A new numerical abstract domain based on difference-bound matrices. In: Danvy, O., Filinski, A. (eds.) Programs as Data Objects, Second Symposium, PADO 2001, Aarhus, Denmark, May 21-23, 2001, Proceedings. Lecture Notes in Computer Science, vol. 2053, pp. 155–172. Springer (2001). https://doi.org/10.1007/3-540-44978-7_10
35. Miné, A.: The octagon abstract domain. In: Burd, E., Aiken, P., Koschke, R. (eds.) Proceedings of the Eighth Working Conference on Reverse Engineering, WCRE'01, Stuttgart, Germany, October 2-5, 2001. p. 310. IEEE Computer Society (2001). https://doi.org/10.1109/WCRE.2001.957836
36. Miné, A.: Tutorial on static inference of numeric invariants by abstract interpretation. Foundations and Trends in Programming Languages **4**(3-4), 120–372 (2017). https://doi.org/10.1561/2500000034
37. Pierro, A.D., Wiklicky, H.: Measuring the precision of abstract interpretations. In: Lau, K. (ed.) Logic Based Program Synthesis and Transformation, 10th International Workshop, LOPSTR 2000 London, UK, July 24-28, 2000, Selected Papers. Lecture Notes in Computer Science, vol. 2042, pp. 147–164. Springer (2000). https://doi.org/10.1007/3-540-45142-0_9
38. Sotin, P.: Quantifying the precision of numerical abstract domains. Tech. Rep. HAL Id: inria-00457324, INRIA (2010), https://hal.inria.fr/inria-00457324
39. Sutter, B.D., Collberg, C.S., Preda, M.D., Wyseur, B.: Software protection decision support and evaluation methodologies (dagstuhl seminar 19331). Dagstuhl Reports **9**(8), 1–25 (2019). https://doi.org/10.4230/DagRep.9.8.1