A^2I

ABSTRACT^2 INTERPRETATION

Patrick Cousot
NYU

Roberto Giacobazzi
U. of Verona & IMDEA

Francesco Ranzato
U. of Padova

POPL 2019

What is invariant in these papers?

- Bourdoncle, Abstract interpretation by dynamic partitioning, JFP, 1992
- Venet, Abstract cofibered domains: application to the alias analysis of untyped programs, SAS, 1996
- Blanchet, Cousot, Cousot, Feret, Mauborgne, Miné, Monniaux, and Rival, A static analyzer for large safety-critical software. PLDI, 2003
- Halbwachs, Merchat, and Parent-Vigouroux, Cartesian factoring of polyhedra in linear relation analysis, SAS, 2003
- Bagnara, Hill, Ricci, Zaffanella. Precise widening operators for convex polyhedra, SCP, 2005
- Halbwachs, Merchat, and Gonnord, Some ways to reduce the space dimension in polyhedra computations, FMSD, 2006
- Giacobazzi, Logozzo, and Ranzato, Analyzing program analyses, POPL, 2015
- Cadar and Donaldson, Analysing the program analyser, ICSE, 2016
- Heo, Oh, and Yang, Learning a variable-clustering strategy for octagon from labeled data generated by a static analysis, SAS 2016
- Oh, Lee, Heo, Yang, and Yi, Selective X-sensitive analysis guided by impact pre-analysis, TOPLAS, 2016
- Lee, Lee, Kang, Heo, Oh, and Yi, Sound non-statistical clustering of static analysis alarms, TOPLAS, 2017
- Li, Berenger, Chang, and Rival, Semantic-directed clumping of disjunctive abstract states, POPL, 2017
- Singh, Püschel, and Vechev, Making numerical program analysis fast, PLDI, 2015
- Singh, Püschel, and Vechev, Fast polyhedra abstract domain, POPL, 2017
- Singh, Püschel, and Vechev, A practical construction for decomposing numerical abstract domains, POPL, 2018

What is invariant in these papers?

- Bourdoncle, Abstract interpretation by dynamic partitioning, JFP, 1992
- Venet, Abstract cofibered domains: application to the alias analysis of untyped programs, SAS, 1996
- Blanchet, Cousot, Cousot, Feret, Mauborgne, Miné, Monniaux, and Rival, A static analyzer for large safety-critical software. PLDI, 2003
- Halbwachs, Merchat, and Parent-Vigouroux, Cartesian factoring of polyhedra in linear relation analysis, SAS, 2003
- Bagnara, Hill, Ricci, Zaffanella. Precise widening operators for convex polyhedra, SCP, 2005
- Halbwachs, Merchat, and Gonnord, Some ways to reduce the space dimension in polyhedra computations, FMSD, 2006
- Giacobazzi, Logozzo, and Ranzato, Analyzing program analyses, POPL, 2015
- Cadar and Donaldson, Analysing the program analyser, ICSE, 2016
- Heo, Oh, and Yang, Learning a variable-clustering strategy for octagon from labeled data generated by a static analysis, SAS 2016
- Oh, Lee, Heo, Yang, and Yi, Selective X-sensitive analysis guided by impact pre-analysis, TOPLAS, 2016
- Lee, Lee, Kang, Heo, Oh, and Yi, Sound non-statistical clustering of static analysis alarms, TOPLAS, 2017
- Li, Berenger, Chang, and Rival, Semantic-directed clumping of disjunctive abstract states, POPL, 2017
- Singh, Püschel, and Vechev, Making numerical program analysis fast, PLDI, 2015
- Singh, Püschel, and Vechev, Fast polyhedra abstract domain, POPL, 2017
- Singh, Püschel, and Vechev, A practical construction for decomposing numerical abstract domains, POPL, 2018

What is invariant in these papers?

- Bourdoncle, Abstract interpretation by dynamic partitioning, JFP, 1992
- Venet, Abstract cofibered domains: application to the alias analysis of untyped programs, SAS, 1996
- Blanchet, Cousot, Cousot, Feret, Mauborgne, Miné, Monniaux, and Rival, A static analyzer for large safety-critical software. PLDI, 2003
- Halbwachs, Merchat, and Parent-Vigouroux, Cartesian factoring of polyhedra in linear relation analysis, SAS, 2003
- Bagnara, Hill, Ricci, Zaffanella. Precise widening operators for convex polyhedra, SCP, 2005
- Halbwachs, Merchat, and Gonnord, Some ways to reduce the space dimension in polyhedra computations, FMSD, 2006
- Giacobazzi, Logozzo, and Ranzato, Analyzing program analyses, POPL, 2015
- Cadar and Donaldson, Analysing the program analyser, ICSE, 2016
- Heo, Oh, and Yang, Learning a variable-clustering strategy for octagon from labeled data generated by a static analysis, SAS 2016
- Oh, Lee, Heo, Yang, and Yi, Selective X-sensitive analysis guided by impact pre-analysis, TOPLAS, 2016
- Lee, Lee, Kang, Heo, Oh, and Yi, Sound non-statistical clustering of static analysis alarms, TOPLAS, 2017
- Li, Berenger, Chang, and Rival, Semantic-directed clumping of disjunctive abstract states, POPL, 2017
- Singh, Püschel, and Vechev, Making numerical program analysis fast, PLDI, 2015
- Singh, Püschel, and Vechev, Fast polyhedra abstract domain, POPL, 2017
- Singh, Püschel, and Vechev, A practical construction for decomposing numerical abstract domains, POPL, 2018

They analyze the analysis!

A generic abstract interpreter and its semantics

Generic abstract interpreter (classical)

For a given program P and initial iterate $X^0 \in D$

$$\mathbf{A}[\![\mathrm{P}]\!](X^0) \triangleq \begin{array}{l} X \texttt{ := } X^0\texttt{; k := 0;} \\ \texttt{while } (\neg C\,(X)) \\ \quad \texttt{\{ } X \texttt{ := } F\quad(X)\texttt{; k := k + 1; \}} \end{array}$$

where, at iteration $k \in \mathbb{N}$,

D		abstract domain
C	$\in D \longrightarrow \mathbb{B}$	convergence
F	$\in D \longrightarrow D$	transformer

Generic abstract interpreter (generalized)

For a given program P and initial iterate $X^0 \in D^0$

$$
\begin{array}{lll}
\mathbf{A}[\![\mathrm{P}]\!](X^0) & \triangleq & X \mathtt{\ :=\ } X^0\mathtt{;\ k\ :=\ 0;} \\
& & \mathtt{while\ (}\neg C^k(X)\mathtt{)} \\
& & \quad \mathtt{\{\ } X \mathtt{\ :=\ } F^{k+1}(X)\mathtt{;\ k\ :=\ k\ +\ 1;\ \}}
\end{array}
$$

where, at iteration $k \in \mathbb{N} \cup \{\omega\}$,

D^k			abstract domain at iteration k
C^k	$\in$	$D^k \longrightarrow \mathbb{B}$	convergence at iteration k
F^{k+1}	$\in$	$D^k \longrightarrow D^{k+1}$	transformer at iteration k
F^ω	$\in$	$\langle D^k,\ k \in \mathbb{N}\rangle \longrightarrow D^\omega$	limit transformer

Examples of abstract interpreters

- The generic interpreter can be instantiated to define the semantics of programs

- Example: denotational semantics
  - D^k is a dcpo $\langle D, \sqsubseteq, \bot, \sqcup \rangle$
  - $X^0 = \bot$
  - F^{k+1} is a Scott continuous transformer F
  - $C^k(X) \quad \triangleq \quad ff$
  - $F^\omega(\langle X^k, k \in \mathbb{N} \rangle) \quad \triangleq \quad \bigsqcup_{k \in \mathbb{N}} X^k \quad = \quad \mathsf{lfp}^{\sqsubseteq} F$

- The generic interpreter can be instantiated to define dynamic/static analyzes of programs

- Example: widening abstract interpreter
  - $F^{k+1}(X) \quad \triangleq \quad X \nabla^k F(X)$
  - the widening ∇^k may change during iteration (e.g. delayed widening, moving thresholds, etc.)

Trace semantics of the generic abstract interpreter

$$X^{k+1} \triangleq F^{k+1}(X^k) \in D^{k+1}$$

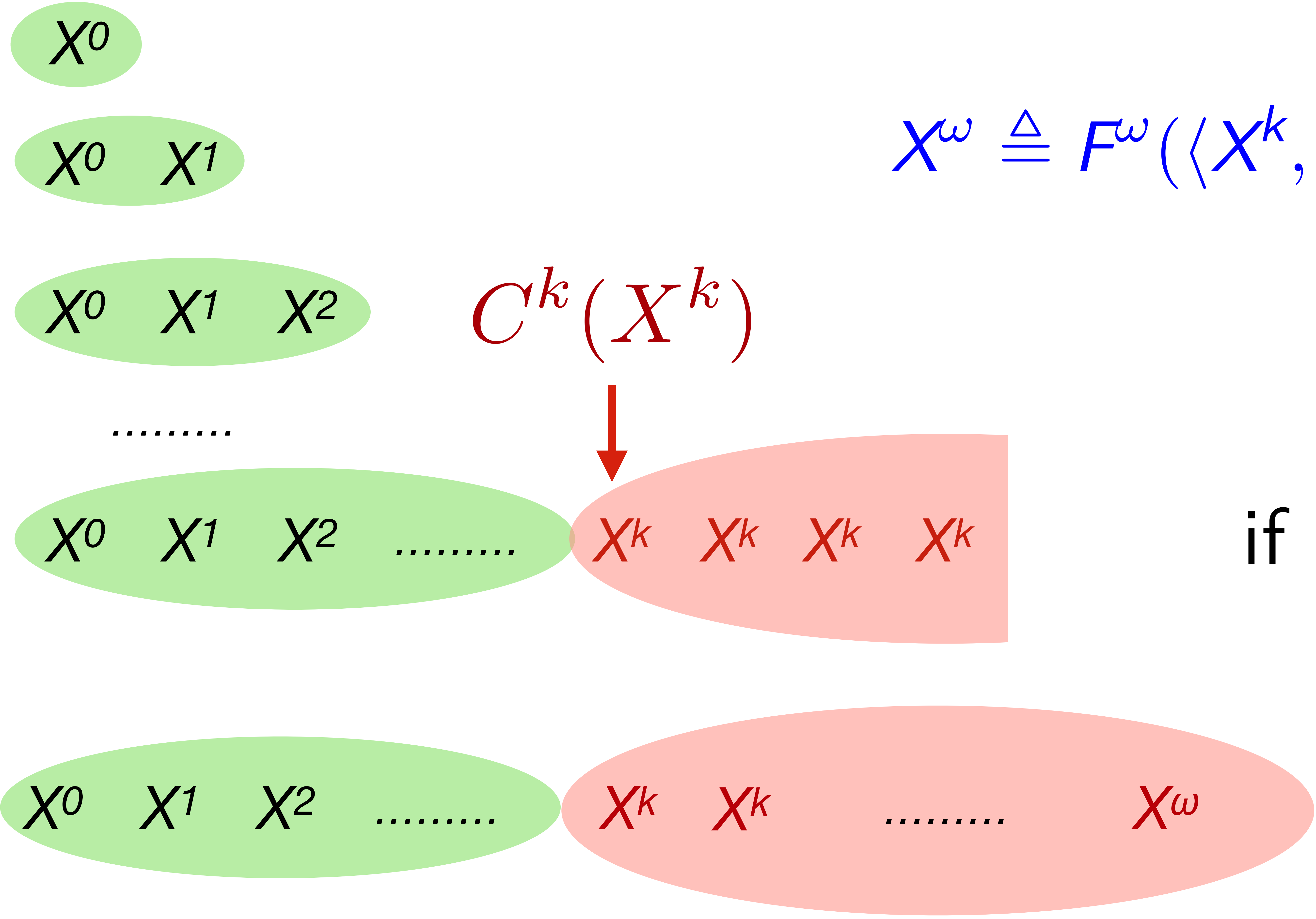
$$X^{\omega} \triangleq F^{\omega}(\langle X^k,\ k \in \mathbb{N}\rangle) \in D^{\omega}$$

$$\neg C^0(X^0)$$

$$\neg C^0(X^0) \wedge \neg C^1(X^1)$$

$$\bigwedge_{i=0}^{2} \neg C^i(X^i)$$

$$\bigwedge_{i=0}^{k-1} \neg C^i(X^i)$$



if stable

Trace semantics of the generic abstract interpreter

$$X^{k+1} \triangleq F^{k+1}(X^k) \in D^{k+1}$$

$$X^{\omega} \triangleq F^{\omega}(\langle X^k,\ k \in \mathbb{N}\rangle) \in D^{\omega}$$

$\neg C^0(X^0)$ — X^0

$\neg C^0(X^0) \wedge \neg C^1(X^1)$ — $X^0 \quad X^1$

$\bigwedge_{i=0}^{2} \neg C^i(X^i)$ — $X^0 \quad X^1 \quad X^2$

....

$\bigwedge_{i=0}^{k-1} \neg C^i(X^i)$ — $X^0 \quad X^1 \quad X^2 \quad \ldots\ldots\ldots \quad X^{k-1}$

....

$\bigwedge_{i<\omega} \neg C^i(X^i)$ — $X^0 \quad X^1 \quad X^2 \quad \ldots\ldots\ldots \quad X^k \quad X^{k+1} \quad \ldots\ldots\ldots \quad X^{\omega}$ if unstable

Hierarchy of abstract interpreters

- The semantics of the generic abstract interpreter is an instance of the generic abstract interpreter

- The collecting semantics of the generic abstract interpreter is an instance of the generic abstract interpreter

- A sound abstraction of an instance of the generic interpreter is an instance of the generic abstract interpreter

→ the generic interpreter can be used to analyze an instance of the generic interpreter

$\mathrm{A}^2\mathrm{I}$: Abstract^2 Interpretation

How it works with a simple example: Analysis

Program: `x=0; while` $^{\ell_1}$ `(true) { x=x+2;` $^{\ell_2}$ `}`

Interval equations: $\left\{\begin{array}{lclcl} X_1 & = & F_1(X_1, X_2) & \triangleq & [0,0] \sqcup X_2 \\ X_2 & = & F_2(X_1, X_2) & \triangleq & X_1 \oplus [2,2] \end{array}\right.$

Jacobi iterates (no widening):

$$\begin{bmatrix}\bot\\ \bot\end{bmatrix}, \begin{bmatrix}[0,0]\\ \bot\end{bmatrix}, \begin{bmatrix}[0,0]\\ [2,2]\end{bmatrix}, \begin{bmatrix}[0,2]\\ [2,2]\end{bmatrix}, \ldots, \begin{bmatrix}[0,2n]\\ [2,2n]\end{bmatrix}, \begin{bmatrix}[0,2n]\\ [2,2(n+1)]\end{bmatrix}, \begin{bmatrix}[0,2(n+1)]\\ [2,2(n+1)]\end{bmatrix}, \ldots, \begin{bmatrix}[0,\infty]\\ [2,\infty]\end{bmatrix}$$

How it works with a simple example: Meta-collecting semantics

Equations of the collecting semantics:

$$\begin{cases} \overline{X}_1 = \overline{F}_1(\overline{X}_1, \overline{X}_2) \triangleq \overline{X}_1 \bullet ([0,0] \sqcup \mathrm{last}(\overline{X}_2)) \\ \overline{X}_2 = \overline{F}_2(\overline{X}_1, \overline{X}_2) \triangleq \overline{X}_2 \bullet (\mathrm{last}(\overline{X}_1) \oplus [2,2]) \end{cases}$$

Jacobi iterates of the collecting semantics:

$$\begin{bmatrix}\bot\\ \bot\end{bmatrix}, \begin{bmatrix}\bot \bullet [0,0] \\ \bot \bullet \bot\end{bmatrix}, \begin{bmatrix}\bot \bullet [0,0] \bullet [0,0] \\ \bot \bullet \bot \bullet [2,2]\end{bmatrix}, \begin{bmatrix}\bot \bullet [0,0] \bullet [0,0] \bullet [0,2] \\ \bot \bullet \bot \bullet [2,2] \bullet [2,2]\end{bmatrix},$$

$$\begin{bmatrix}\bot \bullet [0,0] \bullet [0,0] \bullet [0,2] \bullet [0,2] \\ \bot \bullet \bot \bullet [2,2] \bullet [2,2] \bullet [2,4]\end{bmatrix}, \ldots, \begin{bmatrix}\bot \bullet [0,0] \bullet [0,0] \bullet [0,2] \bullet \cdots \bullet [0,2n] \\ \bot \bullet \bot \bullet [2,2] \bullet [2,2] \bullet \cdots \bullet [2,2(n+1)]\end{bmatrix},$$

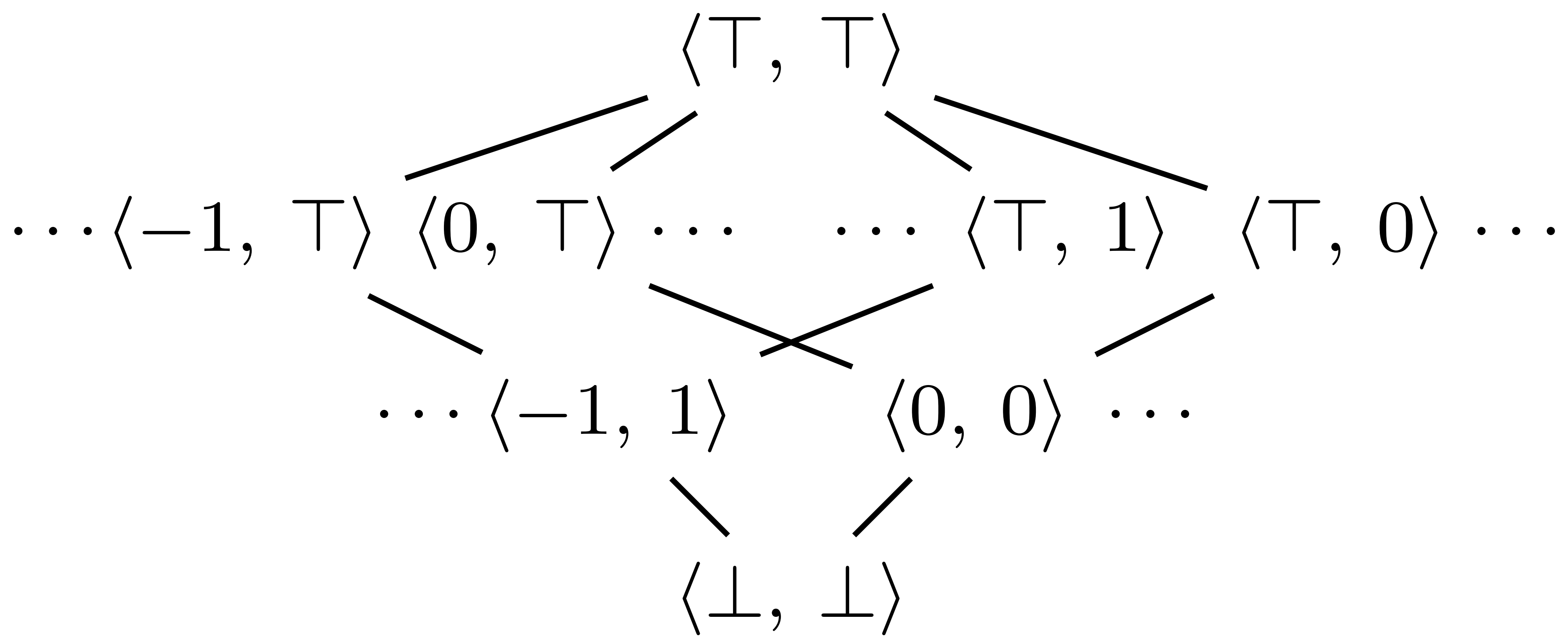
$$\begin{bmatrix}\bot \bullet [0,0] \bullet [0,0] \bullet [0,2] \bullet \cdots \bullet [0,2n] \bullet [0,2(n+1)] \\ \bot \bullet \bot \bullet [2,2] \bullet [2,2] \bullet \cdots \bullet [2,2(n+1)] \bullet [2,2(n+1)]\end{bmatrix}, \ldots$$

Limit of the collecting iterates:

$$\begin{bmatrix}\bot \bullet [0,0] \bullet [0,0] \bullet [0,2] \bullet \cdots \bullet [0,2n] \bullet \cdots \\ \bot \bullet \bot \bullet [2,2] \bullet [2,2] \bullet \cdots \bullet [2,2n] \bullet \cdots\end{bmatrix}_{n \geq 1}$$

How it works with a simple example: Meta-analysis

Abstraction domain for the iterates:



Abstraction:

$$\alpha^2(\langle \overline{X}_1,\ \overline{X}_2\rangle) \triangleq \langle \alpha(\overline{X}_1),\ \alpha(\overline{X}_2)\rangle$$

$$\alpha(\bot \bullet [\ell_1, h_1] \bullet [\ell_2, h_2] \bullet \cdots \bullet [\ell_n, h_n]) \triangleq \langle \bigsqcup_{i=1}^{n} \ell_i,\ \bigsqcup_{i=1}^{n} h_i \rangle$$

Equations of the meta analysis:

$$\begin{cases} \langle l_1,\ h_1\rangle = F_1(\langle l_1,\ h_1\rangle, \langle l_2,\ h_2\rangle) \triangleq \langle l_1 \sqcup 0 \sqcup \min(0, l_2),\ h_1 \sqcup 0 \sqcup \max(0, h_2)\rangle \\ \langle l_2,\ h_2\rangle = F_2(\langle l_1,\ h_1\rangle, \langle l_2,\ h_2\rangle) \triangleq \langle l_2 \sqcup (l_1 \oplus^c 2),\ h_2 \sqcup (h_1 \oplus^c 2)\rangle \end{cases}$$

Iterates of the meta analysis:

$$\begin{bmatrix}\langle \bot,\ \bot\rangle \\ \langle \bot,\ \bot\rangle\end{bmatrix}, \begin{bmatrix}\langle 0,\ 0\rangle \\ \langle \bot,\ \bot\rangle\end{bmatrix}, \begin{bmatrix}\langle 0,\ 0\rangle \\ \langle 2,\ 2\rangle\end{bmatrix}, \begin{bmatrix}\langle 0,\ \top\rangle \\ \langle 2,\ 2\rangle\end{bmatrix}, \begin{bmatrix}\langle 0,\ \top\rangle \\ \langle 2,\ \top\rangle\end{bmatrix}$$

The meta-analysis provides a widening for the analysis

Calculational design of the abstract meta-interpreter

A.1 Calculational design of the meta abstract interpreter of Section 4

Proof. The Jacobi iterates of (2) belong to $\mathcal{X} = \left\{ \begin{bmatrix} \bot \cdot [\ell_1^1, h_1^1] \cdot [\ell_1^2, h_1^2] \cdot \ldots \cdot [\ell_1^n, h_1^n] \\ \bot \cdot [\ell_2^1, h_2^1] \cdot [\ell_2^2, h_2^2] \cdot \ldots \cdot [\ell_2^m, h_2^m] \end{bmatrix} \;\middle|\; n, m \geqslant 0 \right\}$. The Jacobi iterates of (3) belong to $\overline{\mathcal{X}} = \left\{ \begin{bmatrix} \langle \ell_1,\ h_1 \rangle \\ \langle \ell_2,\ h_1 \rangle \end{bmatrix} \;\middle|\; \ell_1, h_1, \ell_2, h_1 \in \mathcal{D}_c \right\}$. We have the Galois connection $\langle \mathcal{X},\ {\preccurlyeq_{pf}}^2 \rangle \underset{\alpha_c^2}{\overset{\gamma_c^2}{\leftrightarrows}} \langle \overline{\mathcal{X}},\ {\sqsubseteq_c}^2 \rangle$.

For the semi-commutation condition, let $\overline{X} \in \mathcal{X}$ be an iterate of iterates of (2).

$$\alpha_c^2(\overline{F}(\overline{X}))$$

$$= \begin{bmatrix} \alpha_c(\overline{\bot} \curlyvee \llbracket \overline{X}_1 \cdot ([0,0] \sqcup x) \parallel \overline{X}_2 = \overline{X} \cdot x \rrbracket) \\ \alpha_c(\overline{\bot} \curlyvee \llbracket \overline{X}_2 \cdot (x \oplus [2,2]) \parallel \overline{X}_1 = \overline{X} \cdot x \rrbracket) \end{bmatrix} \qquad \wr\text{def. } \alpha_c^2 \int$$

Let us calculate the first term.

$$\alpha_c(\overline{\bot} \curlyvee \llbracket \overline{X}_1 \cdot ([0,0] \sqcup x) \parallel \overline{X}_2 = \overline{X} \cdot x \rrbracket)$$

$$= \langle \bot_c,\ \bot_c \rangle \sqcup_c^2 \alpha_c(\llbracket \overline{X}_1 \cdot ([0,0] \sqcup x) \parallel \overline{X}_2 = \overline{X} \cdot x \rrbracket)$$

≀in a Galois connection, α_c preserves existing joins∫

$$= \alpha_c(\llbracket \overline{X}_1 \cdot ([0,0] \sqcup x) \parallel \overline{X}_2 = \overline{X} \cdot x \rrbracket) \qquad \wr\text{def. infimum}\int$$

$$= \alpha_c(\llbracket \overline{X}_1 \cdot ([0,0] \sqcup \llbracket m = 0 \mathrel{?} \bot \mathrel{⁏} [\ell_2^m, h_2^m] \rrbracket) \rrbracket)$$

≀by def. of the set $\mathcal{X}$ of iterates, $\overline{X}_2$ has the form $\bot \cdot [\ell_2^1, h_2^1] \cdot [\ell_2^2, h_2^2] \cdot \ldots \cdot [\ell_2^m, h_2^m]$ where $m > 0$ and $\overline{X} = \bot \cdot [\ell_2^1, h_2^1] \cdot [\ell_2^2, h_2^2] \cdot \ldots \cdot [\ell_2^{m-1}, h_2^{m-1}]$, or $\overline{X}_2 = \bot$ with $\overline{X} = \ni$ is the empty sequence whenever $m = 0$∫

$$= \alpha_c(\overline{X}_1) \sqcup_c^2 \llbracket m = 0 \mathrel{?} \alpha_c([0,0] \sqcup \bot) \mathrel{⁏} \alpha_c([0,0] \sqcup [\ell_2^m, h_2^m]) \rrbracket \qquad \wr\text{def. } \alpha_c \text{ and conditional}\int$$

$$= \llbracket m = 0 \mathrel{?} \alpha_c(\overline{X}_1) \sqcup_c^2 \alpha_c([0,0]) \mathrel{⁏} \alpha_c(\overline{X}_1) \sqcup_c^2 \alpha_c([\min(0, \ell_2^m), \max(0, h_2^m)]) \rrbracket$$

≀def. infimum $\bot$, join $\sqcup$ in intervals, and def. conditional∫

$$\sqsubseteq_c^2 \llbracket m = 0 \mathrel{?} \alpha_c(\overline{X}_1) \sqcup_c^2 \alpha_c([0,0]) \mathrel{⁏} \alpha_c(\overline{X}_1) \sqcup_c^2 \alpha_c([0,0] \sqcup [\min(0, \ell_2^m), \max(0, h_2^m)]) \rrbracket$$

≀since $[0,0] \sqsubseteq [\min(0, \ell_2^m), \max(0, h_2^m)]$ and α_c is increasing∫

$$= \llbracket m = 0 \mathrel{?} \alpha_c(\overline{X}_1) \sqcup_c^2 \langle 0,\ 0 \rangle \mathrel{⁏} \alpha_c(\overline{X}_1) \sqcup_c^2 \langle 0,\ 0 \rangle \sqcup_c^2 \alpha_c([\min(0, \ell_2^m), \max(0, h_2^m)]) \rrbracket$$

≀α_c preserves existing joins and def. α_c so that $\alpha_c([0,0]) = \langle 0,\ 0 \rangle$∫

$$= \alpha_c(\overline{X}_1) \sqcup_c^2 \langle 0,\ 0 \rangle \sqcup_c^2 \llbracket m = 0 \mathrel{?} \langle \bot_c,\ \bot_c \rangle \mathrel{⁏} \alpha_c([\min(0, \ell_2^m), \max(0, h_2^m)]) \rrbracket$$

≀factorizing $\alpha_c(\overline{X}_1) \sqcup_c^2 \langle 0,\ 0 \rangle$ in the conditional and $\langle \bot_c,\ \bot_c \rangle$ is the infimum for the lub $\sqcup_c^2$∫

$$= \alpha_c(\overline{X}_1) \sqcup_c^2 \langle 0,\ 0 \rangle \sqcup_c^2 \llbracket \langle \min(0, \ell_2),\ \max(0, h_2) \rangle \parallel \alpha_c(\overline{X}_2) = \langle l_2,\ h_2 \rangle \rrbracket$$

≀since if $m = 0$ then $\overline{X}_2$ is $\bot$ hence $\alpha_c(\overline{X}_2) = \langle \bot_c,\ \bot_c \rangle$ so $\langle l_2,\ h_2 \rangle = \langle \bot_c,\ \bot_c \rangle$ and therefore $\langle \min(0, \ell_2),\ \max(0, h_2) \rangle = \langle \bot_c,\ \bot_c \rangle$ by our convention that $\bot_c$ is absorbent for both min and max∫

$$= \llbracket \langle l_1,\ h_1 \rangle \sqcup_c^2 \langle 0,\ 0 \rangle \sqcup_c^2 \langle \min(0, \ell_2),\ \max(0, h_2) \rangle \parallel \alpha_c(\overline{X}_1) = \langle l_1,\ h_1 \rangle, \alpha_c(\overline{X}_2) = \langle l_2,\ h_2 \rangle \rrbracket$$

≀def. let construct∫

$$= \llbracket \langle l_1 \sqcup_c 0 \sqcup_c \min(0, l_2),\ h_1 \sqcup_c 0 \sqcup_c \max(0, h_2) \rangle \parallel \alpha_c(\overline{X}_1) = \langle l_1,\ h_1 \rangle, \alpha_c(\overline{X}_2) = \langle l_2,\ h_2 \rangle \rrbracket$$

≀pairwise def. $\sqcup_c^2$ in $(\mathcal{D}_c)^2$∫

$$= F_1^c(\alpha_c(\overline{X}_2), \alpha_c(\overline{X}_2)) \qquad \wr\text{def. } F_1^c \text{ in (3)}\int$$

Let us calculate the second term.

$$\alpha_c(\overline{\bot} \curlyvee \llbracket \overline{X}_2 \cdot (x \oplus [2,2]) \parallel \overline{X}_1 = \overline{X} \cdot x \rrbracket)$$

$$= \langle \bot_c,\ \bot_c \rangle \sqcup_c^2 \alpha_c(\llbracket \overline{X}_2 \cdot (x \oplus [2,2]) \parallel \overline{X}_1 = \overline{X} \cdot x \rrbracket) \qquad \wr\alpha_c \text{ preserves existing joins}\int$$

$$= \alpha_c(\llbracket \overline{X}_2 \cdot (x \oplus [2,2]) \parallel \overline{X}_1 = \overline{X} \cdot x \rrbracket) \qquad \wr\text{def. infimum}\int$$

$$= \alpha_c(\llbracket n = 0 \mathrel{?} \overline{X}_2 \cdot \bot \mathrel{⁏} \overline{X}_2 \cdot ([\ell_1^n, h_1^n] \oplus [2,2]) \rrbracket)$$

≀by def. of the set $\mathcal{X}$ of iterates, $\overline{X}_1$ has the form $\bot \cdot [\ell_1^1, h_1^1] \cdot [\ell_1^2, h_1^2] \cdot \ldots \cdot [\ell_1^n, h_1^n]$ when $n > 0$ and $\overline{X} = \bot \cdot [\ell_1^1, h_1^1] \cdot [\ell_1^2, h_1^2] \cdot \ldots \cdot [\ell_1^{n-1}, h_1^{n-1}]$, or $n = 0$ so $\overline{X}_1 = \bot$ with $\overline{X} = \ni$ is the empty sequence and $\bot \oplus [2,2] = \bot$∫

$$= \alpha_c(\overline{X}_2 \cdot \llbracket n = 0 \mathrel{?} \bot \mathrel{⁏} ([\ell_1^n + 2, h_1^n + 2]) \rrbracket) \qquad \wr\text{factoring } \overline{X}_2 \text{ and def. } \oplus \text{ for intervals}\int$$

$$= \alpha_c(\overline{X}_2) \sqcup_c^2 \llbracket n = 0 \mathrel{?} \langle \bot_c,\ \bot_c \rangle \mathrel{⁏} ([\ell_1^n + 2, h_1^n + 2]) \rrbracket) \qquad \wr\text{def. } \alpha_c \text{ and } \oplus_c \text{ on } \mathcal{D}_c\int$$

$$= \llbracket \alpha_c(\overline{X}_2) \sqcup_c^2 \langle \ell_1 \oplus_c 2,\ h_1 \oplus_c 2 \rangle \parallel \langle \ell_1,\ h_1 \rangle = \alpha_c(\overline{X}_1) \rrbracket)$$

≀since if $n = 0$ then $\overline{X}_1$ is $\bot$ hence $\alpha_c(\overline{X}_1) = \langle \bot_c,\ \bot_c \rangle$ so $\langle l_1,\ h_1 \rangle = \langle \bot_c,\ \bot_c \rangle$ and therefore $\langle \ell_1 \oplus_c 2,\ h_1 \oplus_c 2 \rangle = \langle \bot_c \oplus_c 2,\ \bot_c \oplus_c 2 \rangle \langle \bot_c,\ \bot_c \rangle$ since $\bot_c$ is absorbent for $\oplus_c$∫

$$= \llbracket \langle \ell_2,\ h_2 \rangle \sqcup_c^2 \langle \ell_1 \oplus_c 2,\ h_1 \oplus_c 2 \rangle \parallel \langle \ell_1,\ h_1 \rangle = \alpha_c(\overline{X}_1), \langle \ell_2,\ h_2 \rangle = \alpha_c(\overline{X}_2) \rrbracket)$$

≀def. let construct∫

$$= \llbracket \langle l_2 \sqcup_c (l_1 \oplus^c 2),\ h_2 \sqcup_c (h_1 \oplus^c 2) \rangle \parallel \langle \ell_1,\ h_1 \rangle = \alpha_c(\overline{X}_1), \langle \ell_2,\ h_2 \rangle = \alpha_c(\overline{X}_2) \rrbracket$$

≀pairwise def. $\sqcup_c^2$ in $(\mathcal{D}_c)^2$∫

$$= F_2^c(\alpha_c(\overline{X}_1), \alpha_c(\overline{X}_2)) \qquad \wr\text{def. } F_2^c \text{ in (3)}\int$$

Grouping the two terms, we have proved the semi-commutation $\alpha_c^2(\overline{F}(\overline{X})) \mathrel{\dot{\sqsubseteq}_c^2} F^c(\alpha_c^2(\overline{X}))$. By Theorem 3.4, we conclude that $\mathsf{lfp}_{\langle \bot,\ \bot \rangle}^{{\preccurlyeq_{pf}}^2} \overline{F} \mathrel{{\preccurlyeq_{pf}}^2} \alpha_c^2(\mathsf{lfp}_{\langle \langle \bot_c, \bot_c \rangle, \langle \bot_c, \bot_c \rangle \rangle}^{\dot{\sqsubseteq}_c^2} F^c)$. □

Meta abstract interpretation

Offline

- before starting the analysis/static/beforehand

Online

- during the analysis/dynamic/on the fly

Offline Meta Abstract Interpretation

Examples of offline meta abstract interpretation

- Widening in interval analysis

  A beforehand constant propagation meta analysis determines which unstable interval bounds should be widened

- Packing in Astrée

  A beforehand meta analysis determines at each program points which packs of variables should be related by octagonal invariants

  Variables in different packs will definitely be not related

Online Meta Abstract Interpretation

Online abstract interpreter

$$\mathbf{A^2}[\![\mathtt{P}]\!](X^0, \quad , \qquad\qquad\qquad\qquad) \triangleq$$

```
X := X⁰; k := 0;
while (¬Cᵏ(X)) {
  X := Fᵏ⁺¹(X); k := k + 1;


}
```

- an instance of the generic abstract interpreter

Online abstract interpreter

$$\mathbf{A^2}[\![\mathtt{P}]\!](X^0, \alpha_{\mathsf{pa}}, \gamma_{\mathsf{pa}} \qquad) \triangleq$$

```
X := X⁰; k := 0; X̄ := α_pa(X⁰);
while (¬Cᵏ(X)) {
  X := Fᵏ⁺¹(X); k := k + 1;
  X̄ := α_pa(γ_pa(X̄) • X);

}
```

- an instance of the generic abstract interpreter
- keeping an abstraction $\overline{X}$ of its iterations

Online meta abstract interpreter

$$\mathbf{A^2}[\![\mathrm{P}]\!](X^0, \alpha_{\mathsf{pa}}, \gamma_{\mathsf{pa}}, D^0, D^1, F^1, C^0) \triangleq$$

$$\begin{array}{l}
\quad X \;\texttt{:=}\; X^0\texttt{;}\;\; \texttt{k} \;\texttt{:=}\; \texttt{0;}\;\; \overline{X} \;\texttt{:=}\; \alpha_{\mathsf{pa}}(X^0)\texttt{;}\\
\quad \texttt{while}\; (\neg C^k(X))\; \texttt{\{}\\
\qquad X \;\texttt{:=}\; F^{k+1}(X)\texttt{;}\;\; \texttt{k} \;\texttt{:=}\; \texttt{k + 1;}\\
\qquad \overline{X} \;\texttt{:=}\; \alpha_{\mathsf{pa}}(\gamma_{\mathsf{pa}}(\overline{X}) \cdot X)\texttt{;}\\
\qquad \langle D^{k+1},\; F^{k+1},\; C^k\rangle \;\texttt{:=}\; \mathbf{MA}[\![\mathrm{P}]\!](\overline{X}, \gamma_{\mathsf{pa}}, D^k, F^k, C^{k-1})\texttt{;}\\
\quad \texttt{\}}
\end{array}$$

- an instance of the generic abstract interpreter
- keeping an abstraction $\overline{X}$ of its iterations
- passed to the meta interpreter to compute the next abstract domain, transformer, and convergence criterion

Online meta abstract interpreter

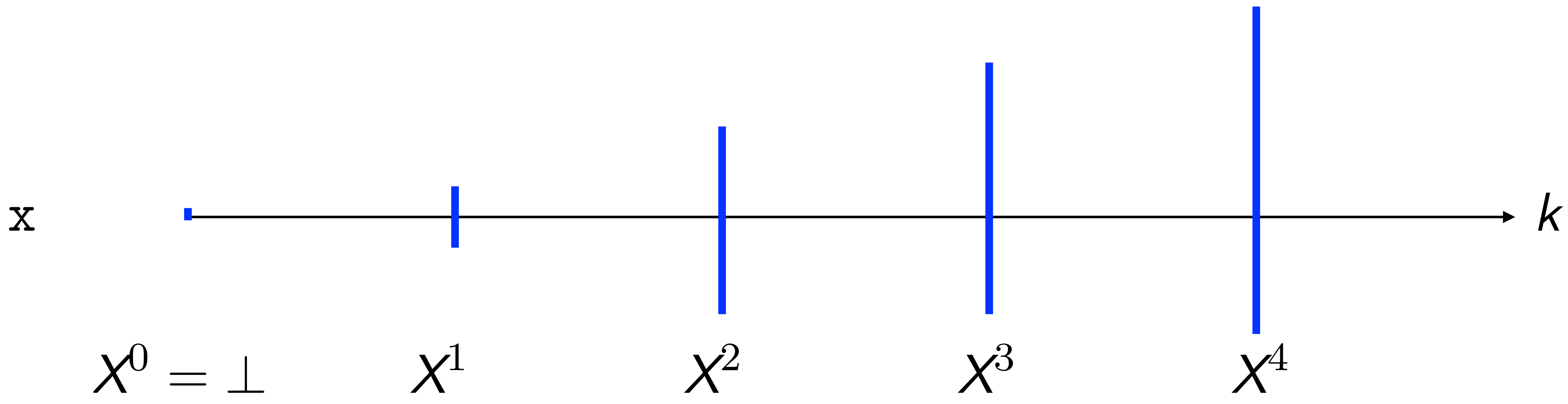
$$
\begin{array}{l}
\mathbf{A^2}[\![\mathrm{P}]\!](X^0, \alpha_{\mathsf{pa}}, \gamma_{\mathsf{pa}}, D^0, D^1, F^1, C^0) \triangleq \\
\quad X \texttt{ := } X^0\texttt{; k := 0; } \overline{X} \texttt{ := } \alpha_{\mathsf{pa}}(X^0)\texttt{;} \\
\quad \texttt{while } (\neg C^k(X)) \texttt{ \{} \\
\quad\quad X \texttt{ := } F^{k+1}(X)\texttt{; k := k + 1;} \\
\quad\quad \overline{X} \texttt{ := } \alpha_{\mathsf{pa}}(\gamma_{\mathsf{pa}}(\overline{X}) \bullet X)\texttt{;} \\
\quad\quad \langle D^{k+1}, F^{k+1}, C^k\rangle \texttt{ := } \mathbf{MA}[\![\mathrm{P}]\!](\overline{X}, \gamma_{\mathsf{pa}}, D^k, F^k, C^{k-1})\texttt{;} \\
\quad \texttt{\}}
\end{array}
$$

$$
\begin{array}{l}
\mathbf{MA}[\![\mathrm{P}]\!](\overline{X}, \gamma_{\mathsf{pa}}, D, F, C,) \triangleq \\
\quad \mathcal{X} \texttt{ := } \langle D, F, C, \gamma_{\mathsf{pa}}(\overline{X})\rangle\texttt{; k := 0;} \\
\quad \texttt{while } (\neg \mathcal{C}^k_{\mathsf{ma}}(\mathcal{X})) \texttt{ \{} \\
\quad\quad \mathcal{X} \texttt{ := } \mathcal{F}^{k+1}_{\mathsf{ma}}(\mathcal{X})\texttt{; k := k+1;} \\
\quad \texttt{\}} \\
\quad \texttt{let } \langle D, F, C, X\rangle = \mathcal{X} \texttt{ in return } \langle D, F, C\rangle\texttt{;}
\end{array}
$$

- an instance of the generic abstract interpreter
- computing the next abstract domain D, transformer F, and convergence criterion C

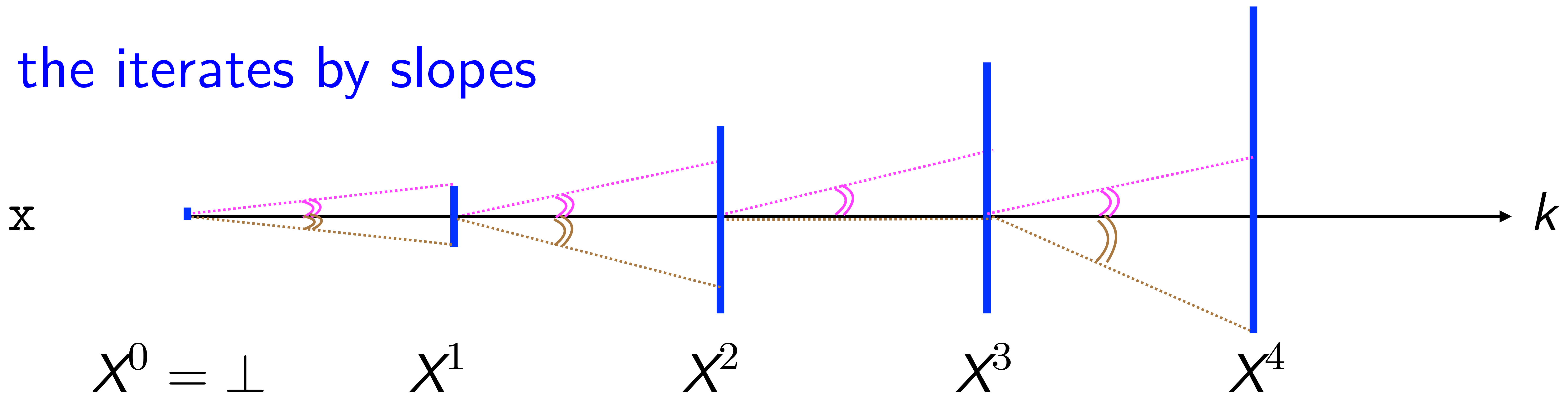
An application of online meta abstract interpretation to the design of a widening

Iterates of the interval abstract interpreter

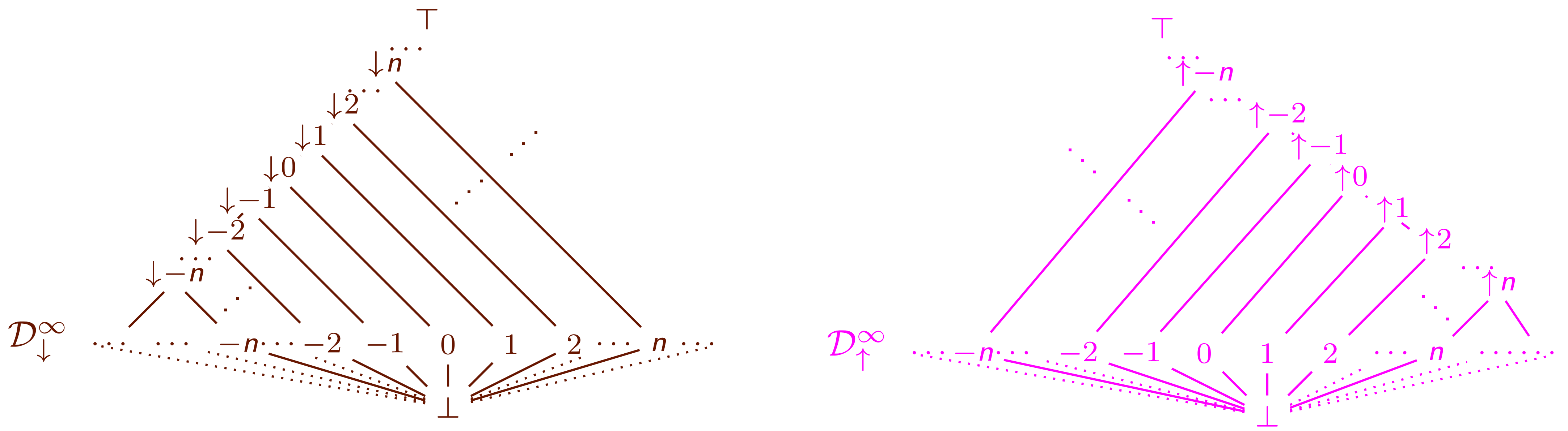


Meta-abstraction of the iterates of the interval abstract interpreter

- Abstract the iterates by slopes



- Abstract sequences of slopes by there maximum



- Enforce convergence of the meta-abstract interpreter by a widening
- An iteration dependent widening is designed using a meta-widening

An application of online meta abstract interpretation to relational domains

Online meta abstract interpreter

- Numerical relational analyzes
  - Can be costly (polynomial (octagons) / exponential (polyhedra) in the number of variables)
  - Cost can be reduced by decomposition into a conjunction of relations on packs of variables such that variables in different packs are unrelated
  - Packs determined offline for Miné's octagons in Astrée, with loss of information
  - Packs determined online for Halbwachs et al's polyhedra, without any loss of information
  - Generalized to octagons and then arbitrary relational numerical domains by Singh, Püschel, and Vechev

Online meta abstract interpreter

- Relational analyzes
  - Generalized to arbitrary relational domains in this paper
  - Example of decomposition:

$$\begin{aligned} & r_1(x_1, x_2) \\ \wedge\ & r_2(x_2, x_3) \\ \wedge\ & r_3(x_3, x_1) \\ \wedge\ & r_4(x_4) \\ \wedge\ & r_5(x_5, x_6) \\ \wedge\ & r_6(x_6) \end{aligned}$$

Online meta abstract interpreter

- Relational analyzes
  - Generalized to arbitrary relational domains in this paper
  - Example of decomposition:

$$
\begin{array}{ll}
\ r_1(x_1, x_2) & \rightarrow\ \{x_1, x_2, x_3\} \\
\wedge\ r_2(x_2, x_3) & \\
\wedge\ r_3(x_3, x_1) & \\
\wedge\ r_4(x_4) & \ \{x_4\} \\
\wedge\ r_5(x_5, x_6) & \ \{x_5, x_6\} \\
\wedge\ r_6(x_6) &
\end{array}
$$

Online meta abstract interpreter

- Relational analyzes
  - Generalized to arbitrary relational domains in this paper
  - Example of decomposition:

$$
\begin{array}{lclcl}
r_1(x_1,x_2) & \to & \{x_1,x_2,x_3\} & \Rightarrow & r_1(x_1,x_2) \\
\wedge\, r_2(x_2,x_3) & & & & \wedge\, r_2(x_2,x_3) \\
\wedge\, r_3(x_3,x_1) & & & & \wedge\, r_3(x_3,x_1) \\
\wedge\, r_4(x_4) & & \{x_4\} & \Rightarrow & \times\, r_4(x_4) \\
\wedge\, r_5(x_5,x_6) & & \{x_5,x_6\} & & \times\, r_5(x_5,x_6) \\
\wedge\, r_6(x_6) & & & \Rightarrow & \wedge\, r_6(x_6)
\end{array}
$$

Online meta abstract interpreter

- Relational analyzes
  - Generalized to arbitrary relational domains in this paper
  - Example of decomposition:

$$
\begin{array}{llllll}
r_1(x_1,x_2) & \rightarrow \{x_1,x_2,x_3\} & \Rightarrow & r_1(x_1,x_2) & \xrightarrow{r(x_2,x_4)} & \{x_1,x_2,x_3,x_4\} \\
\wedge\, r_2(x_2,x_3) & & & \wedge\, r_2(x_2,x_3) & & \\
\wedge\, r_3(x_3,x_1) & & & \wedge\, r_3(x_3,x_1) & & \\
\wedge\, r_4(x_4) & \{x_4\} & \Rightarrow & \times\, r_4(x_4) & & \\
\wedge\, r_5(x_5,x_6) & \{x_5,x_6\} & & \times\, r_5(x_5,x_6) & & \{x_5,x_6\} \\
\wedge\, r_6(x_6) & & \Rightarrow & \wedge\, r_6(x_6) & &
\end{array}
$$

Online meta abstract interpreter

- Relational analyzes
  - Generalized to arbitrary relational domains in this paper
  - Example of decomposition:

$$
\begin{array}{llllllll}
 & & & & & & & r(x_2,x_4)\\
r_1(x_1,x_2) & \rightarrow & \{x_1,x_2,x_3\} & \Rightarrow & r_1(x_1,x_2) & \xrightarrow{r(x_2,x_4)} \{x_1,x_2,x_3,x_4\} & \Rightarrow & \wedge\ r'_1(x_1,x_2)\\
\wedge\ r_2(x_2,x_3) & & & & \wedge\ r_2(x_2,x_3) & & & \wedge\ r'_2(x_2,x_3)\\
\wedge\ r_3(x_3,x_1) & & & & \wedge\ r_3(x_3,x_1) & & & \wedge\ r_3(x_3,x_1)\\
\wedge\ r_4(x_4) & & \{x_4\} & \Rightarrow & \times\ r_4(x_4) & & & \wedge\ r'_4(x_4)\\
\wedge\ r_5(x_5,x_6) & & \{x_5,x_6\} & & \times\ r_5(x_5,x_6) & \{x_5,x_6\} & \Rightarrow & \times\ r_5(x_5,x_6)\\
\wedge\ r_6(x_6) & & & \Rightarrow & \wedge\ r_6(x_6) & & & \wedge\ r_6(x_6)
\end{array}
$$

  - A beautiful example of online meta abstract interpretation: the decomposition hence the abstract domain and the blockwise transformer change at each iteration

More in the paper
(semantics, abstractions, algorithms, etc)

Conclusion

Abstract interpretation

- Dynamic program analysis
- Static program analysis
  - deductive analysis (e.g. Hoare logic)
  - data flow analysis
  - model checking
  - types
  - symbolic execution
  - ...

Abstract interpretation

- Dynamic program analysis
- Static program analysis
  - deductive analysis (e.g. Hoare logic)
  - data flow analysis
  - model checking
  - types
  - symbolic execution
  - ...
- Introspection: $\mathbf{A^2I}$

The end, distinguished thanks