Abstract Interpretation Frameworks

Patrick Cousot
LIENS, École Normale Supérieure
45, rue d'Ulm
75230 Paris cedex 05 (France)
cousot@dmi.ens.fr

Radhia Cousot
LIX, École Polytechnique
91128 Palaiseau cedex (France)
radhia@polytechnique.fr

Abstract

We introduce abstract interpretation frameworks which are variations on the archetypal framework using Galois connections between concrete and abstract semantics, widenings and narrowings and are obtained by relaxation of the original hypotheses. We consider various ways of establishing the correctness of an abstract interpretation depending on how the relation between the concrete and abstract semantics is defined. We insist upon those correspondences allowing for the inducing of the approximate abstract semantics from the concrete one. Furthermore we study various notions of widening and narrowing as a means of obtaining convergence in the iterations used in abstract interpretation.

Keywords: Abstract interpretation; standard and collecting semantics; concrete and abstract semantics; discrete approximation; soundness relation; abstraction; concretization; Galois connections; widening; narrowing;

1 Introduction

The *semantics* of programs describes the set of all possible behaviours of these programs when executed for all possible input data. For example, these behaviours can be non-termination, termination with an error or correct termination delivering one or more output results.

Abstract interpretation is a method for designing approximate semantics of programs which can be used to gather information about programs in order to provide sound answers to questions about their run-time behaviours. These semantics can be used to design manual proof methods or to specify automatic program analysers. When the semantic analysis of programs is to be automated, the answers can only be partial or approximate since questions such as termination for all input data are undecidable. The purpose of abstract interpretation is to prove the soundness of such program analysis methods with respect to a semantics, or better to formally design them by approximation of the semantics of programs. Hence from a theoretical point of view, the purpose of abstract interpretation is to design hierarchies of interrelated semantics specifying at various levels of details the behaviour of programs when executed by computers. This corresponds to understanding of 'interpret' as 'to explain the meaning of', the qualification as 'abstract' enforcing 'to understand in a specified way'. From a practical point of view, the purpose of abstract interpretation is to design automatic program analysis tools for determining statically dynamic properties of programs. This corresponds to understanding of 'interpret' as 'to act as an interpreter', the qualification as 'abstract' enforcing the idea that the concrete domain of values is replaced by a domain of descriptions of values and concrete operators are given a corresponding non-standard interpretation.

The abstract interpretation framework that we introduced in [6, 8, 7, 9, 10, 12, 18, 19] is based on the use of Galois connections (or equivalently closure operators, Moore families, complete join congruence relations or families of principal ideals) to establish the correspondence between the domain of concrete or exact properties and the domain of abstract or approximate properties. This corresponds to a perfect situation, where each concrete property has a unique best abstract approximation. It is theoretically always possible to satisfy this hypothesis, by adding more properties to the abstract domain. However, in some practical cases, this might lead to a combinatorial explosion (as observed in paragraph 9.2 of [12]). In this case and more generally, when the abstract domain is large or infinite, widening and narrowing operators [6, 7] should be used to tune the cost/precision compromise. In the following presentation of this framework, we first relax this uniqueness condition to study more flexible but less elaborated variants of the original framework, which is finally obtained as a powerful special case. The study covers most of the cases that we have encountered in our practical experience with abstract interpretation. However, not all situations can be exhaustively considered so that the presentation of the abstract interpretation frameworks should be understood as themes on which further variations can be constructed.

2 Standard, collecting and abstract semantics

Since an abstract interpretation of programs is an approximation of their semantics, the correctness proof of an abstract interpretation requires the existence of the *standard semantics* describing the possible behaviours of programs during their execution. Then, the abstract interpretation focuses on a class of properties of program executions, which is usually defined by a *collecting semantics* (also called *static* in [7]). This collecting semantics can be an instrumented version of the standard semantics in order to gather information about programs executions. Alternatively, it can also be a version of the standard semantics reduced to essentials in order to ignore irrelevant details about program execution. The main utilization of the collecting semantics is to provide a sound and relatively complete proof method for the considered class of properties. It can be used subsequently as a reference semantics for proving the correctness of all other approximate semantics for that class of properties. Finally, the *abstract semantics* usually considers effectively computable properties of programs. It is therefore usually incomplete. The soundness of this abstract semantics is proved with respect to the collecting semantics.

Example 2.1 (Invariants)
In [12], the standard semantics is an operational semantics specified by a transition system $\langle S, t, \iota, \varsigma\rangle$ which consists of a set S of states, a subset $\iota \subseteq S$ of initial states, a subset $\varsigma \subseteq S$ of final states and a binary transition relation $t \in \wp(S \times S)$ between a state and its possible successors.[1] Two collecting semantics of a program are considered where programs properties are represented by sets of finite partial execution traces. In the forward collecting semantics, traces are finite sequences of states, starting with an initial state and such that two consecutive states on the trace satisfy the

[1] The fact that the abstract interpretation framework of [7, 12, 18] is applicable not only to flowchart programs but to other languages defined by their operational semantics, including first-order functions considered in [10] and parallel programs considered in [13, 14], and more generally fixpoint based semantics, is sometimes misunderstood.

transition relation. In the backward collecting semantics, traces are finite sequences of states, starting with a final state and such that two consecutive states on the trace satisfy the inverse of the transition relation. Various abstractions are considered, in particular the descendant states of the initial states, which happened to be the collecting semantics considered in [7]. This abstraction consists in approximating a set of forward traces by the set of states which occur on any one of these traces. It is appropriate for discussing invariance properties of programs.

Example 2.2 (Mycroft's strictness analysis)
Another original and important example was given by Mycroft [34, 35]. Strictness analysis, which consists in determining whether the result of a function is undefined whenever its argument is undefined is useful for speeding up sequential or parallel implementations of lazy functional languages [5]. There the standard semantics is a denotational semantics. The collecting semantics is obtained using powerdomains so as to specify the possible results of a function given a set of possible values for the actual arguments, including undefined, that is non-terminating ones [30]. The abstract semantics concern strictness and termination analyses.

In order to describe the correspondences between standard and collecting, collecting and abstract or standard and abstract semantics in a uniform formalism, we now study such connections between concrete and abstract semantics. The fact of being concrete or abstract is relative and simply means that the abstract semantics is an approximation of the concrete semantics (including inducings into another semantic domain without loss or gain of information). For example the collecting semantics is abstract with respect to the standard semantics and concrete for subsequent abstract interpretations.

We will examine various means of establishing a correspondence between concrete and abstract semantics.

3 Iterative specification of the concrete and abstract semantics

But first we must hypothesize how the concrete and abstract semantics are defined.

Concrete semantic domain. The concrete semantics describes properties of the possible executions of a program represented by means of *concrete semantic properties* c chosen in a given set $\mathcal{P}^\natural$ called the *concrete semantic domain.*

For example elements c of $\mathcal{P}^\natural$ can be a set of maximal execution traces, an input-output relation, a function, a set of states, etc. The design of a concrete semantics giving a semantic description of a class of properties of the possible executions of a program may be a non-trivial task. This might not be so much a problem when the properties of interest can be characterized directly with respect to the standard semantics (this can be done for Mycroft's strictness analysis as shown in forthcoming Example 6.11) or when there is a simple construction for deriving the collecting semantics from the standard semantics (which is the case for invariance properties as shown in Example 3.2). This may be more difficult in analyses such as binding-time analysis, where the relevant properties are of second-order nature [28], or in an analysis of storage use, where the properties must be formulated with respect to some model of the store [21]. However, we deliberately ignore the distinction between the semantics of a program and the properties that this program semantics satisfies so as to remain fully comprehensive.

Concrete semantics. A semantics definition associates with each program its *concrete semantics*, which is a concrete semantic property c chosen in $\mathcal{P}^\natural$ representing some characteristics of its possible executions. Several hypotheses can be made on the methods which can be used to define this concrete semantics of a program:

- It is usually obtained as the limit $F^{\natural\epsilon}$ of a transfinite and ultimately stationary sequence $F^{\natural\lambda}$ of elements of $\mathcal{P}^\natural$, for all ordinals $\lambda \in \mathit{Ord}$. This sequence $F^{\natural\lambda}$, $\lambda \in \mathit{Ord}$ will be called the *concrete iteration* and its elements the *concrete iterates*.
- More specifically, the concrete iteration is often specified by transfinite recursion using a basis $\underline{\bot}^\natural \in \mathcal{P}^\natural$, a partial map $F^\natural \in \mathcal{P}^\natural \rightarrowtail \mathcal{P}^\natural$ called the *concrete semantic function* (where $S \rightarrowtail T$ is the set of *partial functions* of the set S into the set T) and an *inductive join* $\amalg^\natural \in \wp(\mathcal{P}^\natural) \rightarrowtail \mathcal{P}^\natural$ so that:

$$\left\{\begin{array}{rcll} F^{\natural 0} & = & \underline{\bot}^\natural & \\ F^{\natural\lambda+1} & = & F^\natural(F^{\natural\lambda}) & \\ F^{\natural\lambda} & = & \amalg^\natural_{\beta<\lambda} F^{\natural\beta} & \text{when } \lambda > 0 \text{ is a limit ordinal.} \end{array}\right. \tag{4.1}$$

  Since partial operations $F^\natural$ and $\amalg^\natural$ are used, we say that the iteration is *total* when all its iterates are well-defined and otherwise that it is *partial*. The iteration is said to be *convergent* with limit $F^{\natural\epsilon}$ whenever it is total and ultimately stationary, that is to say $\exists \epsilon \in \mathit{Ord} : \forall \lambda \geq \epsilon : F^{\natural\lambda} = F^{\natural\epsilon}$. Many other forms of sequence definitions can be considered, such as $F^{\natural\lambda} = F^\natural\big(\amalg^\natural_{\beta<\lambda} F^{\natural\beta}\big) \amalg^\natural \big(\amalg^\natural_{\beta<\lambda} F^{\natural\beta}\big)$ for all $\lambda \in \mathit{Ord}$.
- In particular, the concrete iterates may be in increasing order for a partial order $\sqsubseteq^\natural \in \wp(\mathcal{P}^\natural \times \mathcal{P}^\natural)$. This partial order relation may induce a complete partial order (cpo) or even a complete lattice structure on $\mathcal{P}^\natural$ and $\amalg^\natural$ may be the corresponding least upper bound $\sqcup^\natural$ whereas $\underline{\bot}^\natural$ often happens to be the infimum $\bot^\natural$. This ensures that the concrete iteration is convergent. Note, however, that the least upper bounds $\sqcup^\natural_{\beta<\lambda} F^{\natural\beta}$ are needed for the iteration sequence $F^{\natural\lambda}$, $\lambda \leq \epsilon$ only, not for all directed sets of $\mathcal{P}^\natural$. Whence the concrete domain $\mathcal{P}^\natural$ needs not to be a cpo (and it is not for the Smyth order [2]).

Example 3.1 (Denotational semantics)
In denotational semantics [31], the domain $\langle D^\partial \overset{c}{\longmapsto} D^\partial;\ \sqsubseteq^\partial,\ \lambda\, x \cdot \bot^\partial,\ \sqcup^\partial\rangle$ of denotations of recursively defined functions is the cpo of continuous total functions from the domain D^∂ into D^∂ [24], the concrete iteration is given by means of a continuous map $F^\partial \in (D^\partial \overset{c}{\longmapsto} D^\partial) \overset{c}{\longmapsto} (D^\partial \overset{c}{\longmapsto} D^\partial)$. Since the concrete iteration is increasing, its limit exists and is $\sqcup^\partial_{n\geq 0} F^{\partial^n}(\bot^\partial)$. By continuity, $\epsilon = \omega$ (where ω is the first infinite ordinal and we can let subsequent iterates be equal to this limit).

Example 3.2 (Collecting semantics for invariance properties)
Another example is the forward collecting semantics for invariance properties of a transition system $\langle S,\ t,\ \iota,\ \varsigma\rangle$, where S is the set of states, $t \in \wp(S \times S)$ is the

transition relation, $\iota \subseteq S$ is the set of initial states and $\varsigma \subseteq S$ is the set of final states considered in [19]. Program invariance properties can be characterized by sets of states, that is elements of $\wp(S)$ where $\langle \wp(S);\ \subseteq, \emptyset, \cup \rangle$ is a complete lattice. The concrete semantic function is the strongest post-condition operator $sp^t_\iota \in \wp(S) \mapsto \wp(S)$ (where $S \mapsto T$ is the set of *total functions* of the set S into the set T) such that $sp^t_\iota(I) \stackrel{\text{def}}{=} \iota \ \cup\ \{ s \mid \exists s' \in I : \langle s',\ s \rangle \in t \}$. If I is an invariant, then $sp^t_\iota(I)$ is the invariant which holds after one more program step. The set of descendant states of the initial states is given as the limit $\cup_{n \geq 0} {sp^t_\iota}^n(\emptyset)$. $\epsilon = \omega$ since sp^t_ι is a complete $\cup$-morphism, hence continuous. As observed in [12], backward collecting semantics needs no more theoretical study since it is the forward collecting semantics for the inverse transition system $\langle S,\ t^{-1},\ \varsigma,\ \iota \rangle$ and therefore is obtained by duality. A defect of denotational semantics is that this duality principle is not applicable (since the inverse of a function is not a function). However, it is applicable to the relational semantics considered in [16].

Abstract semantic domain. The first basic choice to be made in an abstract interpretation is to design an *abstract semantic domain* $\mathcal{P}^\sharp$ which is an approximate version of the concrete semantic domain $\mathcal{P}^\natural$. For example invariance properties, such as partial correctness, can be represented by a set of states. In full generality, $\mathcal{P}^\sharp$ is just assumed to be a set without further hypotheses on its structure.

Abstract semantics. The objective of an abstract interpretation is to find an abstract property a, if any, in the abstract semantic domain $\mathcal{P}^\sharp$ which is a correct approximation of the concrete semantics $c \in \mathcal{P}^\natural$ of the program, in a sense which remains to be defined. Therefore the second basic choice in an abstract interpretation is to design a method for associating an abstract semantics $a \in \mathcal{P}^\sharp$ to programs.

– A natural idea is to give an abstract interpretation of the concrete iteration $F^{\natural\lambda}$, $\lambda \in \mathit{Ord}$ by means of an *abstract iteration*, $F^{\sharp\lambda}$, $\lambda \in \mathit{Ord}$, which is ultimately stationary and therefore has a limit $F^{\sharp\varepsilon}$.

– As this was the case for the concrete semantics, the abstract iteration can be specified by transfinite recursion using an abstract basis $\dot{\bot}^\sharp \in \mathcal{P}^\sharp$, an abstract semantic function $F^\sharp \in \mathcal{P}^\sharp \rightarrowtail \mathcal{P}^\sharp$ and an abstract inductive join $\amalg^\sharp \in \wp(\mathcal{P}^\sharp) \rightarrowtail \mathcal{P}^\sharp$ so that:

$$\left\{ \begin{array}{rcll} F^{\sharp 0} & = & \dot{\bot}^\sharp & \\ F^{\sharp \lambda+1} & = & F^\sharp(F^{\sharp\lambda}) & \\ F^{\sharp\lambda} & = & \amalg^\sharp_{\beta<\lambda} F^{\sharp\beta} & \quad \text{when } \lambda > 0 \text{ is a limit ordinal.} \end{array} \right. \tag{4.2}$$

– In particular, the abstract iterates may be in increasing order for a partial order $\sqsubseteq^\sharp \in \wp(\mathcal{P}^\sharp \times \mathcal{P}^\sharp)$ which may induce an order structure $\langle \mathcal{P}^\sharp;\ \sqsubseteq^\sharp,\ \bot^\sharp,\ \sqcup^\sharp \rangle$ ensuring that the abstract iteration is convergent.

4 Soundness correspondence between concrete and abstract semantics

We start with the use of a soundness relation σ to reason about the correspondence between a concrete and an abstract semantics. Early examples are the relation between direct and continuation denotational semantics given by [36] and the weakest pre-condition $wp(S\,,\,P)$ understood as a relation between the semantics of statement S and the set of states satisfying predicate P [22].

Soundness relation between concrete and abstract semantics. The third basic choice to be made in the design of an abstract interpretation is to specify the correspondence between the concrete and abstract properties. We can define the meaning of the abstract properties by means of a *soundness relation*:

$$\sigma \in \wp(\mathcal{P}^\natural \times \mathcal{P}^\sharp) \ . \tag{4.3}$$

$\langle c,\ a\rangle \in \sigma$ means that the concrete semantics c of the program has the abstract property a.

Example 4.1 (Invariance properties as an abstraction of execution traces)
If the concrete semantics of a program is the set of its maximal execution traces and its abstract property is invariance represented as a set of states then the correspondence between the concrete semantics and the abstract properties can be defined by a relation which associates with a set of traces any superset of the set of states encountered along these traces (see Example 7.1 for more details).

Example 4.2 (Mycroft's strictness analysis)
In strictness analysis, Mycroft proposed to use $D^\sharp = \{0, 1\}$ and $0 \sqsubseteq^\sharp 0 \sqsubseteq^\sharp 1 \sqsubseteq^\sharp 1$, which is extended pointwise to functions and componentwise to vectors of functions so as to obtain continuous functions $F^\sharp \in (D^\sharp \overset{c}{\longmapsto} D^\sharp) \overset{c}{\longmapsto} (D^\sharp \overset{c}{\longmapsto} D^\sharp)$ on a cpo $\langle D^\sharp \overset{c}{\longmapsto} D^\sharp;\ \sqsubseteq^\sharp,\ \bot^\sharp,\ \sqcup^\sharp\rangle$ where $\bot^\sharp = \lambda\, x \cdot 0$. Since the abstract iteration is increasing and the domain is finite, its limit $\sqcup^\sharp_{n \geq 0} F^{\sharp^n}(\bot^\sharp)$ is reached after finitely many steps.

$\varphi^\sharp \in D^\sharp \overset{c}{\longmapsto} D^\sharp$ is a sound abstraction of $\varphi^\partial \in D^\partial \overset{c}{\longmapsto} D^\partial$ if and only if $\langle \varphi^\partial,\ \varphi^\sharp\rangle \in \sigma$ where:

$$\sigma \ \stackrel{\text{def}}{=}\ \big\{\langle \varphi^\partial,\ \varphi^\sharp\rangle \ \big|\ \begin{array}{l}(\varphi^\sharp(0) = 0 \ \Rightarrow\ \varphi^\partial(\bot^\partial) = \bot^\partial) \wedge \\ (\varphi^\sharp(1) = 0 \ \Rightarrow\ \forall x \in D^\partial : \varphi^\partial(x) = \bot^\partial)\big\}\ .\end{array} \tag{4.4}$$

Observe that $\varphi^\sharp = \lambda\, x \cdot 1$ is a trivial solution to the strictness analysis problem so that the soundness relation σ provides no information on the best abstract interpretations.

On the existence of abstract approximations. Very few hypotheses need to be made on the abstract semantic domain $\mathcal{P}^\sharp$ and the soundness relation σ. A common although not indispensable one is a very weak form of expressiveness. Observe that the abstract interpretation problem would have no solution for a program with semantics c when the set $\{a \mid \langle c,\ a\rangle \in \sigma\}$ is empty. Therefore a common assumption is that every concrete property has an abstract approximation. This is the *existence of abstract approximations assumption* stating:

$$\forall c \in \mathcal{P}^\natural : \exists a \in \mathcal{P}^\sharp : \langle c,\ a\rangle \in \sigma \ . \tag{4.5}$$

Proof of soundness of the abstract semantics. To prove the soundness condition, $\langle F^{\natural\epsilon}, F^{\sharp\varepsilon}\rangle \in \sigma$, one can imagine proceeding by induction: the basis $\langle F^{\natural 0}, F^{\sharp 0}\rangle \in \sigma$ and induction step $\forall \lambda,\ \lambda' \in \mathit{Ord} : \langle F^{\natural\lambda},\ F^{\sharp\lambda'}\rangle \in \sigma \Rightarrow \exists \mu > \lambda,\ \mu' > \lambda' \in \mathit{Ord} :$

$\langle F^{\natural\mu},\ F^{\sharp\mu'}\rangle \in \sigma$ ensure $\langle F^{\natural\epsilon},\ F^{\sharp\varepsilon}\rangle \in \sigma$ for convergent iteration sequences. In general (4.5) ensures the existence of the $F^{\sharp\lambda}$. In particular when using semantic functions and inductive joins, we obtain:

Proposition 4.3 (Inductive soundness proof)
Given the set $\mathcal{P}^\natural$ of concrete properties, the set $\mathcal{P}^\sharp$ of abstract properties, the relation $\rho \in \wp(\mathcal{P}^\natural \times \mathcal{P}^\sharp)$,[2] the bases $\perp^\natural \in \mathcal{P}^\natural$ and $\perp^\sharp \in \mathcal{P}^\sharp$ such that $\langle \perp^\natural,\ \perp^\sharp\rangle \in \rho$, the concrete $F^\natural \in \mathcal{P}^\natural \mapsto \mathcal{P}^\natural$ and abstract $F^\sharp \in \mathcal{P}^\sharp \mapsto \mathcal{P}^\sharp$ semantic functions such that $\forall c \in \mathcal{P}^\natural : \forall a \in \mathcal{P}^\sharp : \langle c,\ a\rangle \in \rho \Rightarrow \langle F^\natural(c),\ F^\sharp(a)\rangle \in \rho$, the concrete $\coprod^\natural \in \wp(\mathcal{P}^\natural) \rightarrowtail \mathcal{P}^\natural$ and abstract $\coprod^\sharp \in \wp(\mathcal{P}^\sharp) \rightarrowtail \mathcal{P}^\sharp$ inductive joins such that $(\forall \beta < \lambda : \langle F^{\natural\beta},\ F^{\sharp\beta}\rangle \in \rho) \Rightarrow \langle \coprod^\natural_{\beta<\lambda} F^{\natural\beta},\ \coprod^\sharp_{\beta<\lambda} F^{\sharp\beta}\rangle \in \rho$ for all limit ordinals $\lambda > 0$[3] and assuming that the concrete and abstract iteration sequences are convergent[4] then their respective limits $F^{\natural\epsilon}$ and $F^{\sharp\varepsilon}$ are such that $\langle F^{\natural\epsilon},\ F^{\sharp\varepsilon}\rangle \in \rho$.

Proof. By transfinite induction on λ, we have $\langle F^{\natural\lambda},\ F^{\sharp\lambda}\rangle \in \rho$ for all $\lambda \in \mathit{Ord}$. Since both sequences are convergent, we have $F^{\natural\epsilon} = F^{\natural\mu}$, $F^{\sharp\varepsilon} = F^{\sharp\mu}$ and $\langle F^{\natural\mu},\ F^{\sharp\mu}\rangle \in \rho$ for the maximum μ of ϵ and ε. ■

This proposition has numerous corollaries, in particular when the limits are fixpoints of monotonic or continuous semantic functions on cpos or lattices.

Example 4.4 (Soundness proof of strictness analysis)
For strictness analysis, the basis $\langle \lambda\, x\cdot\perp^\partial,\ \lambda\, x\cdot 0\rangle \in \sigma$ is obvious. For the induction step we have to prove $\langle \varphi^\partial,\ \varphi^\sharp\rangle \in \sigma \Rightarrow \langle F^\partial(\varphi^\partial),\ F^\sharp(\varphi^\sharp)\rangle \in \sigma$ and proceed by induction on the syntax of programs using the equations defining F^∂ and $F^\sharp$. It remains to prove the last condition which, by continuity and termination of the abstract iteration, follows, for example, from the fact that if $\varphi^\partial_n, n \geq 0$ is an increasing chain for $\sqsubseteq^\partial$ such that $\forall n \geq 0 : \langle \varphi^\partial_n,\ \varphi^\sharp\rangle \in \sigma$ then $\langle \sqcup^\partial_{n\geq 0}\varphi^\partial_n,\ \varphi^\sharp\rangle \in \sigma$. This immediately follows from the remark that $\sqcup^\partial_{n\geq 0}\varphi^\partial_n(x) = \perp^\partial$ when $\varphi^\partial_n(x) = \perp^\partial$ for all $n \geq 0$.

Other examples of application are given in [1, 22, 29, 36].

Inducing the abstract semantics using extrapolation operators. When the soundness relation σ is one to many, the concrete iterates $F^{\natural\lambda}$, $\lambda \in \mathit{Ord}$ may have many abstract approximations $F^{\sharp\lambda}$, $\lambda \in \mathit{Ord}$ which are sound, that is $\forall \lambda \in \mathit{Ord} : \langle F^{\natural\lambda},\ F^{\sharp\lambda}\rangle \in \sigma$. In this case the soundness relation σ is not very helpful for inducing the abstract semantics from the concrete semantics since a discriminating choice must be done among all possibilities. To discriminate, we can use extrapolation operators:[5]

$$\begin{array}{ll} \triangledown^\sharp \in \wp(\mathcal{P}^\sharp) \rightarrowtail \mathcal{P}^\sharp & \quad \textit{widening} \\ \triangle^\sharp \in \wp(\mathcal{P}^\sharp) \rightarrowtail \mathcal{P}^\sharp & \quad \textit{narrowing} \end{array} \tag{4.6}$$

[2] Subsequently, ρ can be σ, α or γ^{-1}.

[3] For $\lambda = 0$ this condition subsumes $\langle \perp^\natural,\ \perp^\sharp\rangle \in \rho$ when $\coprod^\natural\emptyset = \perp^\natural$ and $\coprod^\sharp\emptyset = \perp^\sharp$.

[4] Convergent iterates have been defined as total, that is well-defined which ensures existence, and ultimately stationary, which ensures convergence.

[5] This is the first use of the widening and narrowing as sound choice functions (which are partial since they are needed in the abstract iteration only), the second one is to ensure convergence, the third is to guarantee rapid termination.

which preserve soundness so that for all $A \subseteq \mathcal{P}^\sharp$, we have:

$$\big(\triangledown^\sharp A \text{ exists}\big) \quad\Rightarrow\quad \forall c \in \mathcal{P}^\natural : (\exists a \in A : \langle c,\ a\rangle \in \sigma) \Rightarrow \big(\langle c,\ \triangledown^\sharp A\rangle \in \sigma\big) \tag{4.7}$$

$$\big(\vartriangle^\sharp A \text{ exists}\big) \quad\Rightarrow\quad \forall c \in \mathcal{P}^\natural : (\forall a \in A : \langle c,\ a\rangle \in \sigma) \Rightarrow \big(\langle c,\ \vartriangle^\sharp A\rangle \in \sigma\big) \tag{4.8}$$

Proposition 4.5 (Soundness of the induced iterates using extrapolation operators) Let $\mathcal{P}^\natural$ be the set of concrete properties, $\bot^\natural \in \mathcal{P}^\natural$ be the concrete basis, $F^\natural \in \mathcal{P}^\natural \rightarrowtail \mathcal{P}^\natural$ be the concrete semantic function and $\amalg^\natural \in \wp(\mathcal{P}^\natural) \rightarrowtail \mathcal{P}^\natural$ be the concrete inductive join such that the concrete iteration $F^{\natural\lambda}$, $\lambda \in \mathit{Ord}$ converges with limit $F^{\natural\epsilon}$. Let $\triangledown^\sharp$ be a widening and $\vartriangle^\sharp$ be a narrowing satisfying (4.6), (4.7) and (4.8). Define the abstract iteration $F^{\sharp\lambda}$, $\lambda \in \mathit{Ord}$ using:

$$\bot^\sharp \ \stackrel{\text{def}}{=}\ \alpha(\bot^\natural) \tag{4.9}$$

$$F^\sharp(a) \ \stackrel{\text{def}}{=}\ \triangledown^\sharp\Big\{\alpha\big(F^\natural(c)\big)\ \Big|\ c \in \mathcal{P}^\natural \wedge \langle c,\ a\rangle \in \sigma\Big\} \tag{4.10}$$

$$\coprod_{i\in I}^{\sharp} a_i \ \stackrel{\text{def}}{=}\ \triangledown^\sharp\Big\{\alpha\big(\coprod_{i\in I}^{\natural} c_i\big)\ \Big|\ \forall i \in I : c_i \in \mathcal{P}^\natural \wedge \langle c_i,\ a_i\rangle \in \sigma\Big\} \tag{4.11}$$

where $\alpha \in \mathcal{P}^\natural \rightarrowtail \mathcal{P}^\sharp$ is defined by $\alpha(c) \stackrel{\text{def}}{=} \vartriangle^\sharp \{a \,|\, \langle c,\ a\rangle \in \sigma\}$. If the abstract iteration is convergent with limit $F^{\sharp\varepsilon}$ then $\langle F^{\natural\epsilon},\ F^{\sharp\varepsilon}\rangle \in \sigma$.

Proof. By (4.8) we have $\langle c,\ \alpha(c)\rangle \in \sigma$ for all $c \in \mathcal{P}^\natural$ for which $\alpha(c)$ exists, whence in particular (4.9) implies $\langle \bot^\natural,\ \bot^\sharp\rangle \in \sigma$. Assume that $c \in \mathcal{P}^\natural$ and $a \in \mathcal{P}^\sharp$ satisfy $\langle c,\ a\rangle \in \sigma$. We have $\langle F^\natural(c),\ \alpha(F^\natural(c))\rangle \in \sigma$ so that by (4.7) and (4.10) it follows that $\langle F^\natural(c),\ F^\sharp(a)\rangle \in \sigma$. Assume that $\forall \beta < \lambda : \langle F^{\natural\beta},\ F^{\sharp\beta}\rangle \in \sigma$ where $\lambda > 0$ is a limit ordinal. We have $\langle \coprod_{\beta<\lambda}^{\natural} F^{\natural\beta},\ \alpha\big(\coprod_{\beta<\lambda}^{\natural} F^{\natural\beta}\big)\rangle \in \sigma$ so that by (4.7) and (4.11) we infer that $\langle \coprod_{\beta<\lambda}^{\natural} F^{\natural\beta},\ \coprod_{\beta<\lambda}^{\sharp} F^{\sharp\beta}\rangle \in \sigma$. We conclude using Proposition 4.3. ∎

Example 4.6 (The Galois connection framework)
In the classical framework of [7, 10, 12], the concrete properties $\langle \mathcal{P}^\natural;\ \sqsubseteq^\natural,\ \sqcup^\natural,\ \sqcap^\natural\rangle$ and abstract properties $\langle \mathcal{P}^\sharp;\ \sqsubseteq^\sharp,\ \sqcup^\sharp,\ \sqcap^\sharp\rangle$ are complete lattices. The correspondence between concrete and abstract properties is given by a Galois connection $\langle \mathcal{P}^\natural;\ \sqsubseteq^\natural\rangle \mathrel{\underset{\alpha}{\overset{\gamma}{\leftrightarrows}}} \langle \mathcal{P}^\sharp;\ \sqsubseteq^\sharp\rangle$ that is an abstraction map $\alpha \in \mathcal{P}^\natural \mapsto \mathcal{P}^\sharp$ and a concretization map $\gamma \in \mathcal{P}^\sharp \mapsto \mathcal{P}^\natural$ such that, by definition:[6]

$$\forall c \in \mathcal{P}^\natural : \forall a \in \mathcal{P}^\sharp : \alpha(c) \sqsubseteq^\sharp a \ \Leftrightarrow\ c \sqsubseteq^\natural \gamma(a)\ . \tag{4.12}$$

The soundness relation is $\langle c,\ a\rangle \in \sigma \stackrel{\text{def}}{=} \alpha(c) \sqsubseteq^\sharp a \Leftrightarrow c \sqsubseteq^\natural \gamma(a)$. Let us define $\triangledown^\sharp \stackrel{\text{def}}{=} \sqcup^\sharp$ and $\vartriangle^\sharp \stackrel{\text{def}}{=} \sqcap^\sharp$ so that, by definition of the greatest lower bound, for all $c \in \mathcal{P}^\natural$ we have $\vartriangle^\sharp \{a \,|\, \langle c,\ a\rangle \in \sigma\} = \sqcap^\sharp \big\{a \,|\, \alpha(c) \sqsubseteq^\sharp a\big\} = \alpha(c)$.

By (4.10) we have:

$$F^\sharp(a) \quad=\quad \sqcup^\sharp \big\{\alpha\big(F^\natural(c)\big) \,\big|\, \alpha(c) \sqsubseteq^\sharp a\big\} \tag{4.13}$$

[6] Observe that γ and α are monotonic since $\alpha(c') \preceq^\sharp \alpha(c') \ \Rightarrow\ c' \preceq^\natural \gamma(\alpha(c'))$ whence $c \preceq^\natural c' \ \Rightarrow\ c \preceq^\natural \gamma(\alpha(c')) \ \Rightarrow\ \alpha(c) \preceq^\sharp \alpha(c')$ and $\gamma(a) \preceq^\natural \gamma(a) \ \Rightarrow\ \alpha(\gamma(a)) \preceq^\sharp a$ whence $a \preceq^\sharp a' \ \Rightarrow\ \alpha(\gamma(a)) \preceq^\sharp a' \ \Rightarrow\ \gamma(a) \preceq^\natural \gamma(a')$ so that this requirement was redundant in definition 5.3.0.1 of [12].

For Galois connections, α is a complete join morphism (theorem 5.3.0.5.(4) of [12]), that is $\forall S \subseteq \mathcal{P}^\natural$: $\alpha(\sqcup^\natural S) = \sqcup^\sharp \alpha^\star(S)$ where $f^\star(S) \stackrel{\text{def}}{=} \{ f(x) \,|\, x \in S\}$ so that $F^\sharp(a) = \alpha(\sqcup^\natural \{ F^\natural(c) \,|\, \alpha(c) \sqsubseteq^\sharp a\})$. Moreover, α and $F^\natural$ are monotonic so that $\sqcup^\natural F^{\natural\star}(S) \sqsubseteq^\natural F^\natural(\sqcup^\natural S)$ whence $F^\sharp(a) \sqsubseteq^\sharp \alpha(F^\natural(\sqcup^\natural \{c \,|\, \alpha(c) \sqsubseteq^\sharp a\}))$. For Galois connections we have:[7]

$$\forall a \in \mathcal{P}^\sharp : \gamma(a) = \sqcup^\natural \{c \,|\, \alpha(c) \sqsubseteq^\sharp a\} \qquad \forall c \in \mathcal{P}^\natural : \alpha(c) = \sqcap^\sharp \{a \,|\, c \sqsubseteq^\natural \gamma(a)\} \tag{4.14}$$

(theorem of 5.3.0.5.(3) [12]) so that $F^\sharp(a) \sqsubseteq^\sharp \alpha(F^\natural(\gamma(a)))$. $F^\natural$ is often a complete join morphism, in which case:

$$F^\sharp \quad = \quad \alpha \circ F^\natural \circ \gamma \tag{4.15}$$

(where $\circ$ denotes the composition of functions) as originally proposed in paragraph 4.1.7 of [10]. Moreover, in [7, 10, 12], the inductive join $\amalg^\natural$ is the least upper bound $\sqcup^\natural$. Therefore (4.11) implies $\amalg^\sharp_{i \in I} a_i = \bigsqcup^\sharp \Big\{ \alpha\big(\sqcup^\natural_{i\in I} c_i\big) \;\Big|\; \forall i \in I : c_i \in \mathcal{P}^\natural \wedge \alpha(c_i) \sqsubseteq^\sharp a_i \Big\}$. Since α is a complete join morphism, this is equal to $\bigsqcup^\sharp \Big\{ \sqcup^\sharp_{i\in I} \alpha(c_i) \;\Big|\; \forall i \in I : c_i \in \mathcal{P}^\natural \wedge \alpha(c_i) \sqsubseteq^\sharp a_i \Big\} = \bigsqcup^\sharp_{i\in I} \sqcup^\sharp \{\alpha(c) \,|\, \alpha(c) \sqsubseteq^\sharp a_i\} = \bigsqcup^\sharp_{i\in I} \alpha(\sqcup^\natural \{c \,|\, \alpha(c) \sqsubseteq^\sharp a_i\})$ and therefore by (4.14) to $\alpha\big(\sqcup^\natural_{i\in I} \gamma(a_i)\big)$, that is:

$$\forall A \subseteq \mathcal{P}^\sharp : \sqcup^\sharp A \;=\; \alpha\big(\sqcup^\natural \gamma^\star(A)\big) \;. \tag{4.16}$$

We conclude that in the Galois connection framework Proposition 4.5 amounts to theorem 4.1.7.1 of [10] for lifting an abstract interpretation to higher-order functional spaces, using the fact that if $\langle \mathcal{P}^\natural_1;\ \sqsubseteq^\natural_1\rangle \overset{\gamma_1}{\underset{\alpha_1}{\leftrightarrows}} \langle \mathcal{P}^\sharp_1;\ \sqsubseteq^\sharp_1\rangle$ and $\langle \mathcal{P}^\natural_2;\ \sqsubseteq^\natural_2\rangle \overset{\gamma_2}{\underset{\alpha_2}{\leftrightarrows}} \langle \mathcal{P}^\sharp_2;\ \sqsubseteq^\sharp_2\rangle$ are Galois connections then:

$$\langle \mathcal{P}^\natural_1 \xmapsto{m} \mathcal{P}^\natural_2;\ \sqsubseteq^\natural\rangle \overset{\lambda\, \phi\bullet\gamma_2\circ\phi\circ\alpha_1}{\underset{\lambda\, \varphi\bullet\alpha_2\circ\varphi\circ\gamma_1}{\leftrightarrows}} \langle \mathcal{P}^\sharp_1 \xmapsto{m} \mathcal{P}^\sharp_2;\ \sqsubseteq^\sharp\rangle \tag{4.17}$$

is also a Galois connection for the pointwise orderings $\sqsubseteq^\natural$ and $\sqsubseteq^\sharp$ of monotonic maps.

Assessment of the soundness relation framework. This soundness relation framework is very general. It dispenses with a collecting semantics but then no sound and relatively complete method is offered for proving the abstract properties. We have shown that it is possible to infer the abstract semantics from the concrete semantics using widening and narrowing operators generalizing the extrapolation operators introduced in [6, 7]. The main weakness of that framework is that the relative precisions of abstract properties is not taken into account in the approximation of concrete properties.[8]

[7] The notion of adjoinedness is also studied in category theory and there the equation (4.14) introduced in [12] is said to express a *Kan extension*, as first observed by [1]. More generally, abstract interpretation could be rephrased in category theory [37].

[8] However, there is a natural notion of equivalence, since $c \equiv c' \stackrel{\text{def}}{=} (\exists a \in \mathcal{P}^\sharp : \langle c,\ a\rangle \in \sigma) \wedge (\forall a \in \mathcal{P}^\sharp : \langle c,\ a\rangle \in \sigma \Leftrightarrow \langle c',\ a\rangle \in \sigma)$ is a PER, which is an equivalence relation when (4.5) is satisfied. The development of an abstract interpretation framework based upon this PER is considered in paragraph 6.3 of [12] where the additional existence of a best abstract approximation hypothesis (4.25) leads to its further refinement as a complete join congruence relation. This should not be confused with the use of strict uniform inductive PERs by [25, 26] as elements of the concrete semantic domain $\mathcal{P}^\sharp$.

5 Approximation and computational orderings

The next step introduces a notion of precision in order to compare properties.

Modelling precision by a pre-order. In general, a program may satisfy many concrete properties c and each one may be approximated by many abstract properties a (according to the soundness condition $\langle c,\, a\rangle \in \sigma$). In order to distinguish the more precise ones, one can introduce a notion of *approximation* on the domain of concrete properties, on the domain of abstract properties or on both. In the absence of a metric distance specifying closeness of properties, one often indicates their relative precision using a pre-order relation $\preceq\, \in \wp(\mathcal{P} \times \mathcal{P})$ which is reflexive and transitive. $p \preceq p'$ means that "p is more precise than p'" or "p logically implies p'". $\preceq$ can be called the *approximation relation*. It may not be antisymmetric when the same property can be expressed by several elements of $\mathcal{P}$. One can take advantage of the existence of a normal form for such elements, by considering the quotient set $\mathcal{P}/_{\approx}$ for the equivalence relation $\approx$ defined by $p \approx p' \stackrel{\text{def}}{=} (p \preceq p') \wedge (p' \preceq p)$ which is tantamount to assuming that $\preceq$ is a partial order.

Example 5.1 (Relative precision of invariance properties)
In the case of invariance properties, this approximation relation would be set inclusion $\subseteq$ since I is approximated by J if and only if $I \subseteq J$ in the sense that any superset of the reachable states during execution represents a correct but approximate invariant. If, for example, the value of an integer variable is strictly positive during execution then it is sound, but less precise, to assert that it is positive or zero.

Observe that, in general, the *computational ordering*, that is the partial order $\sqsubseteq$ between the iterates, and the *approximation ordering* $\preceq$ are totally unrelated (although in [7], they happened to be the same, that is $\subseteq$).

The approximation ordering can be defined either on the abstract domain or on the concrete domain or on both. We now examine these alternatives in turn.

6 Abstraction correspondence between concrete and abstract semantics

We now examine a framework in which the notion of precision is formalized on the abstract properties using an approximation relation $\preceq^\sharp$:

$$\begin{array}{rcll} \preceq^\sharp & \in & \wp(\mathcal{P}^\sharp \times \mathcal{P}^\sharp) & \text{is a pre-order,} \\ a \approx^\sharp a' & \stackrel{\text{def}}{=} & (a \preceq^\sharp a') \wedge (a' \preceq^\sharp a) & \end{array} \tag{4.18}$$

and the connection between the concrete and abstract semantics is established via an abstraction relation or function α.

Soundness relation and abstract upper approximation. The objective of an abstract interpretation is to find an abstract property a, if any, in the abstract semantic domain $\mathcal{P}^\sharp$ which is a correct approximation of the concrete semantics $c \in \mathcal{P}^\natural$ of the program, i.e. such that $\langle c,\, a\rangle \in \sigma$. In general, an approximation $a' \in \mathcal{P}^\sharp$ such that $a \preceq^\sharp a'$ will also be a sound, although less precise, abstract property of

c. Since the approximation is sound, we should also have $\langle c,\, a'\rangle \in \sigma$. A natural *abstract soundness of upper approximations assumption*, for short *abstract soundness assumption* is therefore:[9]

$$\forall c \in \mathcal{P}^\natural : \forall a,\, a' \in \mathcal{P}^\sharp : (\langle c,\, a\rangle \in \sigma \,\wedge\, a \preceq^\sharp a') \;\Rightarrow\; \langle c,\, a'\rangle \in \sigma \ . \qquad (4.19)$$

Example 6.1 (Supremum)
In the context of upper approximations (4.19), a very simple way to ensure the existence of an abstract approximation for all concrete properties (4.5) is to assume that $\mathcal{P}^\sharp$ has a supremum $\top^\sharp$ for $\preceq^\sharp$ such that $\forall c \in \mathcal{P}^\natural : \langle c,\, \top^\sharp\rangle \in \sigma$. It follows that $\top^\sharp$ represents the absence of information on the concrete semantics c. If absent from $\mathcal{P}^\sharp$, such a supremum can be added to $\mathcal{P}^\sharp$. This can be justified for incomplete abstract interpreters by the existence of input programs for which the only analysis which can be done in a finite time is the output of a message indicating absence of information, which is formalized by the supremum $\top^\sharp$. For example in the case of invariance, this supremum would be the set of all possible states which provides no information about the run-time behaviours of a program. It has often been argued that such a supremum is not natural in the domains used for denotational semantics, and by an ill-considered generalization, this same claim has been made for the case of abstract interpretations. But recall that $\sqsubseteq^\natural$ and $\preceq^\sharp$ are, in general, unrelated and that the supremum $\top^\sharp$, corresponding to the "I don't know" answer, is only for $\preceq^\sharp$.

On the consequences of the abstract soundness assumption. Assume that the correctness of an abstract interpretation is specified using a soundness relation σ and an abstract approximation relation $\preceq^\sharp$ satisfying (4.3), (4.5), (4.18) and (4.19). Two different roles can be given to σ that is either to specify which are all possible abstract properties a that can be used to approximate a given concrete property c or else to specify the preferred ones. The second role is useful, for example, when $\preceq^\sharp$ is not antisymmetric. In this case all abstract properties in the equivalence class $[a]_{\approx^\sharp}$ are equally precise for approximating c and σ might select a unique representative in the class. Assumption (4.19) has the unfortunate consequence that this is not possible since all $a' \in [a]_{\approx^\sharp}$ must be such that $\langle c,\, a'\rangle \in \sigma$. Requiring $\preceq^\sharp$ to be antisymmetric would not help. For example, if we would like σ to be a bijection (so as, for example, to prove an equivalence between semantics in which case the abstract domain is just a different but equivalent representation of the concrete properties), then $\preceq^\sharp$ should be equality. In order to avoid these troubles we have to separate the two roles that may be given to σ. We think of σ as specifying which are all possible abstract properties which can be used to approximate a given concrete property and look for another way of specifying the preferred ones.

Abstraction relation between the concrete and abstract properties. Hence one can start by introducing an *abstraction relation*:

$$\alpha \in \wp(\mathcal{P}^\natural \times \mathcal{P}^\sharp) \qquad (4.20)$$

specifying which abstract properties can be used safely to represent concrete properties, together with a pre-order $\preceq^\sharp$ on $\mathcal{P}^\sharp$ to specify the relative precision of abstract

[9] As observed after definition 4.0.1 of [12]: "The dual one might be useful (e.g. for proving termination)" and this is the case for Mycroft termination analysis. However, following a well-established mathematical practice, we do not explicitely formulate dual results.

properties. Afterwards, the soundness relation $\sigma \in \wp(\mathcal{P}^\natural \times \mathcal{P}^\sharp)$ can be defined according to the following *abstraction based soundness assumption*:

$$\sigma = \{\langle c,\, a'\rangle \mid \exists a \in \mathcal{P}^\sharp : \langle c,\, a\rangle \in \alpha \ \wedge\ a \preceq^\sharp a'\} \ . \tag{4.21}$$

The relationship between $\preceq^\sharp$ and σ is that "less precise abstract properties approximate greater sets of concrete properties": for all a, a' in $\mathcal{P}^\sharp$, $a \preceq^\sharp a'$ implies $\{c \mid \langle c,\, a\rangle \in \sigma\} \subseteq \{c' \mid \langle c',\, a'\rangle \in \sigma\}$ by (4.21).

Minimal abstract approximations. A possible trouble is now that α may be equal to σ which is the case for example when α satisfies $\langle c,\, a\rangle \in \alpha \ \wedge\ a \preceq^\sharp a' \Rightarrow \langle c,\, a'\rangle \in \alpha$. To avoid this redundancy, the *soundness assumption*:

$$\alpha \subseteq \sigma \tag{4.22}$$

can sometimes be sharpened by requiring α to select minimal elements, that is the most precise ones. This consists in assuming the *abstract minimality assumption*:

$$\alpha = \{\langle c,\, a\rangle \in \sigma \mid \forall a' \in \mathcal{P}^\sharp : (\langle c,\, a'\rangle \in \sigma \wedge a' \preceq^\sharp a) \ \Rightarrow\ (a \preceq^\sharp a')\} \ . \tag{4.23}$$

The intent is that σ specifies the abstract properties which can be used to approximate a given concrete property, α specifies the preferred ones and $\preceq^\sharp$ specifies their relative precision. Hypotheses (4.21) and (4.23) are beneficial to dispense with one of σ or α. However, this is not always possible. For example in the case of a concrete property c which can be approximated by an infinite set $a_\lambda, \lambda < \delta$, $\omega \leq \delta \in \mathit{Ord}$ of abstract properties forming a strictly decreasing chain for $\preceq^\sharp$ with no lower bound, (4.23) would define α as the empty relation. In such a case, both σ and α are useful with α specifying an arbitrary choice in the infinite chain $a_\lambda, \lambda < \delta$.

On the uniqueness of the abstract interpretation. The most common situation however is that the set $\{a \mid \langle c,\, a\rangle \in \alpha\}$ of minimal abstract properties of the concrete semantics c of a program is not empty and has many elements. Therefore it may be preferable to consider the set $\{a \mid \langle c,\, a\rangle \in \alpha\}/_{\approx^\sharp}$ where equivalent abstract properties have been identified. But this set can also have many incomparable elements. Positiveness and evenness of the value of a variable are examples of incomparable invariance properties. This situation represents a lack of expressiveness of $\mathcal{P}^\sharp$ in that the best or more precise property of programs cannot be stated within the set of abstract properties $\mathcal{P}^\sharp$ (but can be in $\wp(\mathcal{P}^\sharp)$, which is a form of completion considered in paragraph 9.2 of [12], see Example 6.6 below). This is also somewhat impractical in that it either forces to switch to an abstract domain enriched by adding new abstract properties expressing the conjunction of properties in $\wp(\mathcal{P}^\sharp)$, which is equivalent to assuming uniqueness of the abstract approximation of any concrete property or to make an arbitrary choice among the possible properties of a program. We now examine these two alternatives, first exploring the advantages of the existence of a best approximation and then investigating what is to be done in absence of a best approximation.

Existence of a best abstract approximation. The "very reasonable assumption" of [12] consists in reasoning about $\mathcal{P}^\sharp/_{\approx^\sharp}$, which is equivalent to the assumption that $\preceq^\sharp$ is a partial order and in assuming the uniqueness of the approximation, more precisely that any concrete property c has at least one sound approximation in $\mathcal{P}^\sharp$ and that among these sound abstract properties, there is a best one. After [12], this is stated by the following *best abstract approximation assumption*:

$$\forall c \in \mathcal{P}^\natural : \big(\exists a \in \mathcal{P}^\sharp : \langle c,\, a\rangle \in \sigma\big) \;\Rightarrow\; \big(\forall a' \in \mathcal{P}^\sharp : \langle c,\, a'\rangle \in \sigma \;\Rightarrow\; a \preceq^\sharp a'\big)\ . \tag{4.24}$$

We frequently use the conjunction of the existence of abstract approximations assumption (4.5) and of the best abstract approximation assumption (4.24) which we call the *existence of a best abstract approximation assumption*:

$$\forall c \in \mathcal{P}^\natural : \exists a \in \mathcal{P}^\sharp : \langle c,\, a\rangle \in \sigma \;\wedge\; \forall a' \in \mathcal{P}^\sharp : \langle c,\, a'\rangle \in \sigma \;\Rightarrow\; a \preceq^\sharp a'\ . \tag{4.25}$$

In this case α is a function such that:

$$\forall c \in \mathcal{P}^\natural : \forall a \in \mathcal{P}^\sharp : \langle c,\, a\rangle \in \sigma \;\Leftrightarrow\; \alpha(c) \preceq^\sharp a\ . \tag{4.26}$$

Proposition 6.2 (Existence of an abstraction function)
1° Assume that $\preceq^\sharp$ is a partial order relation on $\mathcal{P}^\sharp$ and that $\sigma \in \wp(\mathcal{P}^\natural \times \mathcal{P}^\sharp)$ satisfies the existence of a best abstract approximation assumption (4.25). If $\alpha \in \wp(\mathcal{P}^\natural \times \mathcal{P}^\sharp)$ is defined by (4.23) then $\alpha \in \mathcal{P}^\natural \mapsto \mathcal{P}^\sharp$.
2° If, moreover, σ satisfies the soundness assumption (4.19) then (4.26) holds.

Proof. 1° For all $c \in \mathcal{P}^\natural$, there exists $a \in \mathcal{P}^\sharp$ satisfying (4.25) which, by definition of α, implies that $\langle c,\, a\rangle \in \alpha$ proving that α is total. Assume that $\langle c,\, a_1\rangle \in \alpha$ and $\langle c,\, a_2\rangle \in \alpha$. Then, by definition of α, $\langle c,\, a_1\rangle \in \sigma$ and $\langle c,\, a_2\rangle \in \sigma$ so that by (4.25) there exists $a \in \mathcal{P}^\sharp$ such that $a \preceq^\sharp a_1$ and $a \preceq^\sharp a_2$. By definition of α we infer that $a_1 \preceq^\sharp a$ and $a_2 \preceq^\sharp a$. By antisymmetry and transitivity, we conclude that $a_1 = a_2$ thus proving that α is a (total) function.
2° We have seen that for all $c \in \mathcal{P}^\natural$, $\alpha(c)$ is the unique a satisfying (4.25), hence $\forall a' \in \mathcal{P}^\sharp : \langle c,\, a'\rangle \in \sigma \Rightarrow \alpha(c) \preceq^\sharp a'$. Reciprocally, if $\alpha(c) \preceq^\sharp a'$ then $\langle c,\, \alpha(c)\rangle \in \sigma$, by definition of α, hence $\langle c,\, a'\rangle \in \sigma$ according to the soundness assumption (4.19). ■

In the absence of a best abstract approximation. Not all abstract interpretations satisfy the existence of a best abstract approximation assumption. There are essentially two reasons why the existence of a best approximation fails. One is that some concrete property c may have no abstract approximation, which is excluded by (4.5). The other one is that the set of abstract approximations $\{a \mid \langle c,\, a\rangle \in \sigma\}$ of some concrete property c may be an infinite strictly decreasing chain for $\preceq^\sharp$ or a set of finite or infinite non-comparable strictly decreasing chains.

Example 6.3 (Convex polyhedra)
[17] consider the approximation of vectors of reals, that is subsets S of $\mathbb{R}^n$, by a convex polyhedron P such that $S \subseteq P$. If S is finite then the best P is the convex hull of S, but a sphere for example, has no best upper approximation by a convex polyhedron.

In the absence of a best approximation, several solutions are available:

Weakening the abstract properties. If one insists upon having a best abstract upper approximation of concrete properties, one can choose weaker abstract properties, that is a smaller set of choices.

Example 6.4 (Intervals)
Instead of convex polyhedra, one can choose cubes whose faces are parallel to the axes as in [6], in which case any subset S of $\mathbb{R}^n$ has a best upper approximation. However, with this approach, the results of the analysis (intervals of values in the example) are much less precise than the original (conjunctions of linear inequalities in the example).

Example 6.5 (Depth k abstraction of success patterns)
[38] have introduced an abstract interpretation for logic programs where expressions (clauses, atoms or terms) e are approximated by replacing every level k subexpressions of e by a newly created variable. [32] and [33] have established that there is no best approximation. For example the depth 3 abstraction of $e = p(g(f(u), f(f(v))), f(f(v)))$ is $e_1 = p(g(f(u), f(x)), f(f(y)))$ but $e_2 = p(g(f(u), f(z)), f(z))$ would be just as good a choice. The two approximations e_1 and e_2 are sound since e is an instance of both of them but they are not comparable since neither is an instance of the other. [32] have shown the existence of a best abstract upper approximation by weakening [38] depth k abstraction so that no variable occurs more than once (including at depth less than k).

Strengthening the abstract properties. If one insists upon having a best abstract upper approximation of concrete properties, one can also chose stronger abstract properties that is a larger set of choices. This approach was considered in paragraphs 5.2 and 9.2 of [12].

Example 6.6 (Conjunctive completion)
Given the set $\mathcal{P}^\sharp$ of abstract properties, one can move to the set $\overline{\mathcal{P}}^\sharp = \{\overline{a} \subseteq \mathcal{P}^\sharp | \, \forall a, a' \in \overline{a} : \neg(a \preceq^\sharp a')\}$ of subsets of incomparable properties of $\mathcal{P}^\sharp$. The interpretation of $\overline{a}$ is the *conjunction* of the interpretations of the original $a \in \overline{a}$. Therefore the soundness relation $\overline{\sigma}$ is now $\langle c,\, \overline{a}\rangle \in \overline{\sigma}$ if and only if $\forall a \in \overline{a} : \langle c,\, a\rangle \in \sigma$. The approximation relation $\overline{\preceq}^\sharp$ is $\overline{a} \mathrel{\overline{\preceq}^\sharp} \overline{a}'$ if and only if $\forall a' \in \overline{a}' : \exists a \in \overline{a} : a \preceq^\sharp a'$. $\overline{\preceq}^\sharp$ is a pre-order. If σ and $\preceq^\sharp$ satisfy any one of the existence of upper approximations (4.5), abstract soundness (4.19) assumptions then so do $\overline{\sigma}$ and $\overline{\preceq}^\sharp$. If we define $\overline{\alpha} \in \mathcal{P}^\natural \mapsto \overline{\mathcal{P}}^\sharp$ by $\overline{\alpha}(c) \stackrel{\text{def}}{=} \{a \,|\, \langle c,\, a\rangle \in \sigma\}$ then it satisfies the abstract minimality assumption (4.23) and the existence of a best abstract approximation assumption (4.25): $\forall c \in \mathcal{P}^\natural : \langle c,\, \overline{\alpha}(c)\rangle \in \overline{\sigma} \;\wedge\; \forall \overline{a}' \in \overline{\mathcal{P}^\sharp} : \langle c,\, \overline{a}'\rangle \in \overline{\sigma} \;\Rightarrow\; \overline{\alpha}(c) \mathrel{\overline{\preceq}^\sharp} \overline{a}'$. This construction generalizes the use of Moore families in theorem 5.2.0.4 of [12]. It is used in [33] to strengthen the depth k abstraction of [38].

This conjunctive completion can introduce a combinatorial explosion since all original possibilities have to be explored. For our convex polyhedra example, the sphere would be approximated by infinitely many incomparable convex polyhedra and there are infinitely many ways to do so. An implementation based upon backtracking would not be much better since an efficient choice function would not be easy to design. Using convex sets instead of convex polyhedra would not be more helpful in that they are not machine representable.

Example 6.7 (Disjunctive completion)
Given the set $\mathcal{P}^\sharp$ of abstract properties, one can move to the set $\widehat{\mathcal{P}^\sharp} = \{\widehat{a} \subseteq \mathcal{P}^\sharp | \; \forall a, a' \in \widehat{a} : \neg(a \preceq^\sharp a')\}$ of subsets of incomparable properties of $\mathcal{P}^\sharp$. The interpretation of $\widehat{a}$ is the *disjunction* of the interpretations of the original $a \in \widehat{a}$. Therefore the soundness relation $\widehat{\sigma}$ is now $\langle c,\; \widehat{a}\rangle \in \widehat{\sigma}$ if and only if $\exists a \in \widehat{a} : \langle c,\; a\rangle \in \sigma$. The approximation relation $\widehat{\preceq}^\sharp$ is $\widehat{a} \; \widehat{\preceq}^\sharp \; \widehat{a}'$ if and only if $\forall a \in \widehat{a} : \exists a' \in \widehat{a}' : a \preceq^\sharp a'$. $\widehat{\preceq}^\sharp$ is a pre-order.

This disjunctive completion generalizes paragraph 9.2 of [12]. As noted there, it can introduce a combinatorial explosion. Even worse, $\widehat{\mathcal{P}^\sharp}$ might not be machine representable whereas $\mathcal{P}^\sharp$ was. For example, for constant propagation $\mathcal{P}^\sharp = \mathbb{Z} \cup \{\bot, \top\}$ is an infinite lattice satisfying the ascending chain condition whereas $\widehat{\mathcal{P}^\sharp} = \wp(\mathbb{Z})$ does not satisfy this condition for $\subseteq$.

One can also strengthen the abstract properties by complementation and more generally by considering various completion methods.

Discriminating between the abstract properties. Given the soundness relation σ and the problem of designing the abstract semantics $\langle \pm^\sharp,\; F^\sharp,\; \amalg^\sharp\rangle$ starting from the concrete semantics $\langle \pm^\natural,\; F^\natural,\; \amalg^\natural\rangle$, one can abandon the idea of having a best abstract upper approximation of concrete properties c by making an arbitrary choice $\alpha(c)$ among all sound possibilities $\{a \,|\, \langle c,\; a\rangle \in \sigma\}$, or better among the minimal ones, as suggested in Proposition 4.5 using a widening $\triangledown^\sharp$. In the design of the abstract iteration $F^{\sharp\lambda}$, $\lambda \in \mathit{Ord}$, the discretionary choice of the abstract approximation can be made once for all, either:

- "statically", independently of the behaviour of the abstract iteration, in that the often implicit widening $\triangledown^\sharp$ is integrated in the abstract semantic function $F^\sharp$ (4.10); or
- "dynamically", depending at each iteration step upon the previous abstract iterates, in that the explicit widening $\triangledown^\sharp$ is made part of the iteration process.

We first consider refinements of Proposition 4.5 where the widening is hidden and then where it is explicit.

On the inducing of the concrete iterates into the abstract domain using an abstraction function. Proposition 4.3 can be applied with the abstraction function to get the following corollary (generalizing proposition 7.1.0.4.(3) of [12]):

Proposition 6.8 (Semantic inducing using an abstraction function)
Given the set $\mathcal{P}^\natural$ of concrete properties, the set $\mathcal{P}^\sharp$ of abstract properties, the abstraction function $\alpha \in \mathcal{P}^\natural \mapsto \mathcal{P}^\sharp$, the basis $\pm^\sharp \stackrel{\text{def}}{=} \alpha(\pm^\natural)$, the concrete $F^\natural \in \mathcal{P}^\natural \mapsto \mathcal{P}^\natural$ and abstract $F^\sharp \in \mathcal{P}^\sharp \mapsto \mathcal{P}^\sharp$ semantic functions such that $\alpha \circ F^\natural = F^\sharp \circ \alpha$, the concrete $\amalg^\natural \in \wp(\mathcal{P}^\natural) \rightarrowtail \mathcal{P}^\natural$ and abstract $\amalg^\sharp \in \wp(\mathcal{P}^\sharp) \rightarrowtail \mathcal{P}^\sharp$ inductive joins such that $\alpha(\amalg^\natural_{\beta<\lambda} F^{\natural\beta}) = \amalg^\sharp_{\beta<\lambda} \alpha(F^{\natural\beta})$ for all limit ordinals $\lambda > 0$ and assuming that the concrete and abstract iteration sequences are convergent then their respective limits $F^{\natural\epsilon}$ and $F^{\sharp\varepsilon}$ are such that $F^{\sharp\varepsilon} = \alpha(F^{\natural\epsilon})$.

Proof. Use Proposition 4.3 with $\rho \stackrel{\text{def}}{=} \{\langle c,\; \alpha(c)\rangle \,|\, c \in \mathcal{P}^\natural\}$. ∎

Observe that in this case the abstraction process consists in exhibiting an underlying structure of the concrete properties $\mathcal{P}^\natural$ which can be induced without loss of information, through the abstraction function α, into the set of abstract properties $\mathcal{P}^\sharp$. A typical example consists in deriving an abstract semantics from a concrete one, as illustrated by [19, theorem 10-4] to relate transition systems and predicate transformers; by [2, fact 2.3] to relate denotational semantics or by [20, lemma 4.3] to relate infinite computations and denotational semantics.

We habitually use this proposition constructively in order to derive the abstract semantics from the definition of the concrete semantics: for the basis we simply let $\pm^\sharp$ be $\alpha(\pm^\natural)$. For the semantic function $F^\sharp$, starting from the term $\alpha(F^\natural(c))$, we replace α and $F^\natural$ by their definitions and then simplify the expression in order to let the term $\alpha(c)$ come out, in which case we let the resulting expression (where $\alpha(c)$ is replaced by a) be the definition of $F^\sharp(a)$. The same way, for the inductive join, starting from $\alpha(\amalg^\natural_{\beta<\lambda} c^\beta)$, we derive $\amalg^\sharp$. This will be illustrated in Example 6.11 by the derivation of a strictness analysis algorithm from a denotational semantics. Such a formal computation might be, hopefully, mechanizable since exactly the same proof strategy, with similar substitutions and simplifications, can be used to derive many different abstract interpretations. The knowledge of this proof strategy would be a great help for guiding mechanized proofs.

Depending on how the relation between the concrete and abstract properties is defined, Propositions 4.3 and 6.8 have numerous variants. When a concretization function γ and an abstraction function α are available such that no information is lost in the abstract interpretation ($\forall c \in \mathcal{P}^\natural : \gamma(\alpha(c)) = c$) we can use the following (cf. theorem 7.1.0.4-3 of [12]):

Proposition 6.9 (Semantic inducing from a cpo using a pair of abstraction and concretization functions)
1°) Let $F^\natural \in \mathcal{P}^\natural \xmapsto{m} \mathcal{P}^\natural$ be a total monotonic function on the cpo $\langle \mathcal{P}^\natural;\ \sqsubseteq^\natural,\ \bot^\natural,\ \sqcup^\natural\rangle$. Let $\alpha \in \mathcal{P}^\natural \mapsto \mathcal{P}^\sharp$ and $\gamma \in \mathcal{P}^\sharp \mapsto \mathcal{P}^\natural$ be total maps such that $\forall c \in \mathcal{P}^\natural : \gamma(\alpha(c)) = c$ and the basis $\pm^\natural \in \mathcal{P}^\natural$ be such that $\pm^\natural \sqsubseteq^\natural F^\natural(\pm^\natural)$.[10] Define $\pm^\sharp \overset{\text{def}}{=} \alpha(\pm^\natural)$, $F^\sharp \in \mathcal{P}^\sharp \mapsto \mathcal{P}^\sharp$ as $F^\sharp \overset{\text{def}}{=} \alpha \circ F^\natural \circ \gamma$, $\sqcup^\sharp \in \wp(\mathcal{P}^\sharp) \rightarrowtail \mathcal{P}^\sharp$ by $\sqcup^\sharp X \overset{\text{def}}{=} \alpha(\sqcup^\natural \gamma^\star(X))$. Then the abstract iteration sequence $F^{\sharp 0} \overset{\text{def}}{=} \pm^\sharp$, $F^{\sharp\lambda+1} \overset{\text{def}}{=} F^\sharp(F^{\sharp\lambda})$ for $\lambda \in \mathit{Ord}$ and $F^{\sharp\lambda} \overset{\text{def}}{=} \sqcup^\sharp_{\beta<\lambda} F^{\sharp\beta}$ for all limit ordinals $\lambda > 0$ is increasing for $\sqsubseteq^\sharp \overset{\text{def}}{=} \big\{\langle x,\ y\rangle \,\big|\, \gamma(x) \sqsubseteq^\natural \gamma(y)\big\}$ and ultimately stationary. Its limit $F^{\sharp\varepsilon}$ is $\alpha(F^{\natural\epsilon})$ with $\varepsilon \leq \epsilon$ and the limit of the concrete iterates is $F^{\natural\epsilon} = \mathit{lfp}^{\sqsubseteq^\natural}_{\pm^\natural} F^\natural$ (where $\mathit{lfp}^{\sqsubseteq}_{x} F$ is the least fixpoint of F greater than or equal to x for $\sqsubseteq$).
2°) If x is a fixpoint of $F^\natural$ then $\alpha(x)$ is a fixpoint of $F^\sharp$.
3°) α is monotonic with respect to $\sqsubseteq^\natural$ and $\sqsubseteq^\sharp$. Moreover, $F^{\sharp\varepsilon} = \mathit{lfp}^{\sqsubseteq^\sharp}_{\pm^\sharp} F^\sharp$.

Proof. 1°) Let $F^{\natural\lambda}$, $\lambda \in \mathit{Ord}$ be the concrete iteration sequence. Since $\pm^\natural$ is a pre-fixpoint of the monotonic total function $F^\natural$, it is increasing whence ultimately stationary with limit $F^{\natural\epsilon} = \mathit{lfp}^{\sqsubseteq^\natural}_{\pm^\natural} F^\natural$. We have $\alpha(F^{\natural 0}) = \alpha(\pm^\natural) = \pm^\sharp = F^{\sharp 0}$. If $F^{\sharp\lambda}$ $= \alpha(F^{\natural\lambda})$ then $F^{\sharp\lambda+1} = F^\sharp(F^{\sharp\lambda}) = \alpha \circ F^\natural \circ \gamma \circ \alpha(F^{\natural\lambda}) = \alpha(F^\natural(F^{\natural\lambda})) = \alpha(F^{\natural\lambda+1})$. If $F^{\sharp\beta} = \alpha(F^{\natural\beta})$ for all $\beta < \lambda$ then $\sqcup^\sharp_{\beta<\lambda} F^{\sharp\beta} = \alpha\Big(\sqcup^\natural_{\beta<\lambda}\gamma\big(F^{\sharp\beta}\big)\Big) = \alpha\Big(\sqcup^\natural_{\beta<\lambda}\gamma\big(\alpha(F^{\natural\beta})\big)\Big)$

[10] This condition is obviously satisfied when $\pm^\natural \overset{\text{def}}{=} \bot^\natural$.

$= \alpha\Big(\sqcup^\natural_{\beta<\lambda} F^{\natural\beta}\Big) = \alpha(F^{\natural\lambda+1})$. We conclude by transfinite induction that $\forall\lambda \in \mathit{Ord} :$ $F^{\sharp\lambda} = \alpha(F^{\natural\lambda})$.

2°) If x is a fixpoint of $F^\natural$ then $F^\sharp(\alpha(x)) = \alpha \circ F^\natural \circ \gamma \circ \alpha(x) = \alpha \circ F^\natural(x) = \alpha(x)$.

3°) Obviously α is monotonic since $c \sqsubseteq^\natural c'$ implies $\gamma(\alpha(c)) \sqsubseteq^\natural \gamma(\alpha(c'))$ whence $\alpha(c) \sqsubseteq^\sharp \alpha(c')$. Let $x \in \mathcal{P}^\sharp$ be a fixpoint of $F^\sharp$ such that $F^{\sharp 0} = \underline{\perp}^\sharp \sqsubseteq^\sharp x$. Obviously γ is monotonic by definition of $\sqsubseteq^\sharp$. It follows that $F^\sharp$ is monotonic since it is the composition of monotonic functions. Therefore if $F^{\sharp\lambda} \sqsubseteq^\sharp x$ then $F^{\sharp\lambda+1} = F^\sharp(F^{\sharp\lambda})$ $\sqsubseteq^\sharp F^\sharp(x) = x$. If $F^{\sharp\beta}$, $\beta < \lambda$ is an increasing chain for $\sqsubseteq^\sharp$ and $F^{\sharp\beta} \sqsubseteq^\sharp x$ for all $\beta < \lambda$, then $\gamma(F^{\sharp\beta})$, $\beta < \lambda$ is an increasing chain for $\sqsubseteq^\natural$ such that $\gamma(F^{\sharp\beta}) \sqsubseteq^\natural \gamma(x)$ for all $\beta < \lambda$ hence $\sqcup^\natural_{\beta<\lambda}\gamma(F^{\sharp\beta}) \sqsubseteq^\natural \gamma(x)$ by definition of well-defined least upper bounds so that $\gamma \circ \alpha(\sqcup^\natural_{\beta<\lambda}\gamma(F^{\sharp\beta})) \sqsubseteq^\natural \gamma(x)$ since $\gamma \circ \alpha$ is identity whence $\alpha(\sqcup^\natural_{\beta<\lambda}\gamma(F^{\sharp\beta})) \sqsubseteq^\sharp x$ by definition of $\sqsubseteq^\sharp$ so that $\sqcup^\sharp_{\beta<\lambda}F^{\sharp\beta} \sqsubseteq^\sharp x$ by definition of $\sqcup^\sharp$. By transfinite induction, for all $\lambda \in \mathit{Ord}$, $F^{\sharp\lambda} \sqsubseteq^\sharp x$, proving for $\lambda = \varepsilon$, that the limit of the iterates of $F^\sharp$ is less than or equal to x. ∎

On the inducing of the approximate abstract iterates using an abstraction function. Since the expressions involved in the formal computation for deriving $F^\sharp$ from $F^\natural$ and α are sometimes intricate or uncomputable, the approximation relation $\preceq^\sharp$ can be used to make further simplifications using abstract properties which are approximations of concrete properties. Given $\alpha \in \mathcal{P}^\natural \mapsto \mathcal{P}^\sharp$, the soundness relation is expressed according to (4.21) by $\langle c,\ a\rangle \in \sigma \overset{\text{def}}{=} \alpha(c) \preceq^\sharp a$ (so that the abstract minimality assumption (4.23) is obviously satisfied). The general idea is that in Proposition 4.5, the widening $\nabla^\sharp$ can be replaced by a *coarser* one $\overline{\overline{\nabla}}^\sharp \in \wp(\mathcal{P}^\sharp) \rightarrowtail \mathcal{P}^\sharp$ such that if $\nabla^\sharp A$ exists then $\overline{\overline{\nabla}}^\sharp A$ exists and $\nabla^\sharp A \preceq^\sharp \overline{\overline{\nabla}}^\sharp A$ in which case $\langle c,\ a\rangle \in \sigma$ implies $\langle c,\ \nabla^\sharp A\rangle \in \sigma$ by (4.7) whence $\alpha(c) \preceq^\sharp \nabla^\sharp A$ by (4.21) and therefore $\alpha(c) \preceq^\sharp$ $\overline{\overline{\nabla}}^\sharp A$ by transitivity, that is $\langle c,\ \overline{\overline{\nabla}}^\sharp A\rangle \in \sigma$. In the absence of an explicit widening, the soundness of this approach can be justified by the following:

Proposition 6.10 (Semantic approximation using an abstraction function)
Given the sets $\mathcal{P}^\natural$ of concrete properties and $\langle\mathcal{P}^\sharp;\ \preceq^\sharp\rangle$ of abstract properties with pre-order $\preceq^\sharp$, the abstraction function $\alpha \in \mathcal{P}^\natural \mapsto \mathcal{P}^\sharp$, the basis $\underline{\perp}^\natural$ such that $\alpha(\underline{\perp}^\natural)$ $\preceq^\sharp F^{\sharp 0}$, the concrete semantic function $F^\natural \in \mathcal{P}^\natural \rightarrowtail \mathcal{P}^\natural$ such that $\alpha(F^{\natural\lambda}) \preceq^\sharp F^{\sharp\lambda} \Rightarrow$ $\alpha(F^\natural(F^{\natural\lambda})) \preceq^\sharp F^{\sharp\lambda+1}$ for all $\lambda \in \mathit{Ord}$, the concrete inductive join $\amalg^\natural \in \wp(\mathcal{P}^\natural) \rightarrowtail \mathcal{P}^\natural$ such that $\forall\beta < \lambda : \alpha(F^{\natural\beta}) \preceq^\sharp F^{\sharp\beta} \Rightarrow \alpha(\amalg^\natural_{\beta<\lambda}F^{\natural\beta}) \preceq^\sharp F^{\sharp\lambda}$ for all limit ordinals $\lambda > 0$ and assuming that the concrete and abstract iteration sequences are convergent then their respective limits $F^{\natural\epsilon}$ and $F^{\sharp\varepsilon}$ are such that $\alpha(F^{\natural\epsilon}) \preceq^\sharp F^{\sharp\varepsilon}$.

Proof. By transfinite induction $\alpha(F^{\natural\lambda}) \preceq^\sharp F^{\sharp\lambda}$ for all $\lambda \in \mathit{Ord}$. ∎

Observe that in this case the abstraction iteration process omits certain details or aspects of the concrete properties which causes some loss of information.

Example 6.11 (Inducing Burn, Hankin and Abramsky's strictness analysis method from a denotational semantics)
Let us illustrate the inducing of Burn, Hankin and Abramsky's strictness analysis method [3] from a denotational semantics for the simple explicitly typed lambda calculus considered in [1].

Syntax. Given base types $A, \ldots$ including booleans $\mathbb{B} = \{\mathrm{tt}, \mathrm{ff}\}$, type expressions have the form $\tau = A \mid \tau_2 \to \tau_1$.

Typed constants c_τ such as a fixpoint combinator $\mathit{fix}_{(\tau\to\tau)\to\tau}$ and a conditional $\mathit{cond}_{\mathbb{B}\to\tau\to\tau}$ as well as typed variables x_τ are given for each type τ. Terms e of type τ are built as follows:

$$x_\tau : \tau \qquad\quad c_\tau : \tau \qquad\quad \frac{e : \tau_1}{\lambda\, x_{\tau_2} \centerdot e : \tau_2 \to \tau_1} \qquad\quad \frac{e_1 : \tau_2 \to \tau_1 \qquad e_2 : \tau_2}{e_1 e_2 : \tau_1} \ . \tag{4.27}$$

Denotational semantics. An interpretation is given by bounded complete domains [24] D_τ with infimum $\bot_\tau$ for each type τ such that $D_{\tau_2\to\tau_1}$ is the domain $D_{\tau_2} \overset{c}{\longmapsto} D_{\tau_1}$ of continuous functions with domain D_{τ_2} and codomain D_{τ_1} and values $\underline{c}_\tau \in D_\tau$ for each constant c_τ such that, in particular:

$$\underline{\mathit{fix}}(\varphi) \stackrel{\text{def}}{=} \bigsqcup_{n\geq 0} \varphi^n(\bot_\tau) \qquad\qquad \begin{array}{rcl} \underline{\mathit{cond}}\,\bot_{\mathbb{B}}\, x\, y & \stackrel{\text{def}}{=} & \bot_\tau \\ \underline{\mathit{cond}}\ \mathrm{tt}\ x\, y & \stackrel{\text{def}}{=} & x \\ \underline{\mathit{cond}}\ \mathrm{ff}\ x\, y & \stackrel{\text{def}}{=} & y \ . \end{array} \tag{4.28}$$

Environments $\rho \in \mathcal{E}$ map variables x_τ to values $\rho(x_\tau) \in D_\tau$. $\rho[x \to \nu]$ is the environment mapping x to ν and everywhere else equal to ρ.

The concrete properties are $\mathcal{P}^\natural \stackrel{\text{def}}{=} \mathcal{E} \mapsto \mathcal{D}$ where $\mathcal{D} \stackrel{\text{def}}{=} \bigcup_\tau D_\tau$. The concrete semantics $[\![e_\tau]\!]^\natural \rho \in \mathcal{E} \mapsto \mathcal{D}_\tau$ of term e_τ in environment ρ is defined by:

$$\begin{array}{rcl@{\qquad\qquad}rcl} [\![x_\tau]\!]^\natural\rho & \stackrel{\text{def}}{=} & \rho(x_\tau) & [\![\lambda\, x_\tau \centerdot e]\!]^\natural\rho & \stackrel{\text{def}}{=} & \lambda\, \nu \in D_\tau \centerdot [\![e]\!]^\natural\ \rho[x_\tau \to \nu] \\ [\![c_\tau]\!]^\natural\rho & \stackrel{\text{def}}{=} & \underline{c}_\tau & [\![e_1 e_2]\!]^\natural\rho & \stackrel{\text{def}}{=} & ([\![e_1]\!]^\natural\rho)([\![e_2]\!]^\natural\rho) \ . \end{array} \tag{4.29}$$

Abstraction. Following Mycroft, the abstract domain $\langle D^\sharp_A;\ \sqsubseteq^\sharp_A,\ 0,\ \sqcup^\sharp_A\rangle$ for base types A is a complete lattice where $D^\sharp_A = \{0, 1\}$ and $0 \sqsubseteq^\sharp_A 0 \sqsubseteq^\sharp_A 1 \sqsubseteq^\sharp_A 1$. The abstraction function $\alpha_A \in D_A \mapsto D^\sharp_A$ is:

$$\alpha_A(\bot_A) \stackrel{\text{def}}{=} 0 \ \text{ and } \ \alpha_A(\nu) \stackrel{\text{def}}{=} 1 \text{ when } \nu \in D_A - \{\bot_A\} \ . \tag{4.30}$$

Given complete lattices $\langle D^\sharp_{\tau_2};\ \sqsubseteq^\sharp_{\tau_2},\ \bot^\sharp_{\tau_2},\ \sqcup^\sharp_{\tau_2}\rangle$ and $\langle D^\sharp_{\tau_1};\ \sqsubseteq^\sharp_{\tau_1},\ \bot^\sharp_{\tau_1},\ \sqcup^\sharp_{\tau_1}\rangle$ then $\langle D^\sharp_{\tau_2\to\tau_1};\ \sqsubseteq^\sharp_{\tau_2\to\tau_1},\ \bot^\sharp_{\tau_2\to\tau_1},\ \sqcup^\sharp_{\tau_2\to\tau_1}\rangle$ is a complete lattice where $D^\sharp_{\tau_2\to\tau_1} \stackrel{\text{def}}{=} D^\sharp_{\tau_2} \overset{c}{\longmapsto} D^\sharp_{\tau_1}$ and $\sqsubseteq^\sharp_{\tau_2\to\tau_1}$ is the pointwise ordering. The corresponding abstraction function $\alpha_{\tau_2\to\tau_1} \in (D_{\tau_2\to\tau_1}) \mapsto (D^\sharp_{\tau_2\to\tau_1})$ is defined, according to (4.13), as:

$$\alpha_{\tau_2\to\tau_1}(\varphi) \ \stackrel{\text{def}}{=} \ \lambda\, a \centerdot \sqcup^\sharp_{\tau_1} \left\{\alpha_{\tau_1}(\varphi(\nu)) \,\middle|\, \alpha_{\tau_2}(\nu) \sqsubseteq^\sharp_{\tau_2} a\right\} \tag{4.31}$$

where $\alpha_{\tau_2} \in D_{\tau_2} \mapsto D^\sharp_{\tau_2}$ and $\alpha_{\tau_1} \in D_{\tau_1} \mapsto D^\sharp_{\tau_1}$. Observe that α_τ is surjective.

Abstract environments $\rho^\sharp \in \mathcal{E}^\sharp$ map variables x_τ to abstract values $\rho^\sharp(x_\tau) \in D^\sharp_\tau$ and define $\alpha_{\mathcal{E}} \in \mathcal{E} \mapsto \mathcal{E}^\sharp$ such that $\alpha_{\mathcal{E}}(\rho)x_\tau \stackrel{\text{def}}{=} \alpha_\tau(\rho(x_\tau))$.

The abstract properties are $\mathcal{P}^\sharp \stackrel{\text{def}}{=} \mathcal{E}^\sharp \mapsto \mathcal{D}^\sharp$ where $\mathcal{D}^\sharp \stackrel{\text{def}}{=} \bigcup_\tau D^\sharp_\tau$.

Soundness. This abstract interpretation is sound since if $\alpha_{A\to A}(\varphi) \sqsubseteq^\sharp \phi$ and $\phi(0) = 0$ then $\sqcup^\sharp_A\big\{\alpha_A(\varphi(\nu)) \ \big|\ \alpha_A(\nu) \sqsubseteq^\sharp_A 0\big\} \sqsubseteq^\sharp 0$ implies $\alpha_A(\varphi(\bot_A)) = 0$ whence $\varphi(\bot_A) = \bot_A$. This strictness proof method is not relatively complete since values are ignored.

Inducing the abstract interpretation. We look for $\llbracket e_\tau \rrbracket^\sharp \in \mathcal{P}^\sharp$ such that $\alpha_\tau\big(\llbracket e_\tau \rrbracket^\natural \rho\big) \sqsubseteq^\sharp_\tau \llbracket e_\tau \rrbracket^\sharp \alpha_{\mathcal{E}}(\rho)$. We proceed by induction on e_τ:[11]

- $\alpha_\tau\big(\llbracket x_\tau \rrbracket^\natural \rho\big) = \alpha_\tau\big(\rho(x_\tau)\big) = \alpha_{\mathcal{E}}(\rho) x_\tau = \llbracket x_\tau \rrbracket^\sharp \alpha_{\mathcal{E}}(\rho)$ by defining $\llbracket x_\tau \rrbracket^\sharp \rho^\sharp \stackrel{\text{def}}{=} \rho^\sharp(x_\tau)$.

- $\alpha_\tau\big(\llbracket c_\tau \rrbracket^\natural \rho\big) = \alpha_\tau(\underline{c}_\tau) = \llbracket c_\tau \rrbracket^\sharp \alpha_{\mathcal{E}}(\rho)$ by defining $\llbracket c_\tau \rrbracket^\sharp \rho^\sharp \stackrel{\text{def}}{=} \alpha_\tau(\underline{c}_\tau)$.

- $\alpha_{\tau_2 \to \tau_1}\big(\llbracket \lambda\, x_{\tau_2} \bullet e_{\tau_1} \rrbracket^\natural \rho\big) = \alpha_{\tau_2 \to \tau_1}\big(\lambda \nu \in D_{\tau_2} \bullet \llbracket e_{\tau_1} \rrbracket^\natural \rho[x_{\tau_2} \to \nu]\big)$ by (4.29) which, by (4.31), is equal to $\lambda a \bullet \sqcup^\sharp \big\{ \alpha_{\tau_1}(\llbracket e_{\tau_1} \rrbracket^\natural \rho\, [x_{\tau_2} \to \nu]) \bigm| \alpha_{\tau_2}(\nu) \sqsubseteq^\sharp a \big\}$ and, by induction hypothesis, is less than or equal to $\lambda\, a \bullet \sqcup^\sharp \big\{ \llbracket e_{\tau_1} \rrbracket^\sharp \alpha_{\mathcal{E}}(\rho\, [x_{\tau_2} \to \nu]) \bigm| \alpha_{\tau_2}(\nu) \sqsubseteq^\sharp a \big\}$. By definition of $\alpha_{\mathcal{E}}$, this is equal to $\lambda\, a \bullet \sqcup^\sharp \big\{ \llbracket e_{\tau_1} \rrbracket^\sharp \alpha_{\mathcal{E}}(\rho)\, [x_{\tau_2} \to \alpha_{\tau_2}(\nu)] \bigm| \alpha_{\tau_2}(\nu) \sqsubseteq^\sharp a \big\}$. α_{τ_2} is surjective so that for $\nu \in D_{\tau_2}$ such that $a = \alpha_{\tau_2}(\nu)$ this is $\sqsubseteq^\sharp$ $\lambda\, a \bullet \llbracket e_{\tau_1} \rrbracket^\sharp \alpha_{\mathcal{E}}(\rho)[x_{\tau_2} \to a]^\sharp$ by monotony and definition of least upper bounds (implying $\sqcup \{ \varphi(x) \mid x \sqsubseteq y \} \sqsubseteq \varphi(y)$). This can be written as $\llbracket \lambda\, x_{\tau_2} \bullet e_{\tau_1} \rrbracket^\sharp \alpha_{\mathcal{E}}(\rho)$ by defining $\llbracket \lambda\, x_\tau \bullet e \rrbracket^\sharp \rho^\sharp \stackrel{\text{def}}{=} \lambda\, \nu \in D^\sharp_\tau \bullet \llbracket e \rrbracket^\sharp\ \rho^\sharp[x_\tau \to \nu]$.

- We have $\alpha_{\tau_1}(\varphi(\nu)) \sqsubseteq^\sharp \sqcup^\sharp \big\{ \alpha_{\tau_1}(\varphi(\upsilon)) \mid \alpha_{\tau_2}(\upsilon) \sqsubseteq^\sharp \alpha_{\tau_2}(\nu) \big\} = \alpha_{\tau_2 \to \tau_1}(\varphi)\ \alpha_{\tau_2}(\nu)$ by (4.31) and definition of least upper bounds when $\upsilon = \nu$. It follows that $\alpha_{\tau_1}\big(\llbracket e_1 e_2 \rrbracket^\natural \rho\big) = \alpha_{\tau_1}\big((\llbracket e_1 \rrbracket^\natural \rho)(\llbracket e_2 \rrbracket^\natural \rho)\big) \sqsubseteq^\sharp \alpha_{\tau_2 \to \tau_1}(\llbracket e_1 \rrbracket^\natural \rho)\ \alpha_{\tau_2}(\llbracket e_2 \rrbracket^\natural \rho)$ by induction hypothesis and pointwise ordering. Again by induction hypothesis and monotony, we have $\alpha_{\tau_2 \to \tau_1}(\llbracket e_1 \rrbracket^\natural \rho)\ \alpha_{\tau_2}(\llbracket e_2 \rrbracket^\natural \rho) \sqsubseteq^\sharp \big(\llbracket e_1 \rrbracket^\sharp \alpha_{\mathcal{E}}(\rho)\big)\big(\alpha_{\tau_2}(\llbracket e_2 \rrbracket^\natural \rho)\big) \sqsubseteq^\sharp \big(\llbracket e_1 \rrbracket^\sharp \alpha_{\mathcal{E}}(\rho)\big)\big(\llbracket e_2 \rrbracket^\sharp \alpha_{\mathcal{E}}(\rho)\big) = \llbracket e_1 e_2 \rrbracket^\sharp \alpha_{\mathcal{E}}(\rho)$ if $\llbracket e_1 e_2 \rrbracket^\sharp \rho^\sharp \stackrel{\text{def}}{=} \big(\llbracket e_1 \rrbracket^\sharp \rho^\sharp\big)\big(\llbracket e_2 \rrbracket^\sharp \rho^\sharp\big)$.

To sum up, we have formally calculated that the abstract semantics $\llbracket e_\tau \rrbracket^\sharp \rho^\sharp$ of term e_τ in abstract environment $\rho^\sharp$ can be defined as:

$$\begin{array}{lcl@{\qquad\qquad}lcl}
\llbracket x_\tau \rrbracket^\sharp \rho^\sharp & \stackrel{\text{def}}{=} & \rho^\sharp(x_\tau) & \llbracket \lambda\, x_\tau \bullet e \rrbracket^\sharp \rho^\sharp & \stackrel{\text{def}}{=} & \lambda\, \nu \in D^\sharp_\tau \bullet \llbracket e \rrbracket^\sharp\ \rho^\sharp[x_\tau \to \nu] \\
\llbracket c_\tau \rrbracket^\sharp \rho^\sharp & \stackrel{\text{def}}{=} & \alpha_\tau(\underline{c}_\tau) \sqsubseteq^\sharp_\tau \underline{c}^\sharp_\tau & \llbracket e_1 e_2 \rrbracket^\sharp \rho^\sharp & \stackrel{\text{def}}{=} & \big(\llbracket e_1 \rrbracket^\sharp \rho^\sharp\big)\big(\llbracket e_2 \rrbracket^\sharp \rho^\sharp\big)\ .
\end{array} \tag{4.32}$$

The abstract interpretation of constants is:

$$\underline{\mathit{fix}}^\sharp(\varphi) \stackrel{\text{def}}{=} \bigsqcup_{n \geq 0}^\sharp \varphi^n(\bot^\sharp_\tau) \qquad\qquad \begin{array}{lcl} \underline{\mathit{cond}}^\sharp\, 0\, x\, y & \stackrel{\text{def}}{=} & \bot^\sharp_\tau \\ \underline{\mathit{cond}}^\sharp\, 1\, x\, y & \stackrel{\text{def}}{=} & x \sqcup y\ . \end{array} \tag{4.33}$$

For example, let us consider $\underline{\mathit{fix}}^\sharp$. By induction on type τ, using (4.30) and (4.31), α_τ is a strict $\alpha_\tau(\bot_\tau) = \bot^\sharp_\tau$ complete join morphism $\alpha_\tau\big(\sqcup_{i\,\tau} \nu_i\big) = \sqcup^\sharp_{i\,\tau} \alpha_\tau(\nu_i)$ when $\sqcup_{i\,\tau} \nu_i$ exists. Moreover, we have proved that if $\varphi = \llbracket e_\tau \rrbracket^\natural \rho$ and $\varphi^\sharp = \llbracket e_\tau \rrbracket^\sharp \alpha_{\mathcal{E}}(\rho)$ then $\alpha_{\tau \to \tau}(\varphi) \sqsubseteq^\sharp_{\tau \to \tau} \varphi^\sharp$ whence, by definition of the pointwise ordering, for all $\nu^\sharp \in {D^\sharp}_\tau$: $(\alpha_{\tau \to \tau}(\varphi))\nu^\sharp \sqsubseteq^\sharp_\tau \varphi^\sharp \nu^\sharp$. It follows by (4.31) that $\sqcup^\sharp_\tau \big\{ \alpha_\tau(\varphi(\nu)) \mid \alpha_\tau(\nu) \sqsubseteq^\sharp_\tau \nu^\sharp \big\} \sqsubseteq^\sharp_\tau \varphi^\sharp \nu^\sharp$ whence in particular for $\nu = \nu^\natural$, if $\alpha_\tau(\nu^\natural) \sqsubseteq^\sharp_\tau \nu^\sharp$ then we have $\alpha_\tau(\varphi(\nu^\natural)) \sqsubseteq^\sharp_\tau \varphi^\sharp \nu^\sharp$ by definition of least upper bounds. By Proposition 6.10, we conclude that $\alpha(\underline{\mathit{fix}}) \sqsubseteq^\sharp \underline{\mathit{fix}}^\sharp$.

[11] The objective of the following calculation is to illustrate the constructive aspect of the approach, which we have found very useful in practice as a guideline for inferring abstract interpreters from a semantics, as opposed to *a posteriori* verifications using a soundness relation, e.g. as in [1].

In general Proposition 6.10 is used with stronger hypotheses such as, for example, that the abstract iteration is defined inductively by $\langle \underline{\perp}^\sharp,\ F^\sharp,\ \sqcup^\sharp\rangle$ where $\underline{\perp}^\sharp \overset{\text{def}}{=} \perp^\sharp$ is the infimum for $\preceq^\sharp$, $\forall c \in \mathcal{P}^\natural\colon \forall a \in \mathcal{P}^\sharp\colon \alpha(c) \preceq^\sharp a \Rightarrow \alpha(F^\natural(c)) \preceq^\sharp F^\sharp(a)$ (which holds in particular when $F^\sharp$ is monotonic for $\preceq^\sharp$ and $\alpha \circ F^\natural \preceq^\sharp F^\sharp \circ \alpha$), $\sqcup^\sharp$ is the least upper bound for $\preceq^\sharp$ and the abstract iterates are total because, among others, $\langle \mathcal{P}^\sharp;\ \preceq^\sharp\rangle$ is a cpo or a complete lattice. Such hypotheses are satisfied in Example 6.11. See for example proposition 7.1.0.4.(2) of [12], which can be stated as follows:

Proposition 6.12 (Semantic approximation using a Galois connection)
If $\langle \mathcal{P}^\natural;\ \sqsubseteq^\natural,\ \perp^\natural,\ \sqcup^\natural\rangle$ is a cpo, $F^\natural \in \mathcal{P}^\natural \xmapsto{m} \mathcal{P}^\natural$, $\langle \mathcal{P}^\sharp;\ \sqsubseteq^\sharp\rangle$ is a poset, $\langle \mathcal{P}^\natural;\ \sqsubseteq^\natural\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathcal{P}^\sharp;\ \sqsubseteq^\sharp\rangle$, $\alpha(\perp^\natural) = \underline{\perp}^\sharp$, $F^\sharp \in \mathcal{P}^\sharp \mapsto \mathcal{P}^\sharp$ is $F^\sharp \overset{\text{def}}{=} \alpha \circ F^\natural \circ \gamma$ then the abstract iteration sequence $F^{\sharp 0} \overset{\text{def}}{=} \underline{\perp}^\sharp$, $F^{\sharp \lambda+1} \overset{\text{def}}{=} F^\sharp(F^{\sharp\lambda})$ for $\lambda \in \mathit{Ord}$ and $F^{\sharp\lambda} \overset{\text{def}}{=} \sqcup^\sharp_{\beta<\lambda} F^{\sharp\beta}$ for all limit ordinals $\lambda > 0$ is convergent such that $\alpha(\mathit{lfp}^{\sqsubseteq^\natural}_{\perp^\natural} F^\natural) \sqsubseteq^\sharp F^{\sharp\varepsilon} = \mathit{lfp}^{\sqsubseteq^\sharp}_{\underline{\perp}^\sharp} F^\sharp$.

Proof. Follows from Proposition 6.10 since the Galois connection ensures the existence of the abstract least upper bounds $\sqcup^\sharp_{\beta<\lambda} F^{\sharp\beta}$. ∎

Example 6.13 (Collecting semantics for Burn, Hankin and Abramsky's strictness analysis method)
The original construction of [3] is based upon Proposition 6.12 using Hoare power domains for the collecting semantics (whereas [35] used a variant of Plotkin's powerdomain so as to handle also termination analysis) and Galois connections. The main idea is to lift Mycroft's abstraction (4.30) to higher-order function spaces. This idea is of general use as shown by theorem 4.1.7.1 of [10] based upon (4.17).

Widening. The specification (4.9), (4.10) and (4.11) of an abstract interpreter is not directly implementable since, in general, $F^\natural$, α, σ and $\amalg^\sharp$ are not computable. The first solution considered in Propositions 6.8, 6.9, 6.10 and 6.12 consists in providing an abstract semantic function $F^\sharp$ which is an approximation of $\lambda a \cdot \triangledown^\sharp\big\{\alpha(F^\natural(c)) \bigm| c \in \mathcal{P}^\natural \wedge \langle c,\ a\rangle \in \sigma\big\}$. Another solution would consist in implementing the widening $\triangledown^\sharp$ as well as an approximation $F^\sharp \in \mathcal{P}^\sharp \mapsto \wp(\mathcal{P}^\sharp)$ of $\lambda a \cdot \big\{\alpha(F^\natural(c)) \bigm| c \in \mathcal{P}^\natural \wedge \langle c,\ a\rangle \in \sigma\big\}$ but we know very few examples of application of this approach since, for practical reasons, one wants to avoid representing sets of abstract properties $F^\sharp(a)$, just to select one $\triangledown^\sharp(F^\sharp(a))$.[12]

However, the idea has been generalized in [7] by considering an abstract iteration depending on previous iterates:

$$\left\{\begin{array}{lcl} F^{\sharp 0} & = & \underline{\perp}^\sharp \\ F^{\sharp\lambda} & = & \triangledown^\sharp(\{\, F^\sharp(F^{\sharp\beta}) \mid \beta < \lambda\} \cup \{\underline{\perp}^\sharp\}) \ . \end{array}\right. \tag{4.34}$$

In [19] it was observed that the widening may differ at each iteration step which consists in considering iterations of the form:

$$\left\{\begin{array}{lcl} F^{\sharp 0} & = & \underline{\perp}^\sharp \\ F^{\sharp\lambda} & = & \triangledown^\sharp_\lambda(\{\, \langle\beta,\ F^\sharp(F^{\sharp\beta})\rangle \mid \beta < \lambda\} \cup \big\{\langle 0,\ \underline{\perp}^\sharp\rangle\big\}) \ . \end{array}\right. \tag{4.35}$$

[12] An example is [23] where local decreasing iterations are considered to improve the conjunction of abstract reductive operators which, for example, appears in tests.

To simplify the presentation, one can consider only two consecutive iterates and define the *abstract iteration sequence with widening*:

$$\left\{\begin{array}{rcll} F^{\sharp\uparrow 0} & \stackrel{\text{def}}{=} & \bot^{\sharp} & \\ F^{\sharp\uparrow\lambda+1} & \stackrel{\text{def}}{=} & F^{\sharp\uparrow\lambda} \mathbin{\nabla^{\sharp}} F^{\sharp}(F^{\sharp\uparrow\lambda}) & \lambda \in \mathit{Ord} \\ F^{\sharp\uparrow\lambda} & \stackrel{\text{def}}{=} & \nabla^{\sharp}_{\beta<\lambda} F^{\sharp\uparrow\beta} & \text{for limit ordinals } \lambda > 0\ . \end{array}\right. \tag{4.36}$$

The general soundness condition (4.7) for the widening can be refined using the approximation ordering $\preceq^{\sharp}$. For example given an abstraction relation α and defining soundness by (4.21), we obtain:

$$\begin{array}{l} \nabla^{\sharp} A \text{ exists} \wedge c \in \mathcal{P}^{\natural} \wedge a' \in \mathcal{P}^{\sharp} \wedge a \in A \ \Rightarrow \\ \quad (\langle c,\, a'\rangle \in \alpha \wedge a' \preceq^{\sharp} a) \ \Rightarrow\ (\exists a'' \in \mathcal{P}^{\sharp} : \langle c,\, a''\rangle \in \alpha \wedge a'' \preceq^{\sharp} \nabla^{\sharp} A)\ . \end{array} \tag{4.37}$$

In particular if α is an abstraction function this is equivalent to:

$$\nabla^{\sharp} A \text{ exists} \ \wedge\ c \in \mathcal{P}^{\natural} \ \wedge\ a \in A \ \wedge\ \alpha(c) \preceq^{\sharp} a \quad\Rightarrow\quad \alpha(c) \preceq^{\sharp} \nabla^{\sharp} A \tag{4.38}$$

which is implied by the stronger requirement:

$$\nabla^{\sharp} A \text{ exists} \ \wedge\ a \in A \quad\Rightarrow\quad a \preceq^{\sharp} \nabla^{\sharp} A \tag{4.39}$$

which holds when the widening is a partially defined upper bound in $\mathcal{P}^{\sharp}$ (but not necessarily the least one). The widening is also used as an abstract join, as in (4.34), in the sense that:

$$\coprod^{\natural}_{i\in I} c_i \text{ exists} \ \wedge\ \mathop{\nabla^{\sharp}}_{i\in I} a_i \text{ exists} \ \wedge\ \forall i \in I : \alpha(c_i) \preceq^{\sharp} a_i \quad\Rightarrow\quad \alpha\Big(\coprod^{\natural}_{i\in I} c_i\Big) \preceq^{\sharp} \mathop{\nabla^{\sharp}}_{i\in I} a_i \tag{4.40}$$

When using a widening, Proposition 6.10 becomes:

Proposition 6.14 (Semantic approximation using an abstraction function and a widening)
Given the sets $\mathcal{P}^{\natural}$ of concrete properties and $\langle \mathcal{P}^{\sharp};\ \preceq^{\sharp}\rangle$ of abstract properties with pre-order $\preceq^{\sharp}$, the abstraction function $\alpha \in \mathcal{P}^{\natural} \mapsto \mathcal{P}^{\sharp}$, the bases $\bot^{\natural}$ and $\bot^{\sharp}$ such that $\alpha(\bot^{\natural}) \preceq^{\sharp} \bot^{\sharp}$, the concrete $F^{\natural} \in \mathcal{P}^{\natural} \rightarrowtail \mathcal{P}^{\natural}$ and abstract $F^{\sharp} \in \mathcal{P}^{\sharp} \rightarrowtail \mathcal{P}^{\sharp}$ semantic functions such that $\forall c \in \mathcal{P}^{\natural}$: $\forall a \in \mathcal{P}^{\sharp}$: $\alpha(c) \preceq^{\sharp} a \Rightarrow \alpha(F^{\natural}(c)) \preceq^{\sharp} F^{\sharp}(a)$ (when $F^{\natural}(c)$ and $F^{\sharp}(a)$ are well-defined and holds in particular when $F^{\sharp}$ is monotonic for $\preceq^{\sharp}$ and $\alpha \circ F^{\natural} \preceq^{\sharp} F^{\sharp} \circ \alpha$), the concrete inductive join $\coprod^{\natural} \in \wp(\mathcal{P}^{\natural}) \rightarrowtail \mathcal{P}^{\natural}$, the widening $\nabla^{\sharp} \in \wp(\mathcal{P}^{\sharp}) \rightarrowtail \mathcal{P}^{\sharp}$ satisfying (4.38) and (4.40) and assuming that the concrete iteration sequence (4.1) and abstract iteration sequence with widening (4.36) are convergent then their respective limits $F^{\natural\epsilon}$ and $F^{\sharp\uparrow\varepsilon}$ are such that $\alpha(F^{\natural\epsilon}) \preceq^{\sharp} F^{\sharp\uparrow\varepsilon}$.

Proof. If $\alpha(F^{\natural\lambda}) \preceq^{\sharp} F^{\sharp\uparrow\lambda}$ then by (4.1) we have $\alpha(F^{\natural\lambda+1}) = \alpha(F^{\natural}(F^{\natural\lambda})) \preceq^{\sharp} F^{\sharp}(F^{\sharp\uparrow\lambda})$ whence $\alpha(F^{\natural\lambda+1}) \preceq^{\sharp} F^{\sharp\uparrow\lambda} \mathbin{\nabla^{\sharp}} F^{\sharp}(F^{\sharp\uparrow\lambda}) = F^{\sharp\uparrow\lambda+1}$ by (4.38) and (4.36). Moreover, for all limit ordinals $\lambda > 0$, $\forall \beta < \lambda$: $\alpha(F^{\natural\beta}) \preceq^{\sharp} F^{\sharp\uparrow\beta}$ implies $\alpha(\coprod^{\natural}_{\beta<\lambda} F^{\natural\beta}) \preceq^{\sharp} F^{\sharp\uparrow\lambda}$ by (4.40) and (4.36). We conclude by Proposition 6.10. ∎

Convergence criteria. Up to now, we have not discussed the convergence problem for the abstract iteration $F^{\sharp\lambda}$, $\lambda \in \mathit{Ord}$ and assumed that it was "ultimately stationary". When the abstract iteration is defined by a semantic function this means that a fixpoint is reached since $F^{\sharp\varepsilon} = F^{\sharp\varepsilon+1} = F^\sharp(F^{\sharp\varepsilon})$. When using Proposition 6.10 or its corollary 6.14 where the iteration introduces further approximations than those required by α, stabilization on a fixpoint may no longer be necessary. So a fourth basic choice in the design of an abstract interpretation is to find a convenient convergence criterion, ensuring the best possible precision.

Example 6.15 (Minimal function graph)
The invariance analysis of side-effect-free first-order procedures with value and result parameters which, but for syntactic sugaring, includes functions is considered in [10]. The standard semantics is relational in that the semantics of mutually recursive procedures is understood as a relation between the values of their value parameters and that of their result parameters. It is obtained as the least fixpoint of a higher-order system of fixpoint equations $\vec{f^\natural} = \vec{F^\natural}(\vec{f^\natural})$. An abstract interpretation $\vec{f^\sharp} = \vec{F^\sharp}(\vec{f^\sharp})$ is derived using (4.17). For example the abstract equation for program `f(x) = if x=0 then 0 else f(-x)` is $F(f) = \lambda x\cdot \text{if } x = \bot \text{ then } \bot \text{ else } 0 \sqcup f(-x)$ for the rule of signs lattice of Example 7.2. Two methods are considered for solving such equations:

1. An algebraic method using (4.2) and a symbolic representation of the abstract functions $f^\sharp$ which can be applied to relational abstract analyses so as to analyse the (mutually recursive) procedure/function bodies, once for all, independently of the main calls. For the above example the iterates would be $f^{\sharp 0} = \lambda a\cdot\bot$, $f^{\sharp 1} = \lambda a\cdot 0 \sqcup f^{\sharp 0}(-a) = \lambda a\cdot 0$, $f^{\sharp 2} = f^{\sharp 1}$. For a recent application of this technique see the"on-line" analyses of [21].

2. A chaotic iteration method (theorem 4.2.1.2 in [10], popularized as "minimal function graph" by [27]), which is based on the observation that in practice $f^\sharp(a)$ needs to be known only for some, but not all its abstract arguments a so that the abstract resolution of $\vec{f^\sharp} = \vec{F^\sharp}(\vec{f^\sharp})$ demands collecting the set of points on which the abstract functions $f^\sharp$ are actually called with during the computation on an initial set of abstract arguments, given by the main calls. For the above example the iterates for $f(+)$ would be $f^{\sharp 0}(a) = \bot$, $f^{\sharp 1}(+) = 0 \sqcup f^{\sharp 0}(-) = 0$, $f^{\sharp 1}(-) = 0 \sqcup f^{\sharp 1}(+) = 0$, $f^{\sharp 2}(+) = f^{\sharp 1}(+)$, $f^{\sharp 2}(-) = f^{\sharp 1}(-)$.

Observe that a fixpoint is computed in case 1 whereas $f^\sharp(\bot)$, $f^\sharp(-)$ and $f^\sharp(\top)$ have not be computed in case 2 so that the iterates do not stabilize on a fixpoint, but on a post-fixpoint by considering that $f^\sharp(\bot)$, $f^\sharp(-)$ and $f^\sharp(\top)$ are unknown hence equal to $\top$.

Example 6.16 (post-fixpoints)
Let us consider the case when the approximation ordering $\preceq^\sharp$ is a partial order, $F^\sharp$ is a monotonic operator on a complete lattice $\langle \mathcal{P}^\sharp;\ \preceq^\sharp,\ \sqcup^\sharp,\ \sqcap^\sharp\rangle$ and it is known that $\alpha(F^{\natural\epsilon}) \preceq^\sharp \mathit{lfp}\, F^\sharp$. Then, as suggested in definition 9.1.1.3 of [7], we can stop iterations (4.36) as soon as a post-fixpoint $F^{\sharp\varepsilon}$ of $F^\sharp$ (such that $F^\sharp(F^{\sharp\varepsilon}) \preceq^\sharp F^{\sharp\varepsilon}$) is reached since, by Tarski's fixpoint theorem, we have $\mathit{lfp}\, F^\sharp = \sqcap^\sharp \left\{ X \in \mathcal{P}^\sharp \,\middle|\, F^\sharp(X) \preceq^\sharp X \right\}$ hence $\mathit{lfp}\, F^\sharp \preceq^\sharp F^{\sharp\varepsilon}$, so that by transitivity $\alpha(F^{\natural\epsilon}) \preceq^\sharp F^{\sharp\varepsilon}$.

Convergence to a post-fixpoint can be enforced using a widening satisfying (4.39) and the result can then be improved using a narrowing. In this case the abstract iteration sequence with widening (4.36) becomes:

$$\left\{\begin{array}{rcllr} F^{\sharp\uparrow 0} & \overset{\text{def}}{=} & \perp^{\sharp} & & \\ F^{\sharp\uparrow \lambda+1} & \overset{\text{def}}{=} & F^{\sharp\uparrow \lambda} & \text{if } F^{\sharp}(F^{\sharp\uparrow \lambda}) \preceq^{\sharp} F^{\sharp\uparrow \lambda} & \lambda \in \mathit{Ord} \\ & \overset{\text{def}}{=} & F^{\sharp\uparrow \lambda} \mathbin{\nabla^{\sharp}} F^{\sharp}(F^{\sharp\uparrow \lambda}) & \text{otherwise} & \\ F^{\sharp\uparrow \lambda} & \overset{\text{def}}{=} & F^{\sharp\uparrow \beta} & \text{if } \exists \beta < \lambda : F^{\sharp}(F^{\sharp\uparrow \beta}) \preceq^{\sharp} F^{\sharp\uparrow \beta} & \lambda > 0 \text{ limit} \\ & \overset{\text{def}}{=} & \nabla^{\sharp}_{\beta<\lambda} F^{\sharp\uparrow \beta} & \text{otherwise} & \text{ordinal.} \end{array}\right. \tag{4.41}$$

The following propositions which explain this use of widenings and narrowings in abstract interpretation are a variation on paragraphs 9.1 and 9.3 of [7]:

Proposition 6.17 (Convergence of the semantic approximation to a post-fixpoint using a widening)
Let $\mathcal{P}^{\sharp}(\preceq^{\sharp})$ be a partially ordered set of abstract properties, $\perp^{\sharp}$ be an abstract basis, $F^{\sharp} \in \mathcal{P}^{\sharp} \rightarrowtail \mathcal{P}^{\sharp}$ be an abstract semantic function and $\nabla^{\sharp} \in \wp(\mathcal{P}^{\sharp}) \rightarrowtail \mathcal{P}^{\sharp}$ be a widening satisfying (4.39) such that the abstract iteration sequence with widening (4.41) is total.
1°) This iteration sequence is increasing with respect to $\preceq^{\sharp}$ whence stationary. Its limit $F^{\sharp\uparrow\varepsilon}$ is a post-fixpoint of $F^{\sharp}$ greater than or equal to $\perp^{\sharp}$.
2°) If $F^{\sharp}$ is monotonic and $\nabla^{\sharp}$ is the partially defined least upper bound $\sqcup^{\sharp}$ in $\mathcal{P}^{\sharp}$, then $F^{\sharp\uparrow\varepsilon}$ is the least such post-fixpoint.
3°) If $F^{\sharp}$ is monotonic, $\perp^{\sharp}$ is a pre-fixpoint of $F^{\sharp}$ and $\nabla^{\sharp}$ is $\sqcup^{\sharp}$ then $F^{\sharp\uparrow\varepsilon}$ is the least fixpoint of $F^{\sharp}$ greater than or equal to $\perp^{\sharp}$.

Proof. 1°) If the iteration sequence $F^{\sharp\uparrow\lambda}$, for $\lambda \in \mathit{Ord}$ is total then its terms are all well-defined. Obviously, $F^{\sharp\uparrow\lambda} \preceq^{\sharp} F^{\sharp\uparrow\lambda} \nabla^{\sharp} F^{\sharp}(F^{\sharp\uparrow\lambda}) = F^{\sharp\uparrow\lambda+1}$ since, by (4.39), $\nabla^{\sharp}$ is an upper bound for the partial order $\preceq^{\sharp}$. If $\lambda > 0$ is a limit ordinal and $\delta < \lambda$ then $F^{\sharp\uparrow\delta} \preceq^{\sharp} \nabla^{\sharp}_{\beta<\lambda} F^{\sharp\uparrow\beta} = F^{\sharp\uparrow\lambda}$, proving that the iteration sequence is an increasing chain. It cannot be strictly increasing since its cardinality must be less than or equal to that of the set $\mathcal{P}^{\sharp}$ and repetitions are disallowed by antisymmetry. Hence there is an ordinal and therefore a smallest one, ε, such that $F^{\sharp\uparrow\varepsilon} = F^{\sharp\uparrow\varepsilon+1} = F^{\sharp\uparrow\varepsilon} \nabla^{\sharp} F^{\sharp}(F^{\sharp\uparrow\varepsilon})$ hence $F^{\sharp}(F^{\sharp\uparrow\varepsilon}) \preceq^{\sharp} F^{\sharp\uparrow\varepsilon}$ since $\nabla^{\sharp}$ is an upper bound for $\preceq^{\sharp}$. We have $\perp^{\sharp} = F^{\sharp\uparrow 0} \preceq^{\sharp} F^{\sharp\uparrow\varepsilon}$.
2°) Let x be any post-fixpoint of $F^{\sharp}$ such that $F^{\sharp\uparrow 0} = \perp^{\sharp} \preceq^{\sharp} x$. If, by induction hypothesis, $F^{\sharp\uparrow\lambda} \preceq^{\sharp} x$, then $F^{\sharp}(F^{\sharp\uparrow\lambda}) \preceq^{\sharp} F^{\sharp}(x) \preceq^{\sharp} x$ by monotony and post-fixpoint property, hence $F^{\sharp\uparrow\lambda+1} = F^{\sharp\uparrow\lambda} \nabla^{\sharp} F^{\sharp}(F^{\sharp\uparrow\lambda}) \preceq^{\sharp} x$ since $\nabla^{\sharp}$ is the least upper bound $\sqcup^{\sharp}$. If, by induction hypothesis, $F^{\sharp\uparrow\beta} \preceq^{\sharp} x$ for all $\beta < \lambda$ then $F^{\sharp\uparrow\lambda} = \nabla^{\sharp}_{\beta<\lambda} F^{\sharp\uparrow\beta} \preceq^{\sharp} x$ since $\nabla^{\sharp}$ is the least upper bound $\sqcup^{\sharp}$ for $\preceq^{\sharp}$. By transfinite induction, $\forall\lambda \in \mathit{Ord} : F^{\sharp\uparrow\lambda} \preceq^{\sharp} x$ so that, in particular, $F^{\sharp\uparrow\varepsilon} \preceq^{\sharp} x$.
3°) We have $F^{\sharp\uparrow 0} = \perp^{\sharp} \preceq^{\sharp} F^{\sharp}(\perp^{\sharp}) \preceq^{\sharp} F^{\sharp\uparrow 1}$ by (4.39) and (4.41). If $F^{\sharp\uparrow\lambda} \preceq^{\sharp} F^{\sharp}(F^{\sharp\uparrow\lambda}) = F^{\sharp\uparrow\lambda+1}$ then by monotony $F^{\sharp\uparrow\lambda+1} = F^{\sharp}(F^{\sharp\uparrow\lambda}) \preceq^{\sharp} F^{\sharp}(F^{\sharp\uparrow\lambda+1})$ proving that $F^{\sharp\uparrow\lambda+1}$ is also a pre-fixpoint of $F^{\sharp}$. Moreover, $\forall\beta < \lambda : F^{\sharp\uparrow\beta} \preceq^{\sharp} \sqcup^{\sharp}_{\beta<\lambda} F^{\sharp\uparrow\beta}$ hence, by monotony, $F^{\sharp\uparrow\beta} \preceq^{\sharp} F^{\sharp}(F^{\sharp\uparrow\beta}) \preceq^{\sharp} F^{\sharp}(\sqcup^{\sharp}_{\beta<\lambda} F^{\sharp\uparrow\beta}) = F^{\sharp}(F^{\sharp\uparrow\lambda})$ so that $F^{\sharp\uparrow\lambda} = \sqcup^{\sharp}_{\beta<\lambda} F^{\sharp\uparrow\beta} \preceq^{\sharp} F^{\sharp}(F^{\sharp\uparrow\lambda})$ proving that $F^{\sharp\uparrow\lambda}$ is also a pre-fixpoint of $F^{\sharp}$. By transfinite induction, all terms of the iteration sequence are pre-fixpoints of $F^{\sharp}$ proving by antisymmetry that $F^{\sharp\uparrow\varepsilon}$ is a fixpoint of $F^{\sharp}$, since it has already been proved to be a post-fixpoint. Since it is

the least post-fixpoint greater than or equal to $\perp^{\sharp}$, it is also the least such fixpoint, since $\preceq^{\sharp}$ is reflexive. ■

Observe that the main use of the widening $\triangledown^{\sharp}$ in Proposition 6.17 is to serve as a palliative to the non-existence of least upper bounds for $\preceq^{\sharp}$ in $\mathcal{P}^{\sharp}$. This is indispensable in [17] for example since the limit of chains of convex polyhedra increasing for inclusion may not be a convex polyhedron. To be more specific on the reasons for convergence of (4.41), it can be assumed, for example, that $\langle\mathcal{P}^{\sharp};\ \sqsubseteq^{\sharp},\ \triangledown^{\sharp}\rangle$ is a cpo and that $F^{\sharp}$ is total and either extensive or monotonic and $\perp^{\sharp}$ is a pre-fixpoint for $\sqsubseteq^{\sharp}$ or that ultimate stabilization of the iteration sequence follows from metric arguments.

Narrowing. Let us define the *abstract iteration sequence with narrowing starting from* $a \in \mathcal{P}^{\sharp}$:

$$\begin{cases} F^{\sharp\downarrow 0} & \overset{\text{def}}{=} & a & \\ F^{\sharp\downarrow \lambda+1} & \overset{\text{def}}{=} & F^{\sharp\downarrow \lambda} \vartriangle^{\sharp} F^{\sharp}(F^{\sharp\downarrow \lambda}) & \lambda \in \mathit{Ord} \\ F^{\sharp\downarrow \lambda} & \overset{\text{def}}{=} & \vartriangle^{\sharp}_{\beta<\lambda} F^{\sharp\downarrow \beta} & \text{for limit ordinals } \lambda > 0\ . \end{cases} \tag{4.42}$$

Definitions (4.36) and (4.42) are an oversimplification since in practice [12]:

- The definitions $a \triangledown^{\sharp} a' \overset{\text{def}}{=} \triangledown^{\sharp}\{a, a'\}$ and $a \vartriangle^{\sharp} a' \overset{\text{def}}{=} \vartriangle^{\sharp}\{a, a'\}$ do not take non-commutativity into account (but see (4.46)).
- The widening and narrowing operators are not always between two consecutive iterates since they are used only once within any loop or recursive call and more than one iteration is needed to handle the loop or procedure/function body, so that:
- The widening and narrowing operators depend upon the rank of the iterates, as in [19]. For example, a simple way to balance the precision/cost tradeoff is to define $x \triangledown_{\lambda} y \overset{\text{def}}{=}$ if $\lambda < n$ then x else $\top^{\sharp}$ where n is a user adjustable parameter. A similar simple narrowing would be $x \vartriangle_{\lambda} y \overset{\text{def}}{=}$ if $\lambda < n$ then x else y. In practice such naïve operators would be effective to handle loops and recursion only.

Hypotheses on the widening (4.39) and on the narrowing (4.44) differ from the original ones [6] which admit several other slight variations (see, for example, [4, 15, 21]).

When soundness is defined by (4.21), the general soundness condition (4.8) for the narrowing can be refined using the abstraction function α and the approximation ordering $\preceq^{\sharp}$:

$$\begin{array}{c} \forall A \subseteq \mathcal{P}^{\sharp} : \vartriangle^{\sharp}A \text{ exists } \ \wedge\ c \in \mathcal{P}^{\natural}\ \Rightarrow \\ (\forall a \in A : \alpha(c) \preceq^{\sharp} a)\ \Rightarrow\ (\alpha(c) \preceq^{\sharp} \vartriangle^{\sharp}A) \end{array} \tag{4.43}$$

which is implied by the stronger requirement that a narrowing is a partially defined upper bound of greatest lower bounds in $\mathcal{P}^{\sharp}$:

$$\forall A \subseteq \mathcal{P}^{\sharp} : \text{if } \vartriangle^{\sharp}A \text{ exists then } \sqcap^{\sharp}A \text{ exists and } \exists a \in A : \sqcap^{\sharp}A \preceq^{\sharp} \vartriangle^{\sharp}X \preceq^{\sharp} a\ . \tag{4.44}$$

A narrowing can be used to improve an abstract upper approximation a of a concrete fixpoint c (typically a is obtained by Proposition 6.14):

Proposition 6.18 (Semantic approximation using an abstraction function and a narrowing)
Given the sets $\mathcal{P}^\natural$ of concrete properties and $\langle \mathcal{P}^\sharp;\ \preceq^\sharp \rangle$ of abstract properties with pre-order $\preceq^\sharp$, the abstraction function $\alpha \in \mathcal{P}^\natural \mapsto \mathcal{P}^\sharp$, the concrete $F^\natural \in \mathcal{P}^\natural \rightarrowtail \mathcal{P}^\natural$ and abstract $F^\sharp \in \mathcal{P}^\sharp \rightarrowtail \mathcal{P}^\sharp$ semantic functions such that $\forall c \in \mathcal{P}^\natural$: $\forall a \in \mathcal{P}^\sharp$: $\alpha(c) \preceq^\sharp a \Rightarrow \alpha(F^\natural(c)) \preceq^\sharp F^\sharp(a)$ (when $F^\natural(c)$ and $F^\sharp(a)$ are well-defined), the abstract upper approximation $a \in \mathcal{P}^\sharp$ of the fixpoint $c = F^\natural(c)$ of $F^\natural$ such that $\alpha(c) \preceq^\sharp a$ and the narrowing $\triangle^\sharp \in \wp(\mathcal{P}^\sharp) \rightarrowtail \mathcal{P}^\sharp$ satisfying (4.43) then the abstract iteration sequence with narrowing (4.42) is such that if $F^{\sharp\downarrow\beta}$ is well-defined for all $\beta \leq \lambda$ then $\alpha(c) \preceq^\sharp F^{\sharp\downarrow\lambda}$.

Proof. The proof is by transfinite induction on λ. For the basis we have $\alpha(c) \preceq^\sharp a = F^{\sharp\downarrow 0}$. Assume that $\alpha(c) \preceq^\sharp F^{\sharp\downarrow\lambda}$ and $F^{\sharp\downarrow\lambda+1}$ is well-defined. Then $\alpha(c) = \alpha(F^\natural(c)) \preceq^\sharp F^\sharp(\alpha(c)) \preceq^\sharp F^\sharp(F^{\sharp\downarrow\lambda}) \preceq^\sharp F^{\sharp\downarrow\lambda+1}$ by (4.42) and (4.43). If $\forall \beta < \lambda : \alpha(c) \preceq^\sharp F^{\sharp\downarrow\beta}$ then $\alpha(c) \preceq^\sharp \triangle^\sharp_{\beta<\lambda} F^{\sharp\downarrow\beta} = F^{\sharp\downarrow\lambda}$ for limit ordinals $\lambda > 0$ by (4.43) when this is well-defined. $\blacksquare$

Observe that in Propositions 6.14 and 6.18, α formalizes a "static" approximation, prior to the elaboration of the abstract version $F^\sharp$ of $F^\natural$ whereas the widening $\triangledown^\sharp$ and narrowing $\triangle^\sharp$ formalize a further "dynamic" approximation during the computation of the abstract iteration sequences $F^{\sharp\uparrow\lambda}$ and $F^{\sharp\downarrow\lambda}$ for $\lambda \in \mathit{Ord}$.

When starting from a post-fixpoint the abstract iteration sequence with narrowing (4.42) is decreasing and remains above any fixpoint of $F^\sharp$:

Proposition 6.19 (Convergence of the semantic approximation from a post-fixpoint using a narrowing)
Let $\langle \mathcal{P}^\sharp;\ \preceq^\sharp \rangle$ be the partially ordered set of abstract properties, $\triangle^\sharp$ be a narrowing in $\mathcal{P}^\sharp$ satisfying (4.44) and $F^{\sharp\downarrow\lambda}$ for $\lambda \in \mathit{Ord}$ be the abstract iteration sequence with narrowing (4.42) starting from $a \in \mathcal{P}^\sharp$ such that $F^\sharp(a) \preceq^\sharp a$.
1°) $F^{\sharp\downarrow\lambda}$, $\lambda \in \mathit{Ord}$ is decreasing piecewise, that is $F^\sharp(F^{\sharp\downarrow\lambda}) \preceq^\sharp F^{\sharp\downarrow\lambda+1} \preceq^\sharp F^{\sharp\downarrow\lambda}$ and $\exists \beta < \lambda : F^{\sharp\downarrow\lambda} \preceq^\sharp F^{\sharp\downarrow\beta}$ for limit ordinals $\lambda > 0$.
2°) If $\triangle^\sharp$ is the partially defined greatest lower bound $\sqcap^\sharp$ in $\mathcal{P}^\sharp$ and the sequence $F^{\sharp\downarrow\lambda}$, $\lambda \in \mathit{Ord}$ is total then it is decreasing and convergent. Its limit $F^{\sharp\downarrow\epsilon}$ is a fixpoint of $F^\sharp$.
3°) If, moreover, $F^\sharp$ is monotonic then $F^{\sharp\downarrow\epsilon}$ is the greatest fixpoint of $F^\sharp$ less than or equal to a.

Proof. 1°) If $F^{\sharp\downarrow\lambda}$ is a post-fixpoint of $F^\sharp$ then the greatest lower bound of $F^{\sharp\downarrow\lambda}$ and $F^\sharp(F^{\sharp\downarrow\lambda})$ is $F^\sharp(F^{\sharp\downarrow\lambda})$ so that by (4.44), we have $F^\sharp(F^{\sharp\downarrow\lambda}) \preceq^\sharp F^{\sharp\downarrow\lambda} \triangle^\sharp F^\sharp(F^{\sharp\downarrow\lambda})$ which is equal to $F^{\sharp\downarrow\lambda+1}$ if well-defined. Moreover, we must either have $F^{\sharp\downarrow\lambda+1} \preceq^\sharp F^{\sharp\downarrow\lambda}$ or else $F^{\sharp\downarrow\lambda+1} \preceq^\sharp F^\sharp(F^{\sharp\downarrow\lambda})$ in which case equality holds since $\preceq^\sharp$ is antisymmetric and therefore $F^{\sharp\downarrow\lambda+1} \preceq^\sharp F^{\sharp\downarrow\lambda}$ by post-fixpoint property. For limit ordinals $\lambda > 0$, we have $F^{\sharp\downarrow\lambda} = \triangle^\sharp_{\beta<\lambda} F^{\sharp\downarrow\beta}$ whence by (4.44) $\exists \beta < \lambda : F^{\sharp\downarrow\lambda} \preceq^\sharp F^{\sharp\downarrow\beta}$.

2°) For limit ordinals $\lambda > 0$, we have $F^{\sharp\downarrow\lambda} = \sqcap^\sharp_{\beta<\lambda} F^{\sharp\downarrow\beta}$ hence $\forall \beta < \lambda : F^{\sharp\downarrow\lambda} \preceq^\sharp F^{\sharp\downarrow\beta}$ when the greatest lower bound is well-defined. By transfinite induction using 1°), we conclude that $F^{\sharp\downarrow\lambda}$, $\lambda \in \mathit{Ord}$ is a decreasing chain. The sequence is assumed to be total hence it is ultimately stationary since its cardinality is less than or equal to that of $\mathcal{P}^\sharp$ and $\preceq^\sharp$ is antisymmetric. Its limit $F^{\sharp\downarrow\epsilon}$ is a fixpoint of $F^\sharp$ since $F^\sharp(F^{\sharp\downarrow\epsilon}) \preceq^\sharp F^{\sharp\downarrow\epsilon+1} = F^{\sharp\downarrow\epsilon}$ by 1°) hence $F^{\sharp\downarrow\epsilon} = F^{\sharp\downarrow\epsilon+1} = F^{\sharp\downarrow\epsilon} \sqcap^\sharp F^\sharp(F^{\sharp\downarrow\epsilon}) = F^\sharp(F^{\sharp\downarrow\epsilon})$.

3) Let x be a fixpoint of $F^\sharp$ less than or equal to $a = F^{\sharp\downarrow 0}$. Assume by induction hypothesis that $x \preceq^\sharp F^{\sharp\downarrow\lambda} \preceq^\sharp a$. Then by monotony and 1), $x = F^\sharp(x) \preceq^\sharp F^\sharp(F^{\sharp\downarrow\lambda}) \preceq^\sharp F^{\sharp\downarrow\lambda+1} \preceq^\sharp F^{\sharp\downarrow\lambda} \preceq^\sharp a$. The same way, if $\lambda > 0$ is a limit ordinal and $\forall\beta < \lambda : x \preceq^\sharp F^{\sharp\downarrow\beta} \preceq^\sharp a$ then $x \preceq^\sharp \sqcap^\sharp_{\beta<\lambda} F^{\sharp\downarrow\beta} \preceq^\sharp a$ by definition of greatest lower bounds. Hence $\forall\lambda \in \mathit{Ord} : x \preceq^\sharp F^{\sharp\downarrow\lambda} \preceq^\sharp a$ which holds in particular for $\lambda = \epsilon$. ∎

The use of an increasing (upper) iteration to a post-fixpoint followed by a decreasing (upper) iteration to a fixpoint is a natural method to reach fixpoints of monotonic functions (or upper approximations of these), as shown by Propositions 6.17 and 6.19 and their duals, which is the basis to the constructive version of Tarski's fixpoint proposition [11]. The usual iteration from the infimum $\perp$ to the least fixpoint is a degenerate case (where the decreasing iteration is constant).

An often misunderstood point is that the narrowing $\vartriangle$ is not the dual of a widening $\triangledown$. A widening is used to approximate an increasing iteration from above. Its dual would approximate a decreasing sequence from below whereas the narrowing is used to approximate a decreasing sequence from above, so as not to overshoot the approximated and unknown fixpoint. As usual, we do not state the dual of propositions, but in doing so we should follow [18] and distinguish four operators $\triangledown = \overline{\triangledown}$, $\vartriangle = \overline{\vartriangle}$ for approximations from above and $\underline{\triangledown}$, $\underline{\vartriangle}$ for approximations from below (which, for example, would be useful for model checking).

Termination. In practice, automated abstract interpretations must terminate. Therefore a fifth basic choice in the design of an abstract interpretation is the method ensuring termination of the abstract interpreter. Therefore an additional hypothesis is:

The abstract iteration sequences (4.36) and (4.42) are eventually stable after finitely many steps. (4.45)

In this case hypotheses such as (4.40) to handle transfinite abstract iteration sequences are useless and A can be assumed to be finite in (4.39), (4.43) and (4.44). The resort to finite domains $\mathcal{P}^\sharp$ of abstract properties, or to the non-existence of strictly increasing chains is not always satisfactory since termination must also be rapid. A widening/narrowing approach can be used to enforce quick termination as in [6, 7]. In case of finite termination, Propositions 6.14 and 6.18 can be made more specific as follows:

Proposition 6.20 (Finite semantic approximation using an abstraction function and a widening)
Given the sets $\mathcal{P}^\natural$ of concrete properties and $\mathcal{P}^\sharp$ of abstract properties with approximation pre-order $\preceq^\sharp$ and computational partial order $\sqsubseteq^\sharp$, the abstraction function $\alpha \in \mathcal{P}^\natural \mapsto \mathcal{P}^\sharp$, the bases $\underset{\cdot}{\perp}^\natural \in \mathcal{P}^\natural$ and $\underset{\cdot}{\perp}^\sharp \in \mathcal{P}^\sharp$, the concrete $F^\natural \in \mathcal{P}^\natural \rightarrowtail \mathcal{P}^\natural$ and abstract $F^\sharp \in \mathcal{P}^\sharp \mapsto \mathcal{P}^\sharp$ semantic functions, the concrete inductive join $\amalg^\natural \in \wp(\mathcal{P}^\natural) \rightarrowtail$

$\mathcal{P}^\natural$, the widening $\triangledown^\sharp \in \wp(\mathcal{P}^\sharp \times \mathcal{P}^\sharp) \mapsto \mathcal{P}^\sharp$ such that:

- $\alpha(\perp^\natural) \preceq^\sharp \perp^\sharp$,
- $\forall c \in \mathcal{P}^\natural$: $\forall a \in \mathcal{P}^\sharp$: $\alpha(c) \preceq^\sharp a \Rightarrow \alpha(F^\natural(c)) \preceq^\sharp F^\sharp(a)$ (when $F^\natural(c)$ is well-defined),
- $(\forall i \in I : \alpha(c_i) \preceq^\sharp a) \;\Rightarrow\; \alpha(\coprod^\natural_{i \in I} c_i) \preceq^\sharp a$ (when $\coprod^\natural_{i \in I} c_i$ is well-defined),
- $\forall a, a' \in \mathcal{P}^\sharp : a \sqsubseteq^\sharp a \triangledown^\sharp a'$,[13]
- $\forall c \in \mathcal{P}^\natural : \forall a, a' \in \mathcal{P}^\sharp : \alpha(c) \preceq^\sharp a' \;\Rightarrow\; \alpha(c) \preceq^\sharp a \triangledown^\sharp a'$,[13]
- For every $\mathbb{N}$-termed sequence $x_0, \ldots, x_i, \ldots$ in $\mathcal{P}^\sharp$, the chain $y_0 = x_0 \sqsubseteq^\sharp \ldots \sqsubseteq^\sharp y_{i+1} = y_i \triangledown^\sharp x_i \sqsubseteq^\sharp \ldots$ is not strictly increasing,[14]

(4.46)

and assuming that the concrete iteration sequence (4.1) is convergent with limit $F^{\natural\epsilon}$ then the abstract iteration sequence with widening:

$$\begin{cases} F^{\sharp\uparrow 0} \stackrel{\text{def}}{=} \perp^\sharp \\ F^{\sharp\uparrow i+1} \stackrel{\text{def}}{=} F^{\sharp\uparrow i} \triangledown^\sharp F^\sharp(F^{\sharp\uparrow i}) \qquad i \in \mathbb{N} \end{cases} \tag{4.47}$$

is eventually stable after $\ell \in \mathbb{N}$ steps and $\alpha(F^{\natural\epsilon}) \preceq^\sharp F^{\sharp\uparrow\ell}$.

Proof. The sequence $F^{\sharp\uparrow i}$, $i \in \mathbb{N}$ is increasing since $F^{\sharp\uparrow i} \sqsubseteq^\sharp F^{\sharp\uparrow i} \triangledown^\sharp F^\sharp(F^{\sharp\uparrow i}) = F^{\sharp\uparrow i+1}$ but not strictly whence there exists a smallest $\ell \in \mathbb{N}$ such that $F^{\sharp\uparrow\ell} = F^{\sharp\uparrow\ell+1}$ because $\sqsubseteq^\sharp$ is antisymmetric. It $\ell \leq k$ and $F^{\sharp\uparrow\ell} = F^{\sharp\uparrow k}$ then $F^{\sharp\uparrow\ell} = F^{\sharp\uparrow\ell+1} = F^{\sharp\uparrow\ell} \triangledown^\sharp F^\sharp(F^{\sharp\uparrow\ell}) = F^{\sharp\uparrow k} \triangledown^\sharp F^\sharp(F^{\sharp\uparrow k}) = F^{\sharp\uparrow k+1}$ proving stability after ℓ steps.

By (4.1) we have $\alpha(F^{\natural 0}) = \alpha(\perp^\natural) \preceq^\sharp \perp^\sharp = F^{\sharp\uparrow 0}$. If $\alpha(F^{\natural i}) \preceq^\sharp F^{\sharp\uparrow i}$ then $\alpha(F^\natural(F^{\natural i})) \preceq^\sharp F^\sharp(F^{\sharp\uparrow i})$ and therefore $\alpha(F^{\natural i+1}) \preceq^\sharp F^{\sharp\uparrow i} \triangledown^\sharp F^\sharp(F^{\sharp\uparrow i}) = F^{\sharp\uparrow i+1}$. By recurrence on i, we have in particular $\alpha(F^{\natural\ell}) \preceq^\sharp F^{\sharp\uparrow\ell}$. Moreover, if $\alpha(F^{\natural\lambda}) \preceq^\sharp F^{\sharp\uparrow\ell}$ then $\alpha(F^\natural(F^{\natural\lambda})) \preceq^\sharp F^\sharp(F^{\sharp\uparrow\ell})$ and therefore $\alpha(F^{\natural\lambda+1}) \preceq^\sharp F^{\sharp\uparrow\ell} \triangledown^\sharp F^\sharp(F^{\sharp\uparrow\ell}) = F^{\sharp\uparrow\ell}$. Finally if $\alpha(F^{\natural\beta}) \preceq^\sharp F^{\sharp\uparrow\ell}$ for all $\beta < \lambda$ then $\alpha\big(\coprod^\natural_{\beta<\lambda} F^{\natural\beta}\big) \preceq^\sharp F^{\sharp\uparrow\ell}$ so that by transfinite induction we have $\alpha(F^{\natural\epsilon}) = \alpha(F^{\natural\mu}) \preceq^\sharp F^{\sharp\uparrow\ell}$ for the maximum μ of ϵ an ℓ. ∎

Proposition 6.21 (Finite semantic approximation using an abstraction function and a narrowing)
Given the sets $\mathcal{P}^\natural$ of concrete properties and $\mathcal{P}^\sharp$ of abstract properties with approximation pre-order $\preceq^\sharp$ and computational partial order $\sqsubseteq^\sharp$, the abstraction function $\alpha \in \mathcal{P}^\natural \mapsto \mathcal{P}^\sharp$, the concrete $F^\natural \in \mathcal{P}^\natural \rightarrowtail \mathcal{P}^\natural$ and abstract $F^\sharp \in \mathcal{P}^\sharp \mapsto \mathcal{P}^\sharp$ semantic functions, the approximation $a \in \mathcal{P}^\sharp$ of the fixpoint $c \in \mathcal{P}^\natural$ of $F^\natural$, the narrowing $\vartriangle^\sharp \in \wp(\mathcal{P}^\sharp \times \mathcal{P}^\sharp) \mapsto \mathcal{P}^\sharp$ such that:

- $F^\natural(c) = c$ and $\alpha(c) \preceq^\sharp a$,
- $\forall a \in \mathcal{P}^\sharp$: $\alpha(c) \preceq^\sharp a \Rightarrow \alpha(F^\natural(c)) \preceq^\sharp F^\sharp(a)$,
- $\forall a, a' \in \mathcal{P}^\sharp : a \vartriangle^\sharp a' \sqsubseteq^\sharp a$,[15]
- $\forall c \in \mathcal{P}^\natural : \forall a, a' \in \mathcal{P}^\sharp : \alpha(c) \preceq^\sharp a \wedge \alpha(c) \preceq^\sharp a' \;\Rightarrow\; \alpha(c) \preceq^\sharp a \vartriangle^\sharp a'$,[15]
- For every $\mathbb{N}$-termed sequence $x_0, \ldots, x_i, \ldots$ in $\mathcal{P}^\sharp$, the chain $y_0 = x_0 \sqsupseteq^\sharp \ldots \sqsupseteq^\sharp y_{i+1} = y_i \vartriangle^\sharp x_i \sqsupseteq^\sharp \ldots$ is not strictly decreasing,[16]

(4.48)

[13] if $F^\sharp$ is monotonic for $\sqsubseteq^\sharp$ then we can assume that $a \sqsubseteq^\sharp a'$ and that $a \triangledown^\sharp a'$ is well-defined only in that case.

[14] if $F^\sharp$ is monotonic for $\sqsubseteq^\sharp$ then we can assume that x_i, $i \in \mathbb{N}$ is an increasing chain for $\sqsubseteq^\sharp$.

then the abstract iteration sequence with narrowing:

$$\left\{\begin{array}{rcll} F^{\sharp\downarrow 0} & \stackrel{\text{def}}{=} & a & \\ F^{\sharp\downarrow i+1} & \stackrel{\text{def}}{=} & F^{\sharp\downarrow i} \mathbin{\triangle^\sharp} F^\sharp(F^{\sharp\downarrow i}) & i \in \mathbb{N} \end{array}\right. \tag{4.49}$$

is eventually stable after $\ell \in \mathbb{N}$ steps and such that $\alpha(c) \preceq^\sharp F^{\sharp\downarrow i}$ for all $i \in \mathbb{N}$.

Proof. The sequence $F^{\sharp\downarrow i}$, $i \in \mathbb{N}$ is decreasing since $F^{\sharp\downarrow i} \sqsupseteq^\sharp F^{\sharp\downarrow i} \mathbin{\triangle^\sharp} F^\sharp(F^{\sharp\downarrow i}) = F^{\sharp\downarrow i+1}$ but not strictly whence there exists a smallest $\ell \in \mathbb{N}$ such that $F^{\sharp\downarrow \ell} = F^{\sharp\downarrow \ell+1}$ because $\sqsubseteq^\sharp$ is antisymmetric. If $\ell \leq k$ and $F^{\sharp\downarrow \ell} = F^{\sharp\downarrow k}$ then $F^{\sharp\downarrow \ell} = F^{\sharp\downarrow \ell+1} = F^{\sharp\downarrow \ell} \mathbin{\triangle^\sharp} F^\sharp(F^{\sharp\downarrow \ell}) = F^{\sharp\downarrow k} \mathbin{\triangle^\sharp} F^\sharp(F^{\sharp\downarrow k}) = F^{\sharp\downarrow k+1}$ proving stability after ℓ steps.

We have $\alpha(c) \preceq^\sharp a = F^{\sharp\downarrow 0}$. If $\alpha(c) \preceq^\sharp F^{\sharp\downarrow i}$ then $\alpha(F^\natural(c)) \preceq^\sharp F^\sharp(F^{\sharp\downarrow i})$ and therefore $\alpha(c) \preceq^\sharp F^{\sharp\downarrow i} \mathbin{\triangle^\sharp} F^\sharp(F^{\sharp\downarrow i}) = F^{\sharp\downarrow i+1}$. By recurrence, we have $\alpha(c) \preceq^\sharp F^{\sharp\downarrow i}$ for all $i \in \mathbb{N}$. ∎

7 Concretization correspondence between concrete and abstract semantics

We have studied the case when the connection between the concrete and abstract semantics is established via an abstraction relation or function α and the notion of precision is formalized using an approximation relation $\preceq^\sharp$ on the abstract properties. We now shortly examine the case when the connection between the concrete and abstract semantics is established via a concretization relation or function γ and the notion of precision is formalized using an approximation relation $\preceq^\natural$ on the concrete properties:

$$\begin{array}{rcll} \preceq^\natural & \in & \wp(\mathcal{P}^\natural \times \mathcal{P}^\natural) & \text{is a pre-order,} \\ c \approx^\natural c' & \stackrel{\text{def}}{=} & (c \preceq^\natural c') \wedge (c' \preceq^\natural c) & \end{array} \tag{4.50}$$

Both cases should be considered since in practice one way may be easier than the other.

Example 7.1 (Invariance)
Let S be the set of program states, S^ω be the set of infinite sequences τ of states of length $|\tau| = \omega$, $S^\star$ be the set of finite sequences τ of states of length $|\tau| \in \mathbb{N}$, $S^\infty = S^\star \cup S^\omega$. We write τ_i for the i^{th} state in τ counting from 0. The semantics of a program is a set of finite or infinite execution traces so that $\mathcal{P}^\natural = \wp(S^\infty)$. A property is a set of states $\mathcal{P}^\sharp = \wp(S)$ which characterizes the states reachable during program execution so that $\alpha \in \mathcal{P}^\natural \mapsto \mathcal{P}^\sharp$ is defined by $\alpha(T) \stackrel{\text{def}}{=} \bigcup_{\tau \in T} \alpha(\tau)$ where $\alpha \in S^\infty \mapsto \mathcal{P}^\sharp$ is defined by $\alpha(\tau) \stackrel{\text{def}}{=} \{\tau_i \mid 0 \leq i < |\tau|\}$.

The *invariance* soundness relation $\sigma_{\mathcal{I}} \in \wp(\mathcal{P}^\natural \times \mathcal{P}^\sharp)$ is:

$$\sigma_{\mathcal{I}} \stackrel{\text{def}}{=} \{\langle T,\, P\rangle \mid \forall \tau \in T : \forall i < |\tau| : \tau_i \in P\} \quad .$$

Observe that $\sigma_{\mathcal{I}}$ satisfies $\forall T \in \mathcal{P}^\natural : \forall P, P' \in \mathcal{P}^\sharp : (\langle T,\, P\rangle \in \sigma_{\mathcal{I}} \wedge P \subseteq P') \Rightarrow \langle T,\, P'\rangle \in \sigma_{\mathcal{I}}$, that is (4.19), where the abstract approximation relation is defined as $\preceq^\sharp \stackrel{\text{def}}{=} \subseteq$.

[15] if $F^\sharp$ is monotonic for $\sqsubseteq^\sharp$ then we can assume that $a' \sqsubseteq^\sharp a$ and that $a \mathbin{\triangle^\sharp} a'$ is well-defined only in that case.

[16] if $F^\sharp$ is monotonic for $\sqsubseteq^\sharp$ then we can assume that x_i, $i \in \mathbb{N}$ is a decreasing chain for $\sqsubseteq^\sharp$.

Concepts	
concrete	abstract
concretization	abstraction
more precise	less precise
is approximated by	approximates
widening	narrowing
minimal	maximal
monotonic	monotonic
increasing	decreasing
least fixpoint	greatest fixpoint
reductive	extensive

Galois connections	
$\langle S;\ \sqsubseteq\rangle \overset{\gamma}{\underset{\alpha}{\leftrightarrows}} \langle S';\ \sqsubseteq'\rangle$	$\langle S';\ \sqsupseteq'\rangle \overset{\alpha}{\underset{\gamma}{\leftrightarrows}} \langle S;\ \sqsupseteq\rangle$

Sets	
$\mathcal{P}^\natural$	$\mathcal{P}^\sharp$
C	A

Set elements	
c	a
$\perp\!\!\!\!\perp^\natural$	$\perp\!\!\!\!\perp^\sharp$
$\bot^\natural$	$\top^\sharp$
$\top^\natural$	$\bot^\sharp$
$\mathit{lfp}^{\sqsubseteq^\natural}_{\bot^\natural}\ F^\natural$	$\mathit{gfp}^{\sqsubseteq^\sharp}_{\top^\sharp}\ F^\sharp$

Functions	
α	γ
$F^\natural$	$F^\sharp$
$\amalg^\natural$	$\amalg^\sharp$

$\nabla^\natural$	$\triangle^\sharp$
$\triangle^\natural$	$\nabla^\sharp$
$\sqcup^\natural$	$\sqcap^\sharp$
$\sqcap^\natural$	$\sqcup^\sharp$

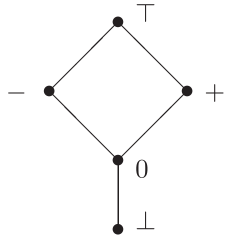
Relations	
ρ	ρ^{-1}
σ	σ^{-1}
α	γ
γ	α
$\preceq^\natural$	$\succeq^\sharp$
$\succeq^\natural$	$\preceq^\sharp$
$\approx^\natural$	$\approx^\sharp$
$\sqsubseteq^\natural$	$\sqsupseteq^\sharp$
$\sqsupseteq^\natural$	$\sqsubseteq^\sharp$

Figure 1: Dual abstract interpretations

The concrete approximation relation is defined as $T \preceq^\natural T' \stackrel{\text{def}}{=} \alpha(T) \subseteq \alpha(T')$, that is $\forall \tau \in T : \forall i \in |\tau| : \exists \tau' \in T' : \exists j \in |\tau'| : \tau_i = {\tau'}_j$, so that the approximation relation is easier to define in the abstract domain.

Example 7.2 (Signs)
Let $s[\mathtt{x}]$ be the value of variable $\mathtt{x}$ in state $s \in S$, $\mathcal{P}^\natural \stackrel{\text{def}}{=} \wp(S)$, $\mathcal{P}^\sharp \stackrel{\text{def}}{=} \{\bot, 0, +, -, \top\}$ and $\gamma \in \mathcal{P}^\sharp \mapsto \mathcal{P}^\natural$ be such that:

a	$\gamma(a)$
$\bot$	$\emptyset$
0	$\{s \in S \mid s[\mathtt{x}] = 0\}$
$+$	$\{s \in S \mid s[\mathtt{x}] \geq 0\}$
$-$	$\{s \in S \mid s[\mathtt{x}] \leq 0\}$
$\top$	S



The concrete approximation relation is $\preceq^\natural \stackrel{\text{def}}{=} \subseteq$ whereas the abstract approximation relation $\preceq^\sharp$ is defined by $a \preceq^\sharp a' \stackrel{\text{def}}{=} \gamma(a) \preceq^\natural \gamma(a')$. A direct definition is not so easy since it involves the description of the above Hasse diagram by cases.

Dual abstract interpretations. Abstract interpretation frameworks based upon a concrete approximation relation $\preceq^\natural$ and a concretization relation or function γ can be obtained by duality from the abstract interpretation frameworks considered in Section 6 which were based upon an abstract approximation relation $\preceq^\sharp$ and an abstraction relation or function α. Duality of abstract interpretations is defined in Fig. 1. Observe that it differs from order-theoretic duality where $\sqsubseteq$, $\bot$, $\top$, $\sqcup$ and $\sqcap$ would be respectively replaced by $\sqsupseteq$, $\top$, $\bot$, $\sqcap$ and $\sqcup$ but γ, α, σ, etc. would be left unchanged. We will often consider the order-theoretic dual of a dual abstract interpretation so as, for example, to stick with least fixpoints but to use concretization instead of abstraction relations or functions. For example the dual of (4.1) is (4.2). The dual of (4.3) is $\sigma^{-1} \in \wp(\mathcal{P}^\sharp \times \mathcal{P}^\natural)$, that is $\sigma \in \wp(\mathcal{P}^\natural \times \mathcal{P}^\sharp)$, so that (4.3) is self-dual. The dual of the existence of abstract approximations assumption (4.5) is $\forall a \in \mathcal{P}^\sharp : \exists c \in \mathcal{P}^\natural : \langle a,\ c\rangle \in \sigma^{-1}$, that is the *usefulness of abstract properties assumption*:

$$\forall a \in \mathcal{P}^\sharp : \exists c \in \mathcal{P}^\natural : \langle c,\ a\rangle \in \sigma\ . \tag{4.51}$$

This assumption states that an abstract property a such that $\left\{c \in \mathcal{P}^\natural \,\middle|\, \langle c,\ a\rangle \in \sigma\right\}$ is empty is useless in the sense that it can be used to approximate no concrete property and therefore can be eliminated from $\mathcal{P}^\sharp$ without harm, that is without losing expressive power with respect to the given concrete properties $\mathcal{P}^\natural$. For example in strictness analysis non-monotonic abstract functions such as $\varphi(0) = 1$ and $\varphi(1) = 0$ are useless. Proposition 4.3 is self-dual since concrete and abstract properties are treated on terms of equality when their correspondence is formalized using a relation. In general the meaning $\{c \,|\, \langle c,\ a\rangle \in \sigma\}$ of an abstract property a has no or multiple maximal elements. In these circumstances the dual of Proposition 4.5 specifies the inducing of the concrete iterates from the abstract iterates by discriminating among all sound possibilities. It uses concrete narrowing $\triangle^\natural$ and widening $\triangledown^\natural$ to respectively mitigate the absence of greatest lower bounds and least upper bounds for $\preceq^\natural$ in $\mathcal{P}^\natural$. In general however, the concrete semantics is a given collecting semantics which is intended to be more precise than its abstraction in which case one uses the dual of Section 6.

Concretization relation between the concrete and abstract properties. One can start by introducing a *concretization relation*:

$$\gamma \in \wp(\mathcal{P}^\sharp \times \mathcal{P}^\natural) \tag{4.52}$$

specifying which concrete properties can be safely approximated by abstract properties, together with a pre-order $\preceq^\natural$ on $\mathcal{P}^\natural$ to specify the relative precision of concrete properties. Afterwards, the soundness relation $\sigma \in \wp(\mathcal{P}^\natural \times \mathcal{P}^\sharp)$ can be defined according to the following *concretization based soundness assumption*:[17]

$$\sigma = \left\{\langle c,\ a\rangle \,\middle|\, \exists c' \in \mathcal{P}^\natural : \langle a,\ c'\rangle \in \gamma\ \wedge\ c \preceq^\natural c'\right\}\ . \tag{4.53}$$

The relationship between $\preceq^\natural$ and σ is that"more precise concrete properties are approximated by greater sets of abstract properties": for all c, c' in $\mathcal{P}^\natural$, $c \preceq^\natural c'$ implies $\{a' \,|\, \langle c',\ a'\rangle \in \sigma\} \subseteq \{a \,|\, \langle c,\ a\rangle \in \sigma\}$ by (4.53). Equations (4.50) and (4.53) imply:

$$\forall c,\ c' \in \mathcal{P}^\natural : \forall a \in \mathcal{P}^\sharp : (\langle c,\ a\rangle \in \sigma\ \wedge\ c' \preceq^\natural c)\ \Rightarrow\ \langle c',\ a\rangle \in \sigma\ . \tag{4.54}$$

Redundancy between γ and σ can sometimes be avoided by requiring γ to select the set of maximal concrete properties c in relation with a by the soundness relation σ. In this case γ satisfies the *concrete maximality assumption*:

$$\gamma = \left\{\langle a,\ c\rangle \in \sigma^{-1} \,\middle|\, \forall c' \in \mathcal{P}^\natural : (\langle c',\ a\rangle \in \sigma \wedge c \preceq^\natural c')\ \Rightarrow\ (c' \preceq^\natural c)\right\}\ . \tag{4.55}$$

However, $\left\{c \,\middle|\, \langle a,\ c\rangle \in \sigma^{-1}\right\}$ may contain infinite chains strictly increasing for $\preceq^\natural$, in which case (4.55) would define γ as empty.

[17] Observe that the set $\{c \,|\, \langle c,\ a\rangle \in \sigma\}$ of concrete properties which can be approximated by a given abstract property a is downwards closed. This remark leads to ideal-based abstract interpretation frameworks as considered in paragraph 6.4 of [12].

(4.1)	(4.3)	(4.5)	(4.12)	(4.14)	(4.18)	(4.19)	(4.20)	(4.21)	(4.23)	(4.25)	(4.26)
(4.2)	(4.3)	(4.51)	(4.12)	(4.14)	(4.50)	(4.54)	(4.52)	(4.53)	(4.55)	(4.56)	(4.57)

Figure 2: Dual definitions and assumptions

Concretization function from the concrete into the abstract properties. When the abstract properties have best meanings:

$$\forall a \in \mathcal{P}^\sharp : \exists c \in \mathcal{P}^\natural : \langle c,\, a\rangle \in \sigma \;\wedge\; \forall c' \in \mathcal{P}^\natural : \langle c',\, a\rangle \in \sigma \;\Rightarrow\; c' \preceq^\natural c \ . \tag{4.56}$$

we can define γ by (4.55) so that, by the dual of Proposition 6.2, it is a concretization function $\gamma \in \mathcal{P}^\sharp \mapsto \mathcal{P}^\natural$ whenever $\preceq^\natural$ is a partial ordering. The notion of soundness can be defined with respect to concrete properties as follows:

$$\forall a \in \mathcal{P}^\sharp : \forall c \in \mathcal{P}^\natural : \langle c,\, a\rangle \in \sigma \;\Leftrightarrow\; c \preceq^\natural \gamma(a) \ . \tag{4.57}$$

The duality of the abstraction and concretization based abstract interpretation frameworks is shown in Fig. 2.

The duals of Propositions 6.8 and 6.10 respectively define the inducing and the approximation of the concrete iterates into the abstract domain using a concretization function. Observe that the dual of corollary 6.9 of Proposition 6.8 involves greatest fixpoints on the cpo $\langle \mathcal{P}^\sharp;\ \sqsupseteq^\sharp,\, \top^\sharp,\, \sqcap^\sharp\rangle$ therefore, as usual, it should be used in its order-theoretic dual form for least fixpoints on the cpo $\langle \mathcal{P}^\sharp;\ \sqsubseteq^\natural,\, \perp^\sharp,\, \sqcup^\sharp\rangle$ to induce a concrete form of abstract fixpoints from a cpo by abstraction/concretization functions.

On the inducing of the abstract iterates using a concretization function. The use of a concretization function is quite similar to that of an abstraction function for designing the abstract iterates. In particular the soundness of widenings and narrowings established in Propositions 6.14 and 6.18 can also be stated using a concretization function, thanks to the following dual of Proposition 6.10:

Proposition 7.3 (Semantic approximation using a concretization function)
Given the sets $\langle \mathcal{P}^\natural;\ \preceq^\natural\rangle$ of concrete properties with pre-order $\preceq^\natural$ and $\mathcal{P}^\sharp$ of abstract properties, the concretization function $\gamma \in \mathcal{P}^\sharp \mapsto \mathcal{P}^\natural$, the basis $\underline{\perp}^\sharp$ such that $F^{\natural 0} \preceq^\natural \gamma(\underline{\perp}^\sharp)$, the abstract semantic function $F^\sharp \in \mathcal{P}^\sharp \rightarrowtail \mathcal{P}^\sharp$ such that $F^{\natural\lambda} \preceq^\natural \gamma(F^{\sharp\lambda}) \Rightarrow F^{\natural\lambda+1} \preceq^\natural \gamma(F^\sharp(F^{\sharp\lambda}))$ for all $\lambda \in \mathit{Ord}$, the abstract inductive join $\amalg^\sharp \in \wp(\mathcal{P}^\sharp) \rightarrowtail \mathcal{P}^\sharp$ such that $\forall \beta < \lambda : F^{\natural\beta} \preceq^\natural \gamma(F^{\sharp\beta}) \Rightarrow F^{\natural\lambda} \preceq^\natural \gamma(\amalg^\sharp_{\beta<\lambda} F^{\sharp\beta})$ for all limit ordinals $\lambda > 0$ and assuming that the concrete and abstract iteration sequences are convergent then their respective limits $F^{\natural\varepsilon}$ and $F^{\sharp\epsilon}$ are such that $F^{\natural\epsilon} \preceq^\natural \gamma(F^{\sharp\varepsilon})$.

Observe that if we are interested in using the widening and narrowing operators on the abstract domain (and not on the dual concrete domain as in dual propositions) then Propositions 6.14, 6.17, 6.18 and 6.19 must be reformulated using a concretization instead of an abstraction function. For short, this consists essentially in replacing the soundness condition $\alpha(c) \preceq^\sharp a$ by its substitute $c \preceq^\natural \gamma(a)$.

8 Abstraction/concretization correspondence between concrete and abstract semantics

We now examine the more symmetrical situation in which the notion of precision is formalized on both concrete and abstract properties using the approximation relations $\preceq^\natural$ and $\preceq^\sharp$ and the connection between the concrete and abstract semantics is established via a pair of abstraction and concretization functions. In particular, $\langle\mathcal{P}^\natural;\ \preceq^\natural\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle\mathcal{P}^\sharp;\ \preceq^\sharp\rangle$ may be a Galois connection.

Weak abstraction/concretization connection with concrete approximation pre-order. A simple way to weaken the classical abstract interpretation framework is to assume that the abstraction function α selects minimal abstract properties (but not necessarily the best ones). Moreover, the notion of approximation can be defined on the concrete semantic domain $\mathcal{P}^\natural$ by a pre-order $\preceq^\natural$ (4.50) (not necessarily a partial order). If the meaning of abstract properties has been defined by a concretization function γ then the soundness relation σ defined by (4.57) obviously satisfies the concrete maximality assumption (4.55). It is possible to define the abstract approximation relation $\preceq^\sharp$ as the mere extension of $\preceq^\natural$ onto $\mathcal{P}^\sharp$ through γ:

$$a \preceq^\sharp a' \ \stackrel{\text{def}}{=}\ \gamma(a) \preceq^\natural \gamma(a') \tag{4.58}$$

so that γ is monotonic by definition of $\preceq^\sharp$. This situation is characterized by the following:

Proposition 8.1 (Weak abstraction/concretization connection)
Let $\langle\mathcal{P}^\natural;\ \preceq^\natural\rangle$ by the set of concrete properties with pre-order $\preceq^\natural$, $\mathcal{P}^\sharp$ be the set of abstract properties, $\alpha \in \mathcal{P}^\natural \mapsto \mathcal{P}^\sharp$ be the abstraction and $\gamma \in \mathcal{P}^\sharp \mapsto \mathcal{P}^\natural$ be the concretization functions. Define the soundness relation $\sigma \in \wp(\mathcal{P}^\natural \times \mathcal{P}^\sharp)$ by (4.57). Define $\preceq^\sharp$ by (4.58). If α satisfies the abstract minimality assumption (4.23), we have:

1°) $\forall c \in \mathcal{P}^\natural : c \preceq^\natural \gamma(\alpha(c))$, which implies (4.5) in that $\forall c \in \mathcal{P}^\natural : \langle c,\ \alpha(c)\rangle \in \sigma$

2°) $\gamma \approx^\natural \gamma \circ \alpha \circ \gamma$

3°) $\forall c \in \mathcal{P}^\natural : \forall a \in \mathcal{P}^\sharp : \alpha(c) \preceq^\sharp a \ \Rightarrow\ c \preceq^\natural \gamma(a) \ \not\Rightarrow\ \alpha(c) \preceq^\sharp a$

4°) $\forall c,\ c' \in \mathcal{P}^\natural : c \preceq^\natural c' \ \not\Rightarrow\ \alpha(c) \preceq^\sharp \alpha(c')$

Proof. 1°) By (4.23), for all $c \in \mathcal{P}^\natural$, $\langle c,\ \alpha(c)\rangle$ belongs to σ so that (4.57) implies $c \preceq^\natural \gamma(\alpha(c))$ so that (4.5) is satisfied because α is total.
2°) For all $a \in \mathcal{P}^\sharp$, we have $\gamma(a) \preceq^\natural \gamma(\alpha(\gamma(a)))$ by 1°). Replacing σ and $\preceq^\sharp$ by their respective definitions (4.57) and (4.58) in (4.23), we get $\forall c \in \mathcal{P}^\natural : \forall a' \in \mathcal{P}^\sharp : (c \preceq^\natural \gamma(a') \wedge \gamma(a') \preceq^\natural \gamma(\alpha(c))) \ \Rightarrow\ \gamma(\alpha(c)) \preceq^\natural \gamma(a')$, so that for $c = \gamma(a)$ and $a' = a$, we get $\gamma(\alpha(\gamma(a))) \preceq^\natural \gamma(a)$. By definition (4.50) of $\approx^\natural$, we conclude that $\gamma(a) \approx^\natural \gamma(\alpha(\gamma(a)))$.
3°) If $\alpha(c) \preceq^\sharp a$ then $\gamma(\alpha(c)) \preceq^\natural \gamma(a)$ by (4.58) hence $c \preceq^\natural \gamma(a)$ by 1°) and transitivity. Define $\mathcal{P}^\sharp = \{x,\ y,\ z\}$ such that $x \preceq^\natural x \preceq^\natural z \preceq^\natural z$ and $x \preceq^\natural y \preceq^\natural y$, $\mathcal{P}^\natural = \{a,\ b\}$ such that $a \preceq^\sharp a$ and $b \preceq^\sharp b$, $\alpha(x) = b$, $\alpha(y) = b$, $\alpha(z) = a$, $\gamma(a) = z$, $\gamma(b) = y$. We have $\sigma = \{\langle x,\ a\rangle,\ \langle x,\ b\rangle,\ \langle y,\ b\rangle,\ \langle z,\ a\rangle\}$ so that $x \preceq^\natural \gamma(a)$ but $\alpha(x) \preceq^\sharp a$ is not true.
4°) Define $\mathcal{P}^\natural = \{x,\ y,\ z\}$ with $x \preceq^\natural x \preceq^\natural y \preceq^\natural y$ and $x \preceq^\natural z \preceq^\natural z$ so that $\preceq^\natural$ is a partial order relation and $\approx^\natural$ is equality. Define $\mathcal{P}^\sharp = \{a,\ b\}$, $\gamma(a) = y$ and $\gamma(b) = z$. a and b are not comparable since y and z are not comparable so that $\preceq^\sharp$ is equality. By (4.57), we have $\sigma = \big\{\langle x,\ a\rangle,\ \langle x,\ b\rangle,\ \langle y,\ a\rangle, \langle z,\ b\rangle\big\}$ so that $\alpha(x)$ can be a or b to

satisfy (4.23). Define $\alpha(x) = b$ so that $x \preceq^{\natural} y$ but $\alpha(x) = b$ and $\alpha(y) = a$ are not comparable by $\preceq^{\sharp}$. ∎

Let us define the *normalization operator* $\eta \in \mathcal{P}^{\sharp} \mapsto \mathcal{P}^{\sharp}$ by $\eta \overset{\text{def}}{=} \alpha \circ \gamma$. By Proposition 8.1–2°, we have $c \preceq^{\natural} \gamma(a) \Leftrightarrow c \preceq^{\natural} \gamma \circ \eta(a)$, $a \approx^{\sharp} \eta(a)$ and η is idempotent on $\mathcal{P}^{\sharp}/_{\approx^{\sharp}}$ so that η can be used to provide a unique representation or normal form of equivalent abstract properties. In [17], η would be the operator used at paragraph 3.3.1.2 to simplify systems of linear inequalities.

Let us define the *approximation operator* $\rho \in \mathcal{P}^{\natural} \mapsto \mathcal{P}^{\natural}$ by $\rho \overset{\text{def}}{=} \gamma \circ \alpha$. It is extensive (by Proposition 8.1–1°) and idempotent (by Proposition 8.1–2°), but not monotonic (by Proposition 8.1–4°). It reflects into $\mathcal{P}^{\natural}$ the *a priori* choice of approximations made by α among the sound ones. It is a weakened form of the approximation operator defined at paragraph 5.2 of [12], since monotony is not required.

For a monotonic concrete semantic function $F^{\natural} \in \mathcal{P}^{\natural} \overset{m}{\longmapsto} \mathcal{P}^{\natural}$, the abstract semantic function would be defined by (4.15) so that $c \preceq^{\natural} \gamma(\alpha(c))$ implies $F^{\natural}(c) \preceq^{\natural} F^{\natural}(\gamma(\alpha(c)))$ whence $F^{\natural}(c) \preceq^{\natural} \gamma \circ \alpha \circ F^{\natural}(\gamma(\alpha(c)))$ (by Proposition 8.1–1° and transitivity) and therefore $F^{\natural}(c) \preceq^{\natural} \gamma \circ F^{\sharp} \circ \alpha(c)$ by (4.15). The same way when defining $\mathrm{II}^{\sharp}$ by (4.11), the hypotheses of Proposition 7.3 would be satisfied. Moreover, we can require the abstract iterations to be normalized by application of the normalization operator η to the basis and inductive joins, with no incidence on correctness.

Weak abstraction/concretization connection with abstract approximation pre-order. The construction is quite similar when starting from the abstract approximation ordering and the abstraction function using the duals of propositions as defined in Fig. 1. Observe that the roles of the normalization and approximation operators are also exchanged. In particular, the approximation operator $\eta = \gamma \circ \alpha$, which is the dual normalization operator, shows that a concrete property and its concrete approximation are equivalent when observed from an abstract point of view. The normalization operator $\rho = \alpha \circ \gamma$, which is the dual approximation operator, replaces abstract properties by more precise ones in the abstract, without improving the accuracy of the corresponding concrete semantic properties. This explains why it might be better to start with a concrete approximation ordering $\preceq^{\natural}$ and a concretization function when approximations with loss of information are considered: the abstract ordering is then coarser than the concrete one and so defining the abstract approximation relation in terms of the concrete one is more informative.

Galois connection between the concrete and abstract semantics. The Galois connection framework of [12], succinctly reviewed in Example 4.6, results from the existence of best approximation assumption:

Proposition 8.2 (Galois connection framework)
Let $\langle \mathcal{P}^{\natural};\ \preceq^{\natural} \rangle$ and $\langle \mathcal{P}^{\sharp};\ \preceq^{\sharp} \rangle$ be partially ordered sets. Assume that the soundness relation $\sigma \in \wp(\mathcal{P}^{\natural} \times \mathcal{P}^{\sharp})$ satisfies the existence of a best approximations assumptions (4.25) and (4.56). Define α by (4.23) and γ by (4.55). Then $\langle \mathcal{P}^{\natural};\ \preceq^{\natural} \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathcal{P}^{\sharp};\ \preceq^{\sharp} \rangle$.

Proof. Follows from Proposition 6.2 and its dual. ∎

Observe that the complete lattice or even cpo assumption is useless since the only upper bounds which may be necessary in the poset $\mathcal{P}^{\natural}$ are those involved in the concrete iteration (4.1) and that they are preserved by α, which gives a preference to the use of the abstraction function in order to induce the abstract iterates from the concrete ones. However, in a complete lattice each of γ and α uniquely determines the other according to (4.14) so that the derivation of the abstract semantics from the concrete semantics can be equivalently based upon the use of any of these adjoined functions. Moreover, if all abstract properties are useful in the sense of (4.51) then α is surjective, which for Galois connections is equivalent to $\forall a \in \mathcal{P}^\sharp : \alpha \circ \gamma(a) = a$, as considered in [7].

9 In conclusion, which framework to use?

Starting from the definition of the concrete semantics $\langle \mathcal{P}^\natural;\ \bot^\natural,\ F^\natural,\ \amalg^\natural\rangle$ of programs, the design of an abstract interpretation consists in choosing:

1. an abstract semantic domain $\mathcal{P}^\sharp$ which is an approximate version of the concrete semantic domain $\mathcal{P}^\natural$; and
2. a method for defining an abstract semantics $\langle \bot^\sharp,\ F^\sharp,\ \amalg^\sharp\rangle$ of programs; and
3. the specification of the soundness correspondence between the concrete and abstract properties; and
4. a convergence criterion of the abstract iteration sequence, ensuring the best possible precision; and
5. a convergence acceleration method ensuring rapid termination of the abstract interpreter.

We have discussed several abstract interpretation frameworks, obtained by weakening the hypotheses made in [6, 7, 10, 12, 19]. Each one has many variants, most of them have not been explicitly formulated for short (such as for example the use of an abstraction function together with a concrete approximation relation). To simplify, the principal alternatives are:

1. using a soundness relation σ; or
2. using an abstraction relation $\alpha \in \wp(\mathcal{P}^\natural \times \mathcal{P}^\sharp)$ and an abstract approximation relation $\preceq^\sharp$ so that the soundness relation is $\langle c,\ a\rangle \in \sigma \Leftrightarrow \exists a' \in \mathcal{P}^\sharp : \langle c,\ a'\rangle \in \alpha \wedge a' \preceq^\sharp a$; or
3. using a concretization relation $\gamma \in \wp(\mathcal{P}^\sharp \times \mathcal{P}^\natural)$ and a concrete approximation relation $\preceq^\natural$ so that the soundness relation is $\langle c,\ a\rangle \in \sigma \Leftrightarrow \exists c' \in \mathcal{P}^\natural : c \preceq^\natural c' \wedge \langle a,\ c'\rangle \in \gamma$; or
4. using an abstraction function $\alpha \in \mathcal{P}^\natural \mapsto \mathcal{P}^\sharp$ and an abstract approximation relation $\preceq^\sharp$ so that the soundness relation is $\langle c,\ a\rangle \in \sigma \ \Leftrightarrow\ \alpha(c) \preceq^\sharp a$; or
5. using a concretization function $\gamma \in \mathcal{P}^\sharp \mapsto \mathcal{P}^\natural$ and a concrete approximation relation $\preceq^\natural$ so that the soundness relation is $\langle c,\ a\rangle \in \sigma \ \Leftrightarrow\ c \preceq^\natural \gamma(a)$; or
6. using an Galois connection $\langle \mathcal{P}^\natural;\ \preceq^\natural\rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle \mathcal{P}^\sharp;\ \preceq^\sharp\rangle$ so that the soundness relation is $\langle c,\ a\rangle \in \sigma \ \Leftrightarrow\ \alpha(c) \preceq^\sharp a \ \Leftrightarrow\ c \preceq^\natural \gamma(a)$.

Moreover, the duality principle can be applied:

1. to the concrete $\sqsubseteq^\natural$ and abstract $\sqsubseteq^\sharp$ computational orderings; and/or

2. to the concrete $\preceq^\natural$ and abstract $\preceq^\sharp$ approximation orderings; and/or
3. to the starting semantics (standard, collecting, etc.) which can be concrete or abstract (which consists in exchanging α and γ); and/or
4. to the widening and narrowing (being applied to the abstract and/or concrete semantics).

For a given application, the more powerful applicable framework should be chosen so as to benefit from the best possible guidelines for designing that application. This choice should be guided by the following principles:

1. Preference should be given to the inducing of the abstract interpretation from the starting semantics over empirical designs followed by *a posteriori* soundness verifications; and

2. Efficiency of the implementation should be taken into account during the design of the abstract interpretation (that is in the choice of $\mathcal{P}^\sharp$ but, in addition, in that of $F^\sharp$, $\bigtriangledown^\sharp$ and $\bigtriangleup^\sharp$).

Numerous abstract interpretation frameworks exist and many more are to come in order to take into account the peculiarities of each practical situation. We give our preference to language and semantics independent formulations and hope that this will lead to a cross-fertilization of the various domains of application of abstract interpretation.

Acknowledgements. We would like to thank Alan Mycroft for numerous judicial and useful comments on a first draft of this paper as well as the anonymous referees for their constructive suggestions for improvement. This work was supported in part by Esprit BRA 3124 *Sémantique* and CNRS PRC C^3.

References

[1] S. Abramsky. Abstract interpretation, logical relations and Kan extensions. *Journal of Logic and Computation*, 1(1):5–40, 1990. 517, 519, 527, 529

[2] K. R. Apt and G. D. Plotkin. Countable nondeterminism and random assignment. *Journal of the Association for Computing Machinary*, 33(4):724–767, October 1986. 514, 526

[3] G. L. Burn, C. L. Hankin, and S. Abramsky. Strictness analysis of higher-order functions. *Science of Computer Programming*, 7:249–278, November 1986. 527, 530

[4] F. Bourdoncle. Abstract interpretation by dynamic partitioning. *Journal of Functional Programming*, 1992. (in press). 534

[5] G. L. Burn. *Lazy Functional Languages: Abstract Interpretation and Compilation*. Research Monographs in Parallel and Distributed Computing. Pitman in association with MIT Press, London, 1991. 513

[6] P. Cousot and R. Cousot. Static determination of dynamic properties of programs. In *Proceedings of the 2nd International Symposium on Programming*, pp. 106–130. Dunod, Paris, 1976. 512, 519, 524, 534, 536, 544

[7] P. Cousot and R. Cousot. Abstract interpretation: a unified lattice model for static analysis of programs by construction or approximation of fixpoints. In *Conference Record of the 4thACM Symposium on Principles of Programming Languages*, pp. 238–252, Los

Angeles, CA. ACM Press, New York. 1977. 512, 513, 518, 519, 520, 530, 532, 533, 536, 544

[8] P. Cousot and R. Cousot. Automatic synthesis of optimal invariant assertions: mathematical foundations. In *ACM Symposium on Artificial Intelligence & Programming Languages*, Rochester, NY, SIGPLAN Notices 12(8):1–12, 1977. 512

[9] P. Cousot and R. Cousot. Static determination of dynamic properties of generalized type unions. In *ACM Symposium on Language Design for Reliable Software*, Raleigh, NC, SIGPLAN Notices 12(3):77–94, 1977. 512

[10] P. Cousot and R. Cousot. Static determination of dynamic properties of recursive procedures. In E.J. Neuhold, ed., *IFIP Conference on Formal Description of Programming Concepts*, St-Andrews, N.B., Canada, pp. 237–277. North-Holland, Amsterdam, 1977. 512, 518, 519, 530, 532, 544

[11] P. Cousot and R. Cousot. Constructive versions of Tarski's fixed point theorems. *Pacific Journal of Mathematics*, 82(1):43–57, 1979. 536

[12] P. Cousot and R. Cousot. Systematic design of program analysis frameworks. In *Conference Record of the 6th ACM Symposium on Principles of Programming Languages*, pp. 269–282, San Antonio, TX. ACM Press, New York. 1979. 512, 515, 518, 519, 521, 522, 523, 524, 525, 526, 530, 534, 540, 543, 544

[13] P. Cousot and R. Cousot. Semantic analysis of communicating sequential processes. In de Bakker J. W. and J. van Leeuwen, eds., *Seventh International Colloquium on Automata, Languages and Programming*, Lecture Notes in Computer Science 85, pp. 119–133. Springer-Verlag, Berlin, July 1980. 512

[14] P. Cousot and R. Cousot. Invariance proof methods and analysis techniques for parallel programs. In A. W. Biermann, G. Guiho, and Y. Kodratoff, eds., *Automatic Program Construction Techniques*, ch. 12, pp. 243–271. Macmillan, New York, 1984. 512

[15] P. Cousot and R. Cousot. Abstract interpretation and application to logic programs. *Journal of Logic Programming*, 1992. (in press). 534

[16] P. Cousot and R. Cousot. Inductive definitions, semantics and abstract interpretation. In *Conference Record of the 19th ACM Symposium on Principles of Programming Languages*, pp. 83–94, Albuquerque, NM. ACM Press, New York. 1992. 515

[17] P. Cousot and N. Halbwachs. Automatic discovery of linear restraints among variables of a program. In *Conference Record of the 5th ACM Symposium on Principles of Programming Languages*, pp. 84–97, Tucson, AZ. ACM Press, New York. 1978. 523, 534, 543

[18] P. Cousot. *Méthodes itératives de construction et d'approximation de points fixes d'opérateurs monotones sur un treillis, analyse sémantique de programmes*. Thèse d'État ès sciences mathématiques, Université scientifique et médicale de Grenoble, Grenoble, 21 March 1978. 512, 536

[19] P. Cousot. Semantic foundations of program analysis. In S. S. Muchnick and N. D. Jones, eds., *Program Flow Analysis: Theory and Applications*, ch. 10, pp. 303–342. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1981. 512, 515, 526, 530, 534, 544

[20] J. W. De Bakker, J.-J. Ch. Meyer, and J. I. Zucker. On infinite computations in denotational semantics. *Theoretical Computer Science*, 26:53–82, 1983. (Corrigendum: Theoretical Computer Science 29:229–230, 1984). 526

[21] A. Deutsch. A storeless model of aliasing and its abstraction using finite representations of right-regular equivalence relations. In *Proceedings of the 1992 International Conference on Computer Languages*, Oakland, CA, pp. 2–13. IEEE Computer Society Press, Los Alamitos, CA, April 20–23, 1992. 514, 532, 534

[22] E. W. Dijkstra. Guarded commands, nondeterminacy and formal derivation of programs. *Communications of the Association for Computing Machinary*, 18(8):453–457, August 1975. 516, 517

[23] P. Granger. Improving the results of static analyses of programs by local decreasing iterations. Research Report LIX/RR/91/08, Laboratoire d'Informatique, École Polytechnique, France, December 1991. 530

[24] C. A. Gunter and D. S. Scott. Semantic domains. In J. van Leeuwen, ed., *Formal Models and Semantics*, volume B of *Handbook of Theoretical Computer Science*, ch. 12, pp. 633–674. Elsevier Science Publishers B.V. (North-Holland), Amsterdam, 1990. 514, 528

[25] S. Hunt and D. Sands. Binding time analysis: A new PERspective. In *Proceedings of the ACM Symposium on Partial Evaluation and Semantics-Based Program Manipulation PEPM'91*, Yale University, New Haven, Connecticut, 17–19, 1991, SIGPLAN Notices 26(9), pp. 154–165. ACM Press, New York, September 1991. 519

[26] S. Hunt. PERs generalize projections for strictness analysis. Technical Report DOC 14/90, Department of Computing, Imperial College, London, August 1990. 519

[27] N. D. Jones and A. Mycroft. Data flow analysis of applicative programs using minimal function graphs: abridged version. In *Conference Record of the 13th ACM Symposium on Principles of Programming Languages*, pp. 296–306, St. Petersburg Beach, FL. ACM Press, New York. 1986. 532

[28] J. Launchbury. *Projection Factorizations in Partial Evaluation*, volume 1 of *Distinguished Dissertations in Computer Science*. Cambridge University Press, Cambridge, 1991. 513

[29] A. Mycroft and N. D. Jones. A relational framework for abstract interpretation. In N. D. Jones and H. Ganzinger, eds., *Programs as Data Objects, Proceedings of a Workshop*, Copenhagen, Denmark, 17-19 October 1985, Lecture Notes in Computer Science 215, pp. 156–171. Springer-Verlag, Berlin, 1986. 517

[30] A. Mycroft and F. Nielson. Strong abstract interpretation using power domains. In J. Diaz, ed., *Tenth International Colloquium on Automata, Languages and Programming*, Lecture Notes in Computer Science 154, pp. 536–547. Springer-Verlag, Berlin, 1983. 513

[31] P. D. Mosses. Denotational semantics. In J. van Leeuwen, ed., *Formal Models and Semantics*, volume B of *Handbook of Theoretical Computer Science*, ch. 11, pp. 575–631. Elsevier Science Publishers B.V. (North-Holland), Amsterdam, 1990. 514

[32] K. Marriott and H. Søndergaard. Bottom-up abstract interpretation of logic programs. In R. Kowalski and K. Bowen, eds., *Proceedings of the Fifth International Conference and Symposium on Logic Programming, Volume 1*, Seattle, Washington, pp. 733–748. MIT Press, Cambridge, MA, August 1988. 524

[33] K. Marriott and H. Søndergaard. On describing success patterns of logic programs. Technical Report 88/12, Department of Computer Science, University of Melbourne, Melbourne, 1988. 524

[34] A. Mycroft. The theory and practice of transforming call-by-need into call-by-value. In B. Robinet, ed., *Proceedings of the Fourth International Symposium on Programming*, Paris, 22-24 April 1980, Lecture Notes in Computer Science 83, pp. 270–281. Springer-Verlag, Berlin, 1980. 513

[35] A. Mycroft. *Abstract Interpretation and Optimising Transformations for Applicative Programs*. Ph.D. Dissertation, CST-15-81, Department of Computer Science, University of Edinburgh, Edinburgh, December 1981. 513, 530

[36] J. C. Reynolds. On the relation between direct and continuation semantics. In J. Lœckx, ed., *Second International Colloquium on Automata, Languages and Programming*, Lecture Notes in Computer Science 14, pp. 141–156. Springer-Verlag, Berlin, July 1974. 516, 517

[37] B. Steffen, C. B. Jay, and M. Mendler. Compositional characterization of observable program properties. LFCS Report Series ECS-LFCS-89-99, Laboratory for Founda-

tions of Computer Science, Department of Computer Science, University of Edinburgh, Edinburgh, November 1989. 519

[38] T. Sato and H. Tamaki. Enumeration of success patterns in logic programs. *Theoretical Computer Science*, 34:227–240, 1984. 524

Received 9 January 1992

This article is reprinted from the "Special Issue on Abstract Interpretation" of :
The Journal of Logic and Computation,
Volume 2, Number 4, Pages 511–547, august 1992.