A Galois Connection Calculus for Abstract Interpretation*

Patrick Cousot
CIMS**, NYU, USA pcousot@cims.nyu.edu

Radhia Cousot
CNRS Emeritus, ENS, France rcousot@ens.fr

Abstract We introduce a Galois connection calculus for language independent specification of abstract interpretations used in programming language semantics, formal verification, and static analysis. This Galois connection calculus and its type system are typed by abstract interpretation.

Categories and Subject Descriptors D.2.4 [*Software/Program Verification*]
General Terms Algorithms, Languages, Reliability, Security, Theory, Verification.
Keywords Abstract Interpretation, Galois connection, Static Analysis, Verification.

1. Galois connections in Abstract Interpretation In *Abstract interpretation* [3, 4, 6, 7] concrete properties (for example (*e.g.*) of computations) are related to abstract properties (*e.g.* types). The abstract properties are always *sound* approximations of the concrete properties (abstract proofs/static analyzes are always correct in the concrete) and are sometimes *complete* (proofs/analyzes of abstract properties can all be done in the abstract only). *E.g.* types are sound but incomplete [2] while abstract semantics are usually complete [9]. The *concrete domain* $\langle\mathcal{C},\ \sqsubseteq\rangle$ and *abstract domain* $\langle\mathcal{A},\ \preccurlyeq\rangle$ of properties are posets (partial orders being interpreted as implication). When concrete properties all have a $\preccurlyeq$-most precise abstraction, the correspondence is a *Galois connection* (*GC*) $\langle\mathcal{C},\ \sqsubseteq\rangle \xrightleftharpoons[\alpha]{\gamma} \langle\mathcal{A},\ \preccurlyeq\rangle$ with *abstraction* $\alpha \in \mathcal{C} \mapsto \mathcal{A}$ and *concretization* $\gamma \in \mathcal{A} \mapsto \mathcal{C}$ satisfying $\forall P \in \mathcal{C} : \forall Q \in \mathcal{A} : \alpha(x) \preccurlyeq y \Leftrightarrow x \sqsubseteq \gamma(y)$ ($\Rightarrow$ expresses soundness and $\Leftarrow$ best abstraction). Each adjoint α/γ uniquely determines the other γ/α. A *Galois retraction* (or *insertion*) has α onto, so γ is one-to-one, and $\alpha\circ\gamma$ is the identity. *E.g.* the *interval abstraction* [3, 4] of the power set $\wp(C)$ of complete $\le$-totally ordered sets $C \cup \{-\infty, \infty\}$ is $\mathcal{S}\llbracket\mathbb{I}[\langle C, \le\rangle, -\infty, \infty]\rrbracket \triangleq \langle\wp(C), \subseteq\rangle \xrightleftharpoons[\alpha^{\mathbb{I}}]{\gamma^{\mathbb{I}}} \langle\mathbb{I}(C\cup\{-\infty,\infty\}, \le), \Subset\rangle$ with $\alpha^{\mathbb{I}}(X) \triangleq [\min X, \max X]$, $\min\emptyset \triangleq \infty$, $\max\emptyset \triangleq -\infty$, $\gamma^{\mathbb{I}}([a, b]) \triangleq \{x \in C \mid a \le x \le b\}$, intervals $\mathcal{S}\llbracket\mathbb{I}(C \cup \{-\infty, \infty\}, \le)\rrbracket \triangleq \{[a, b] \mid a \in C \cup \{-\infty\} \wedge b \in C \cup \{\infty\} \wedge a \le b\} \cup \{[\infty, -\infty]\}$, and inclusion $[a, b] \Subset [c, d] \triangleq c \le a \wedge b \le d$. A *Galois isomorphism* $\langle\mathcal{C}, \subseteq\rangle \xrightleftharpoons[\alpha]{\gamma} \langle\mathcal{A}, \preccurlyeq\rangle$ has both α and γ bijective. *E.g.* global and local invariants are isomorphic by the *right image abstraction* $\mathcal{S}\llbracket\curvearrowright[\mathbb{L}, \mathcal{M}]\rrbracket \triangleq \langle\wp(\mathbb{L}\times\mathcal{M}), \subseteq\rangle \xrightleftharpoons[\alpha^{\curvearrowright}]{\gamma^{\curvearrowright}} \langle\mathbb{L} \mapsto \wp(\mathcal{M}),\ \dot{\subseteq}\rangle$ with $\alpha^{\curvearrowright}(P) \triangleq \boldsymbol{\lambda}\,\ell \bullet \{m \mid \langle\ell,\ m\rangle \in P\}$, $\gamma^{\curvearrowright}(Q) \triangleq \{\langle\ell,\ m\rangle \mid m \in Q(\ell)\}$, and $\dot{\subseteq}$ is the pointwise extension of inclusion $\subseteq$.

2. Equivalent formalizations of *GC*-based Abstract Interpretation *GCs* $\langle\mathcal{C},\ \sqsubseteq\rangle \xrightleftharpoons[\alpha]{\gamma} \langle\mathcal{A},\ \preccurlyeq\rangle$ are Galois retracts of/Galois isomorphic to numerous equivalent mathematical structures [6] such as *join-preserving maps* (α), *meet-preserving maps* (γ), *upper-closures* ($\gamma\circ\alpha$), *Moore families* ($\{\gamma(Q) \mid Q \in \mathcal{A}\}$), *Sierpiński topologies* [5] ($\{\neg\gamma(Q) \mid Q \in \mathcal{A}\}$ where $\neg$ is unique complementation in the concrete domain $\mathcal{C}$, if any), *principal downset families* ($\{{\downarrow}^{\sqsubseteq}\gamma(Q) \mid Q \in \mathcal{A}\}$ where ${\downarrow}^{\sqsubseteq}x \triangleq \{y \in \mathcal{C} \mid y \sqsubseteq x\}$), *maximal convex congruences* ($\{\{P \in \mathcal{C} \mid \alpha(P) = \alpha(\gamma(Q))\} \mid Q \in \mathcal{A}\}$, *soundness relations* (also called *abstraction relation*, *logical relation*, or *tensor product*, $\alpha \fatsemi \preccurlyeq\ = \{\langle P,\ Q\rangle \mid \alpha(P) \preccurlyeq Q\} = \{\langle P,\ Q\rangle \mid P \sqsubseteq \gamma(Q)\} = \sqsubseteq \fatsemi \gamma^{-1}$ where $f \equiv \{\langle x,\ f(x)\rangle \mid x \in \mathsf{dom}(f)\}$, $r \fatsemi r' = \{\langle x,\ z\rangle \mid \exists y : \langle x,\ y\rangle \in r \wedge \langle y,\ z\rangle \in r'\}$), and, for powersets $\mathcal{C} = \wp(C)$, $\mathcal{A} = \wp(A)$, *polarities* of relations ($\gamma(Q) = \{x \in C \mid \forall y \in Q : R(x, y)\}$ where $R = \{\langle x,\ y\rangle \mid x \in \gamma(\{y\})\}$).

3. Basic *GC* semantics Basic *GCs* are primitive abstractions of properties. Classical examples are the *identity abstraction* $\mathcal{S}\llbracket\mathbb{1}[\langle\mathcal{C},\ \sqsubseteq\rangle]\rrbracket \triangleq \langle\mathcal{C},\ \sqsubseteq\rangle \xrightleftharpoons[\boldsymbol{\lambda}\,P\bullet P]{\boldsymbol{\lambda}\,Q\bullet Q} \langle\mathcal{C},\ \sqsubseteq\rangle$, the *top abstraction* $\mathcal{S}\llbracket\top[\langle\mathcal{C},\ \sqsubseteq\rangle, \top]\rrbracket \triangleq \langle\mathcal{C}, \sqsubseteq\rangle \xrightleftharpoons[\boldsymbol{\lambda}\,P\bullet\top]{\boldsymbol{\lambda}\,Q\bullet\top} \langle\mathcal{C}, \sqsubseteq\rangle$, the *join abstraction* $\mathcal{S}\llbracket\cup[C]\rrbracket \triangleq \langle\wp(\wp(C)), \subseteq\rangle \xrightleftharpoons[\alpha^{\wp}]{\gamma^{\wp}} \langle\wp(C), \subseteq\rangle$ with $\alpha^{\wp}(P) \triangleq \bigcup P$, $\gamma^{\wp}(Q) \triangleq \wp(Q)$, the *complement abstraction* $\mathcal{S}\llbracket\neg[C]\rrbracket \triangleq \langle\wp(C),\ \subseteq\rangle \xrightleftharpoons[\neg]{\neg} \langle\wp(C),\ \supseteq\rangle$, the finite/infinite *sequence abstraction* $\mathcal{S}\llbracket\infty[C]\rrbracket \triangleq \langle\wp(C^{\infty}), \subseteq\rangle \xrightleftharpoons[\alpha^{\infty}]{\gamma^{\infty}} \langle\wp(C), \subseteq\rangle$ with $\alpha^{\infty}(P) \triangleq \{\sigma_i \mid \sigma \in P \wedge i \in \mathsf{dom}(\sigma)\}$ and $\gamma^{\infty}(Q) \triangleq \{\sigma \in C^{\infty} \mid \forall i \in \mathsf{dom}(\sigma) : \sigma_i \in Q\}$, the *transformer abstraction* $\mathcal{S}\llbracket\rightsquigarrow[C_1, C_2]\rrbracket \triangleq \langle\wp(C_1\times C_2), \subseteq\rangle \xrightleftharpoons[\alpha^{\rightsquigarrow}]{\gamma^{\rightsquigarrow}} \langle\wp(C_1) \xrightarrow{\cup} \wp(C_2),\ \dot{\subseteq}\rangle$ mapping relations to join-preserving transformers with $\alpha^{\rightsquigarrow}(R) \triangleq \boldsymbol{\lambda}\,X \bullet \{y \mid \exists x \in X : \langle x,\ y\rangle \in R\}$, $\gamma^{\rightsquigarrow}(g) \triangleq \{\langle x,\ y\rangle \mid y \in g(\{x\})\}$, the *function abstraction* $\mathcal{S}\llbracket\mapsto[C_1, C_2]\rrbracket \triangleq \langle\wp(C_1 \mapsto C_2),\ \subseteq\rangle \xrightleftharpoons[\alpha^{\mapsto}]{\gamma^{\mapsto}} \langle\wp(C_1) \mapsto \wp(C_2),\ \dot{\subseteq}\rangle$ with $\alpha^{\mapsto}(P) \triangleq \boldsymbol{\lambda}\,X \bullet \{f(x) \mid f \in P \wedge x \in X\}$, $\gamma^{\mapsto}(g) \triangleq \{f \in C_1 \mapsto C_2 \mid \forall X \in \wp(C_1) : \forall x \in X : f(x) \in g(X)\}$, the *cartesian abstraction* $\mathcal{S}\llbracket\times[I, C]\rrbracket \triangleq \langle\wp(I \mapsto C),\ \subseteq\rangle \xrightleftharpoons[\alpha^{\times}]{\gamma^{\times}} \langle I \mapsto \wp(C),\ \dot{\subseteq}\rangle$ with $\alpha^{\times}(X) \triangleq \boldsymbol{\lambda}\,i \in I \bullet \{x \in C \mid \exists f \in I \mapsto C : f[i \leftarrow x] \in X\}$, $\gamma^{\times}(Y) \triangleq \{f \mid \forall i \in I : f(i) \in Y(i)\}$, and the pointwise extension $\dot{\subseteq}$ of $\subseteq$ to I, *etc.*

4. Galois connector semantics *Galois connectors* build a *GC* from *GCs* provided as parameters. Unary Galois connectors include the *reduction connector* $\mathcal{S}\llbracket\mathsf{R}[\langle\mathcal{C},\ \sqsubseteq\rangle \xrightleftharpoons[\alpha]{\gamma} \langle\mathcal{A},\ \preccurlyeq\rangle]\rrbracket \triangleq \langle\mathcal{C},\ \sqsubseteq\rangle \xrightleftharpoons[\alpha]{\gamma} \langle\{\alpha(P) \mid P \in \mathcal{C}\},\ \preccurlyeq\rangle$ and the *pointwise connector* $\mathcal{S}\llbracket X \multimap \langle C,\ \sqsubseteq\rangle \xrightleftharpoons[\alpha]{\gamma} \langle A,\ \preccurlyeq\rangle\rrbracket \triangleq \langle X \mapsto C,\ \dot{\sqsubseteq}\rangle \xrightleftharpoons[\boldsymbol{\lambda}\,\rho\bullet\alpha\circ\rho]{\boldsymbol{\lambda}\,\overline{\rho}\bullet\gamma\circ\overline{\rho}} \langle X \mapsto A,\ \dot{\preccurlyeq}\rangle$ for the pointwise orderings $\dot{\sqsubseteq}$ and $\dot{\preccurlyeq}$. Binary Galois connectors include the *composition connector* $\mathcal{S}\llbracket\langle\mathcal{C}, \subseteq\rangle \xrightleftharpoons[\alpha_1]{\gamma_1} \langle\mathcal{A}_1, \Subset\rangle \fatsemi \langle\mathcal{A}_2, \sqsubseteq\rangle \xrightleftharpoons[\alpha_2]{\gamma_2} \langle\mathcal{A}_3, \preccurlyeq\rangle\rrbracket \triangleq \llparenthesis\langle\mathcal{A}_1, \Subset\rangle = \langle\mathcal{A}_2, \sqsubseteq\rangle\ ?\ \langle\mathcal{C},\ \subseteq\rangle \xrightleftharpoons[\alpha_2\circ\alpha_1]{\gamma_1\circ\gamma_2} \langle\mathcal{A}_3,\ \preccurlyeq\rangle \mathbin{⨾} \Omega\rrparenthesis$ (where Ω is a static error), the *product connector* $\mathcal{S}\llbracket\langle\mathcal{C}_1,\ \subseteq\rangle \xrightleftharpoons[\alpha_1]{\gamma_1} \langle\mathcal{A}_1,\ \sqsubseteq\rangle \divideontimes \langle\mathcal{C}_2,\ \Subset\rangle \xrightleftharpoons[\alpha_2]{\gamma_2} \langle\mathcal{A}_2,\ \preccurlyeq\rangle\rrbracket \triangleq \langle\mathcal{C}_1 \times \mathcal{C}_2,\ \subseteq\times\Subset\rangle \xrightleftharpoons[\alpha_1\times\alpha_2]{\gamma_1\times\gamma_2} \langle\mathcal{A}_1 \times \mathcal{A}_2,\ \sqsubseteq\times\preccurlyeq\rangle$ (generalizing to tuples), the higher-order *functional connector* $\mathcal{S}\llbracket\langle\mathcal{C}_1,\ \subseteq\rangle \xrightleftharpoons[\alpha_1]{\gamma_1} \langle\mathcal{A}_1,\ \sqsubseteq\rangle \Longmapsto \langle\mathcal{C}_2,\ \preccurlyeq\rangle \xrightleftharpoons[\alpha_2]{\gamma_2} \langle\mathcal{A}_2,\ \leqslant\rangle\rrbracket \triangleq \langle\mathcal{C}_1 \xrightarrow{\nearrow} \mathcal{C}_2,\ \dot{\preccurlyeq}\rangle \xrightleftharpoons[\boldsymbol{\lambda}\,f\bullet\alpha_2\circ f\circ\gamma_1]{\boldsymbol{\lambda}\,g\bullet\gamma_2\circ g\circ\alpha_1} \langle\mathcal{A}_1 \xrightarrow{\nearrow} \mathcal{A}_2,\ \dot{\leqslant}\rangle$ for increasing maps and pointwise orderings $\dot{\sqsubseteq}$ and $\dot{\leqslant}$.

5. Galois connection calculus The *GC* calculus $\mathbb{G}$ (to specify verifiers/analyzers compositionally) is $\mathrm{x}, \ldots \in \mathbb{X}$ for program variables, $\ell, \ldots \in \mathbb{L}$ for labels, $e \in \mathbb{E}$ for elements $e ::= \mathsf{true} \mid 1 \mid \infty \mid \mathrm{x} \mid \ell \mid -e \mid \ldots$, $s \in \mathbb{S}$ for sets $s ::= \mathbb{B} \mid \mathbb{Z} \mid \mathbb{X} \mid \mathbb{L} \mid \{e\} \mid [e, e] \mid \mathbb{I}(s, o) \mid s^{\infty} \mid s \cup s \mid s \mapsto s \mid s \times s \mid \wp(s) \mid \ldots$, $o \in \mathbb{O}$ for partial orders $o ::= \Rightarrow \mid \Leftrightarrow \mid \le \mid \subseteq \mid \Subset \mid = \mid o^{-1} \mid o_1 \times o_2 \mid \dot{o} \mid \ddot{o} \mid \ldots$, $p \in \mathbb{P}$ for posets $p ::= \langle s,\ o\rangle$, and $g \in \mathbb{G}$ for *GCs* $g ::= \mathbb{1}[p] \mid \top[p, e] \mid \mathbb{I}[p, e, e] \mid \curvearrowright[s, s] \mid \cup[s] \mid \neg[s] \mid \infty[s] \mid \rightsquigarrow[s, s] \mid \mapsto[s, s] \mid \times[s, s] \mid \ldots \mid \mathsf{R}[g] \mid s \multimap g \mid g \fatsemi g \mid g \divideontimes g \mid g \Longmapsto g \mid \ldots$. The semantics of interval sets is $\mathcal{S}\llbracket\mathbb{I}(C, \preccurlyeq)\rrbracket \triangleq \llparenthesis\preccurlyeq\ \subseteq C \times C\ ?\ \{[a, b]_{\preccurlyeq} \mid a, b \in C\} \mathbin{⨾} \omega\rrparenthesis$ where ω is a dynamic error (maybe not detectable by typing).

---

* See the auxiliary materials. ** Work supported in part by the CMACS NSF award 0926166.

POPL '14, January 22–24, 2014, San Diego, CA, USA.
ACM 978-1-4503-2544-8/14/01. http://dx.doi.org/10.1145/2535838.2537850

6. Abstraction Papers in semantics, verification, and static analysis can be understood by extracting the semantic domain and *GC* which are used. For the interval example [3, 4, p. 247], the semantic domain $\mathcal{S} \triangleq \wp(\Sigma^\infty)$ is that of (nonempty) sets of nonempty finite or infinite sequences of states in $\Sigma \triangleq \mathbb{L} \times \mathcal{M}$ made of a control state in $\mathbb{L}$ and a memory state in $\mathcal{M} \triangleq \mathbb{X} \mapsto \mathcal{V}$ mapping variables $\mathbb{X}$ to a complete total order $\langle \mathcal{V},\ \leq \rangle$ (*e.g.* $\langle \mathbb{Z}, \leqslant \rangle$ or $\langle [\mathtt{minint}, \mathtt{maxint}],\ \leqslant \rangle$). The *static* (or *collecting*) *semantics* is the *reachability abstraction* of program properties in $\wp(\mathcal{S})$ that is $G^* \triangleq \cup[\Sigma^\infty] \mathbin{;} \infty[\Sigma] \mathbin{;} \curvearrowright[\mathbb{L}, \mathcal{M}]$ with abstract domain $\langle \mathbb{L} \mapsto \wp(\mathcal{M}),\ \dot{\subseteq} \rangle$. The *reduced interval cartesian reachability abstraction* is $G^{\mathfrak{I}^*} \triangleq \mathsf{R}[G^* \mathbin{;} (\mathbb{L} \twoheadrightarrow (\times[\mathbb{X}, \mathcal{V}] \mathbin{;} (\mathbb{X} \twoheadrightarrow \mathbb{I}[\langle \mathcal{V},\ \leq \rangle, -\infty, \infty]))))]$ that is the abstraction $\langle \wp(\wp((\mathbb{L} \times (\mathbb{X} \mapsto \mathcal{V}))^\infty)),\ \subseteq \rangle \xleftrightarrows[\alpha]{\gamma} \langle \mathbb{L} \mapsto \mathbb{X} \mapsto \mathbb{I}(\mathcal{V} \cup \{-\infty, \infty\}),\ \ddot{\sqsubseteq} \rangle$ where $\alpha(P) \triangleq \boldsymbol{\lambda}\, \ell \bullet \mathsf{smash}(\boldsymbol{\lambda}\, \mathtt{x} \bullet \alpha^{\mathbb{I}}(\alpha^{\times}((\alpha^{\curvearrowright}(\alpha^{\infty}(\alpha^{\wp}(P))))(\ell))(\mathtt{x})))$ and $\mathsf{smash}(\boldsymbol{\lambda}\, \mathtt{x} \in \mathbb{X} \bullet [a_{\mathtt{x}}, b_{\mathtt{x}}])$ reduces to $\boldsymbol{\lambda}\, \mathtt{x} \in \mathbb{X} \bullet [\infty, -\infty]$ when some $[a_{\mathtt{x}}, b_{\mathtt{x}}]$ is the empty interval $[\infty, -\infty]$ else to $\boldsymbol{\lambda}\, \mathtt{x} {\in} \mathbb{X} \bullet [a_{\mathtt{x}}, b_{\mathtt{x}}]$.

7. Typing As usual with syntactic definitions, *GC* expression semantics may be undefined (*i.e.* Ω or ω). This can be fixed for Ω by a type system that is an Abstract Interpretation of the properties $\wp(\mathfrak{Gc})$ of the semantics $\mathcal{S}[\![g]\!] \in \mathfrak{Gc}$ of expressions $g \in \mathbb{G}$ belonging to the class $\mathfrak{Gc} \triangleq \{\langle \mathcal{C}, \sqsubseteq \rangle \xleftrightarrows[\alpha]{\gamma} \langle \mathcal{A}, \preccurlyeq \rangle \mid \mathcal{C}, \mathcal{A}$ are sets $\wedge \sqsubseteq \in \wp(\mathcal{C} \times \mathcal{C}) \wedge \preccurlyeq \in \wp(\mathcal{A} \times \mathcal{A})\} \cup \{\Omega, \omega\}$. Typing is formalized by a *GC* [2] $\langle \wp(\mathfrak{Gc}),\ \subseteq \rangle \xleftrightarrows[\alpha^{\mathfrak{T}}]{\gamma^{\mathfrak{T}}} \langle \mathfrak{T}_{/\cong},\ \trianglelefteq \rangle$ where the preorder on types is $\mathsf{T} \trianglelefteq \mathsf{T}' \triangleq \gamma^{\mathfrak{T}}(\mathsf{T}) \subseteq \gamma^{\mathfrak{T}}(\mathsf{T}')$ so that types $\mathfrak{T}_{/\cong}$ are considered up to the equivalence $\cong$ for this preorder $\trianglelefteq$ ($\cong$ is $=$ when $\gamma^{\mathfrak{T}}$ is injective). In absence of a $\trianglelefteq$-most precise *i.e.* principal type, hence of a best abstraction $\alpha^{\mathfrak{T}}$, as *e.g.* in [15] for the polyhedral abstraction, only one of α or γ is used [7]. A *GC* expression $g \in \mathbb{G}$ has sound types $\mathsf{T} \in \mathfrak{T}$ such that $\{\mathcal{S}[\![g]\!]\} \subseteq \gamma^{\mathfrak{T}}(\mathsf{T})$ *i.e.* $\mathcal{S}[\![g]\!] \in \gamma^{\mathfrak{T}}(\mathsf{T})$ or $\rho^{\mathfrak{Gc}}(\mathcal{S}[\![g]\!], \mathcal{T}[\![g]\!])$ for the soundness relation $\rho^{\mathfrak{Gc}}(S, T) \triangleq S \in \gamma^{\mathfrak{T}}(T)$. For *GCs*, this is equivalent to $\alpha^{\mathfrak{T}}(\{\mathcal{S}[\![g]\!]\}) \trianglelefteq \mathsf{T}$, where $\{\mathcal{S}[\![g]\!]\}$ is the strongest property (collecting semantics) of g and $\alpha^{\mathfrak{T}}(\{\mathcal{S}[\![g]\!]\})$ is the best abstraction of g. The type soundness proof is by induction on the structure of the *GC* expressions as in [2] (instead of operational subject reduction *i.e.* induction on program computation steps).

8. Types For elements $\mathsf{E} \in \mathfrak{E}$, $\mathsf{E} ::= \mathtt{var} \mid \mathtt{lab} \mid \mathtt{bool} \mid \mathtt{int} \mid \mathtt{err}$ with $\gamma^{\mathfrak{E}}(\mathtt{int}) \triangleq \mathbb{Z} \cup \{-\infty, \infty\}$, $\gamma^{\mathfrak{E}}(\mathtt{err}) \triangleq \mathcal{S}(\mathbb{E}) \cup \{\Omega, \omega\}$.

For sets $\mathsf{S} \in \mathfrak{S}$, $\mathsf{S} ::= \mathtt{P}\ \mathsf{E} \mid \mathtt{P}\ \mathsf{S} \mid \mathtt{seq}\ \mathsf{S} \mid \mathsf{S} \mathrel{*\!\!\rightarrow} \mathsf{S} \mid \mathsf{S} * \mathsf{S} \mid \mathtt{err}$ with $\gamma^{\mathfrak{S}}(\mathtt{P}\ \mathsf{E}) \triangleq \wp(\gamma^{\mathfrak{E}}(\mathsf{E}))$, $\gamma^{\mathfrak{S}}(\mathtt{P}\ \mathsf{S}) \triangleq \wp(\gamma^{\mathfrak{S}}(\mathsf{S}))$, $\gamma^{\mathfrak{S}}(\mathtt{seq}\ \mathsf{S}) \triangleq \{X^\infty \mid X \in \gamma^{\mathfrak{S}}(\mathsf{S})\}$, $\gamma^{\mathfrak{S}}(\mathsf{S}_1 \mathrel{*\!\!\rightarrow} \mathsf{S}_2) \triangleq \{X \mapsto Y \mid X \in \gamma^{\mathfrak{S}}(\mathsf{S}_1) \wedge Y \in \gamma^{\mathfrak{S}}(\mathsf{S}_2)\}$, $\gamma^{\mathfrak{S}}(\mathsf{S}_1 * \mathsf{S}_2) \triangleq \{X \times Y \mid X \in \gamma^{\mathfrak{S}}(\mathsf{S}_1) \wedge Y \in \gamma^{\mathfrak{S}}(\mathsf{S}_2)\}$, $\gamma^{\mathfrak{S}}(\mathtt{err}) \triangleq \mathcal{S}(\mathbb{S}) \cup \{\Omega, \omega\}$.

For partial orders $\mathsf{O} \in \mathfrak{O}$, $\mathsf{O} ::= \Rightarrow \mid \Leftrightarrow \mid \leq \mid \subseteq \mid = \mid \mathsf{O}^{-1} \mid \mathsf{O} {\star} \mathsf{O} \mid \dot{\mathsf{O}} \mid \ldots \mid \mathtt{err}$ with $\gamma^{\mathfrak{O}}(\mathsf{O}) \triangleq \{\mathsf{O}\}$, $\mathsf{O} \in \{\Rightarrow, \Leftrightarrow, \leq, \subseteq, =\}$, $\Rightarrow \triangleq \{\langle \mathsf{false},\ \mathsf{false} \rangle, \langle \mathsf{true},\ \mathsf{true} \rangle\}$, *etc.*

For posets $\mathsf{P} \in \mathfrak{P}$, $\mathsf{P} ::= \mathsf{S} \circledast \mathsf{O} \mid \mathtt{err}$ with componentwise concretization $\gamma^{\mathfrak{P}}(\mathsf{S} \circledast \mathsf{O}) \triangleq \gamma^{\mathfrak{S}}(\mathsf{S}) \times \gamma^{\mathfrak{O}}(\mathsf{O})$.

For *GCs*, $\mathsf{T} \in \mathfrak{T}$, $\mathsf{T} ::= \mathsf{P} \leftrightarrows \mathsf{P} \mid \mathsf{S} \mathrel{*\!\!\twoheadrightarrow} \mathsf{T} \mid \mathtt{err}$ with $\gamma^{\mathfrak{T}}(\mathsf{P} \leftrightarrows \mathsf{P}') \triangleq \{P \xleftrightarrows[\alpha]{\gamma} P' \mid P \in \gamma^{\mathfrak{P}}(\mathsf{P}) \wedge P' \in \gamma^{\mathfrak{P}}(\mathsf{P}')\}$ and $\gamma^{\mathfrak{T}}(\mathsf{S} \mathrel{*\!\!\twoheadrightarrow} \mathsf{T}) \triangleq \{\langle X \mapsto \mathcal{C},\ \dot{\sqsubseteq} \rangle \xleftrightarrows[\alpha']{\gamma'} \langle X \mapsto \mathcal{A},\ \dot{\preccurlyeq} \rangle \mid X \in \gamma^{\mathfrak{S}}(\mathsf{S}) \wedge \langle \mathcal{C}, \sqsubseteq \rangle \xleftrightarrows[\alpha]{\gamma} \langle \mathcal{A}, \preccurlyeq \rangle \in \gamma^{\mathfrak{T}}(\mathsf{T})\}$.

9. Type inference The type inference algorithm is $\mathcal{E}[\![\mathsf{true}]\!] \triangleq \mathtt{bool}, \ldots, \mathcal{E}[\![-e]\!] \triangleq (\!\!| \mathcal{E}[\![e]\!] = \mathtt{bool} \vee \mathcal{E}[\![e]\!] = \mathtt{int} \mathbin{?} \mathcal{E}[\![e]\!] \mathbin{:} \mathtt{err} |\!\!)$.

For sets $\mathcal{S}[\![\mathbb{B}]\!] \triangleq \mathtt{P}\ \mathtt{bool}, \ldots, \mathcal{S}[\![\{e\}]\!] \triangleq (\!\!| \mathcal{E}[\![e]\!] \neq \mathtt{err} \mathbin{?} \mathtt{P}\ \mathcal{E}[\![e]\!] \mathbin{:} \mathtt{err} |\!\!), \ldots, \mathcal{S}[\![s_1 \cup s_2]\!] \triangleq (\!\!| \mathtt{err} \neq \mathcal{S}[\![s_1]\!] \cong \mathcal{S}[\![s_2]\!] \neq \mathtt{err} \mathbin{?} \mathcal{S}[\![s_1]\!] \mathbin{:} \mathtt{err} |\!\!)$ (note the approximation that s_1 and s_2 must have equivalent types as for alternatives of conditionals in functional languages). Ignoring error propagation, $\mathcal{S}[\![s^\infty]\!] \triangleq \mathtt{seq}\ \mathcal{S}[\![s]\!]$, $\mathcal{S}[\![s_1 \mapsto s_2]\!] \triangleq \mathcal{S}[\![s_1]\!] \mathrel{*\!\!\rightarrow} \mathcal{S}[\![s_2]\!]$, $\mathcal{S}[\![s_1 \times s_2]\!] \triangleq \mathcal{S}[\![s_1]\!] * \mathcal{S}[\![s_2]\!]$, $\mathcal{S}[\![\wp(s)]\!] \triangleq \mathtt{P}\ \mathcal{S}[\![s]\!]$.

For orders and posets, $\mathcal{O}[\![o]\!] \triangleq o$, $o \in \{\Rightarrow, \Leftrightarrow, \leq, \subseteq, =\}$, $\mathcal{O}[\![\dot{\sqsubseteq}]\!] \triangleq \subseteq, \ldots, \mathcal{O}[\![\ddot{o}]\!] \triangleq ((\mathcal{O}[\![\dot{o}]\!]))$, and $\mathcal{P}[\![\langle s,\ o \rangle]\!] \triangleq \mathcal{S}[\![s]\!] \circledast \mathcal{O}[\![o]\!]$.

For *GCs*, $\mathcal{T}[\![\mathbb{1}[p]]\!] \triangleq \mathcal{P}[\![p]\!] \leftrightarrows \mathcal{P}[\![p]\!]$, $\mathcal{T}[\![\curvearrowright[s_{\mathbb{L}}, s_{\mathcal{M}}]]\!] \triangleq \mathtt{P}(\mathcal{S}[\![s_{\mathbb{L}}]\!] * \mathcal{S}[\![s_{\mathcal{M}}]\!]) \circledast \subseteq \leftrightarrows \mathcal{S}[\![s_{\mathbb{L}}]\!] \mathrel{*\!\!\rightarrow} \mathtt{P}\ \mathcal{S}[\![s_{\mathcal{M}}]\!] \circledast \dot{\subseteq}$, $\mathcal{T}[\![\cup[s]]\!] \triangleq \mathtt{P}\ (\mathtt{P}\ \mathcal{S}[\![s]\!]) \circledast \subseteq \leftrightarrows \mathtt{P}\ \mathcal{S}[\![s]\!] \circledast \subseteq$, $\mathcal{T}[\![\neg[s]]\!] \triangleq \mathtt{P}\ \mathcal{S}[\![s]\!] \circledast \subseteq \leftrightarrows \mathtt{P}\ \mathcal{S}[\![s]\!] \circledast \subseteq^{-1}$, $\mathcal{T}[\![\infty[s]]\!] \triangleq \mathtt{P}\ (\mathtt{seq}\ \mathcal{S}[\![s]\!]) \circledast \subseteq \leftrightarrows \mathtt{P}\ \mathcal{S}[\![s]\!] \circledast \subseteq$, $\mathcal{T}[\![\rightsquigarrow[s_1, s_2]]\!] \triangleq \mathtt{P}\ (\mathcal{S}[\![s_1]\!] * \mathcal{S}[\![s_2]\!]) \circledast \subseteq \leftrightarrows \mathtt{P}\ \mathcal{S}[\![s_1]\!] \mathrel{*\!\!\rightarrow} \mathtt{P}\ \mathcal{S}[\![s_2]\!] \circledast \dot{\subseteq}$, $\mathcal{T}[\![\mapsto[s_1, s_2]]\!] \triangleq \mathtt{P}\ (\mathcal{S}[\![s_1]\!] \mathrel{*\!\!\rightarrow} \mathcal{S}[\![s_2]\!]) \circledast \subseteq \leftrightarrows \mathtt{P}\ \mathcal{S}[\![s_1]\!] \mathrel{*\!\!\rightarrow} \mathtt{P}\ \mathcal{S}[\![s_2]\!] \circledast \dot{\subseteq}$, $\mathcal{T}[\![\times[s_1, s_2]]\!] \triangleq \mathtt{P}\ (\mathcal{S}[\![s_1]\!] \mathrel{*\!\!\rightarrow} \mathcal{S}[\![s_2]\!]) \circledast \subseteq \leftrightarrows \mathcal{S}[\![s_1]\!] \mathrel{*\!\!\rightarrow} \mathtt{P}\ \mathcal{S}[\![s_2]\!] \circledast \dot{\subseteq}$, $\mathcal{T}[\![\mathsf{R}[g]]\!] \triangleq \mathcal{T}[\![g]\!]$, $\mathcal{T}[\![s \twoheadrightarrow g]\!] \triangleq \mathcal{S}[\![s]\!] \mathrel{*\!\!\twoheadrightarrow} \mathcal{T}[\![g]\!], \ldots$

Examples of type errors are $\mathcal{T}[\![\top[p, e]]\!] \triangleq (\!\!| \mathcal{E}[\![e]\!] \neq \mathtt{err} \wedge \exists \mathsf{S} \in \mathfrak{S}, \mathsf{O} \in \mathfrak{O} : \mathcal{P}[\![p]\!] = \mathsf{S} \circledast \mathsf{O} \wedge \mathcal{E}[\![e]\!] \Subset \mathsf{S} \mathbin{?} \mathcal{P}[\![p]\!] \leftrightarrows \mathcal{P}[\![p]\!] \mathbin{:} \mathtt{err} |\!\!)$ or $\mathcal{T}[\![\mathbb{I}[\langle s,\ o \rangle, b, t]]\!] \triangleq (\!\!| \mathtt{err} \neq \mathcal{E}[\![b]\!] \Subset \mathcal{S}[\![s]\!] \neq \mathtt{err} \wedge \mathtt{err} \neq \mathcal{E}[\![t]\!] \Subset \mathcal{S}[\![s]\!] \mathbin{?} (\mathtt{P}\ \mathcal{S}[\![s]\!] \circledast \subseteq) \leftrightarrows (\mathtt{P}\ \mathcal{S}[\![s]\!] \circledast \subseteq) \mathbin{:} \mathtt{err} |\!\!)$ where $\Subset$ abstracts set membership $\in$ of top/botton elements to the abstracted set.

This functional presentation is equivalent to a rule-based system *e.g.* $\dfrac{g_1 \vdash \mathsf{P}_1 \leftrightarrows \mathsf{P}_2,\ g_2 \vdash \mathsf{P}_3 \leftrightarrows \mathsf{P}_4,\ \mathsf{P}_2 {\cong} \mathsf{P}_3}{g_1 {\fatsemi} g_2 \vdash \mathsf{P}_1 \leftrightarrows \mathsf{P}_4}$ (where $\mathtt{err}$ is not derivable), $\dfrac{g_1 \vdash \mathsf{S}_1 \circledast \mathsf{O}_1 \leftrightarrows \mathsf{S}_2 \circledast \mathsf{O}_2,\ g_2 \vdash \mathsf{S}_3 \circledast \mathsf{O}_3 \leftrightarrows \mathsf{S}_4 \circledast \mathsf{O}_4}{g_1 {\Longrightarrow} g_2 \vdash \mathsf{S}_1 {\mathrel{*\!\!\rightarrow}} \mathsf{S}_3 \circledast \dot{\mathsf{O}_3} \leftrightarrows \mathsf{S}_2 {\mathrel{*\!\!\rightarrow}} \mathsf{S}_4 \circledast \dot{\mathsf{O}_4}}$, *Id.* for $\divideontimes$.

For example $\mathcal{T}[\![G^{\mathfrak{I}^*}]\!] = \mathtt{P}\ (\mathtt{P}\ (\mathtt{seq}(\mathtt{P}\ \mathtt{lab} * (\mathtt{P}\ \mathtt{var} \mathrel{*\!\!\rightarrow} \mathtt{P}\ \mathtt{int})))) \circledast \subseteq \leftrightarrows (\mathtt{P}\ \mathtt{lab} \mathrel{*\!\!\rightarrow} \mathtt{P}\ \mathtt{var} \mathrel{*\!\!\rightarrow} \mathtt{P}\ \mathtt{P}\ \mathtt{int} \circledast \ddot{\subseteq})$ *i.e.* sets of sets of sequences of states are abstracted to a map of labels to variables to sets of integers (which includes intervals), ordered pointwise.

10. Type soundness Typable expressions $g \in \mathbb{G}$ (for which $\mathcal{T}[\![g]\!] \neq \mathtt{err}$) cannot go wrong since then $\mathcal{S}[\![g]\!] \in \gamma^{\mathfrak{T}}(\mathcal{T}[\![g]\!]) \cup \{\omega\}$ and $\Omega \notin \gamma^{\mathfrak{T}}(\mathcal{T}[\![g]\!])$. However, dynamic errors ($\mathcal{S}[\![g]\!] = \omega$) cannot be excluded (*e.g.* $\mathtt{int}$ does not prevent overflows).

11. Principal type Arbitrary concrete properties in $\wp(\mathfrak{Gc})$ may have no best abstraction (*e.g.* $\emptyset$ so we add the empty type $\varnothing$). Yet, by considering only semantic properties $P = \{\mathcal{S}[\![g_i]\!] \mid i \in \Delta\}$ of *GC* expressions, the principal type is $\alpha^{\mathfrak{T}}(\emptyset) \triangleq \varnothing$, $\alpha^{\mathfrak{T}}(P) \triangleq [\mathsf{T}]_{\cong}$ when $\forall i \in \Delta \neq \emptyset : \mathcal{T}[\![g_i]\!] \cong \mathsf{T}$ else $\alpha^{\mathfrak{T}}(P) \triangleq \mathtt{err}$ so $\langle \wp(\{\mathcal{S}[\![g]\!] \mid g \in \mathbb{G}\}),\ \subseteq \rangle \xleftrightarrows[\alpha^{\mathfrak{T}}]{\gamma^{\mathfrak{T}}} \langle (\mathfrak{T} \cup \{\varnothing\})_{/\cong},\ \trianglelefteq \rangle$ ($\alpha^{\mathfrak{T}}$ onto).

12. Types of types Types $\mathcal{T} \triangleq \{\mathfrak{E}, \mathfrak{S}, \mathfrak{O}, \mathfrak{P}, \mathfrak{T}\}$ have properties $\mathcal{P} \triangleq \wp(\bigcup \mathcal{T})$ can be abstracted to types of types $\overline{\mathfrak{T}} ::= \overline{\varnothing} \mid \overline{\mathfrak{E}} \mid \overline{\mathfrak{S}} \mid \overline{\mathfrak{O}} \mid \overline{\mathfrak{P}} \mid \overline{\mathfrak{T}} \mid \overline{\mathtt{err}}$ by $\alpha^{\overline{\mathfrak{T}}}(P) \triangleq (\!\!| P = \emptyset \mathbin{?} \overline{\varnothing} \parallel P \subseteq \mathsf{T}, \mathsf{T} \in \mathcal{T} \mathbin{?} \overline{\mathsf{T}} \mathbin{:} \overline{\mathtt{err}} |\!\!)$.

13. Static analyzers In static analyzers [1, 12, 14] *GCs* specify abstract domains modules and Galois connectors their combinations by functors. For scalability, rapid convergence acceleration of infinite fixpoint computations by widening/narrowing abstracting induction and/or their duals for co-induction [3–5] is effective and more precise than finite abstractions [8].

Acknowledgments We warmly thank the ACM SIGPLAN Awards Committee for awarding us the 2013 Programming Languages Achievement Award and the programming languages community for its support.

References

[1] B. Blanchet, P. Cousot, R. Cousot, J. Feret, L. Mauborgne, A. Miné, D. Monniaux, and X. Rival. A static analyzer for large safety-critical software. *PLDI'03*, 196–207.

[2] P. Cousot. Types as abstract interpretations. *POPL'97*, 316–331.

[3] P. Cousot and R. Cousot. Static determination of dynamic properties of programs. *Proc. 2nd Int. Symp. on Programming*, 106–130, Paris, 1976. Dunod.

[4] P. Cousot and R. Cousot. Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints. *POPL'77*, 238–252.

[5] P. Cousot and R. Cousot. Static determination of dynamic properties of recursive procedures. *IFIP Conf. on Formal Description of Programming Concepts, St-Andrews, N.B., CN*, 237–277. North-Holland Pub. Co., 1977.

[6] P. Cousot and R. Cousot. Systematic design of program analysis frameworks. *POPL'79*, 269–282.

[7] P. Cousot and R. Cousot. Abstract interpretation frameworks. *J. Logic and Comp.*, 2(4):511–547, 1992.

[8] P. Cousot and R. Cousot. Comparing the Galois connection and widening/narrowing approaches to abstract interpretation. *PLILP'92*, LNCS 631, 269–295.

[9] P. Cousot and R. Cousot. Inductive definitions, semantics and abstract interpretation. *POPL'92*, 83–94.

[10] P. Cousot and R. Cousot. Temporal abstract interpretation. *POPL'00*, 12–25.

[11] P. Cousot and R. Cousot. Systematic design of program transformation frameworks by abstract interpretation. *POPL'02*, 178–190.

[12] P. Cousot and R. Cousot. An abstract interpretation-based framework for software watermarking. *POPL'04*, 173–185.

[13] P. Cousot and R. Cousot. An abstract interpretation framework for termination. *POPL'12*, 245–258.

[14] P. Cousot, R. Cousot, and F. Logozzo. A parametric segmentation functor for fully automatic and scalable array content analysis. *POPL'11*, 105–118.

[15] P. Cousot and N. Halbwachs. Automatic discovery of linear restraints among variables of a program. *POPL'78*, 84–96.