How to Combine Widening and Narrowing for Non-monotonic Systems of Equations

Kalmer Apinis Helmut Seidl Vesal Vojdani

Abstract interpretation (Goblint)

1. Take a CFG of a program.

Abstract interpretation (Goblint)

1. Take a CFG of a program.
2. Pick a complete lattice $(\mathbb{D}, \sqsubseteq)$
   - that represents statements about the program points
   - together with its implication order.

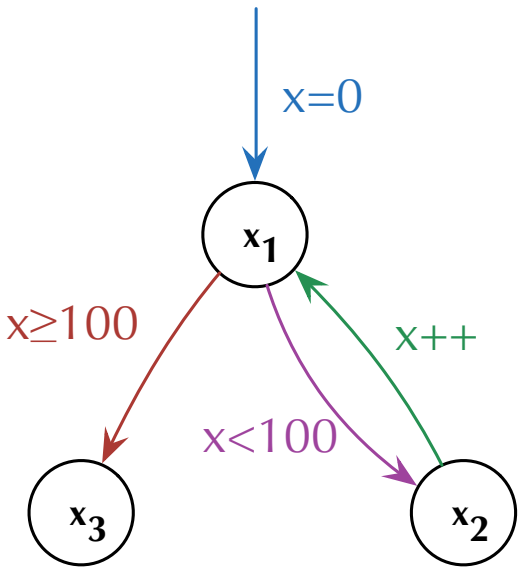
Abstract interpretation (Goblint)

1. Take a CFG of a program.
2. Pick a complete lattice $(\mathbb{D}, \sqsubseteq)$
   - that represents statements about the program points
   - together with its implication order.
3. Generate an equation system.
   - $x_i = f_i(x_1, x_2, \ldots, x_n)$
   - nodes $\rightarrow$ variables of the equation system
   - edges $\rightarrow$ contributions to variables

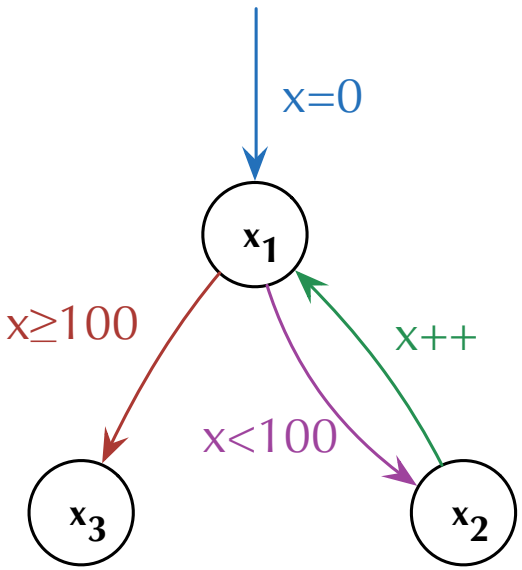
Abstract interpretation (Goblint)

1. Take a CFG of a program.
2. Pick a complete lattice $(\mathbb{D}, \sqsubseteq)$
   - that represents statements about the program points
   - together with its implication order.
3. Generate an equation system.
   - $x_i = f_i(x_1, x_2, \ldots, x_n)$
   - nodes $\rightarrow$ variables of the equation system
   - edges $\rightarrow$ contributions to variables
4. Use a generic solver on the equation system.

Example



Example



$$x_1 = 0 \sqcup (x_2 + [1, 1])$$
$$x_2 = x_1 \sqcap [-\infty, 99]$$
$$x_3 = x_1 \sqcap [100, \infty]$$

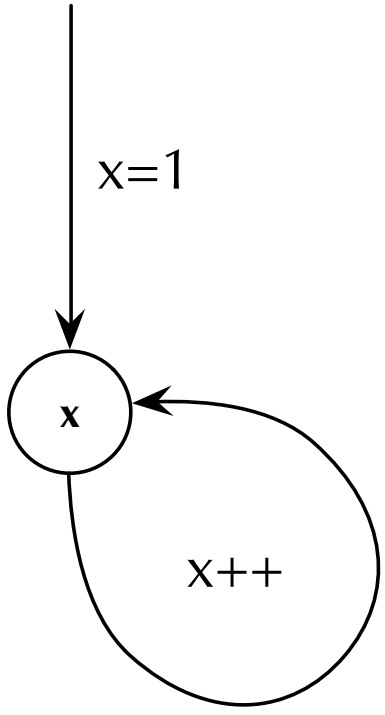
$\sqcup$ — interval union

$\sqcap$ — interval intersection

Infinite ascending chains ...

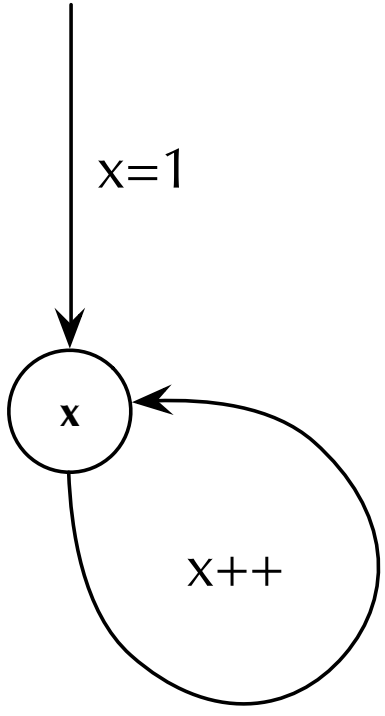
$$x = x + 1$$

$$[1,1] \sqsubseteq [1,2] \sqsubseteq [1,3] \sqsubseteq [1,4] \sqsubseteq \ldots$$



Infinite ascending chains ...

$$x = x + 1$$



$$[1,1] \sqsubseteq [1,2] \sqsubseteq [1,3] \sqsubseteq [1,4] \sqsubseteq \ldots$$

Fixpoint iteration may not terminate!

The Classical Approach

To speed up solving and enforce termination, Cousot and Cousot proposed the Widening and Narrowing approach.

The Classical Approach

To speed up solving and enforce termination, Cousot and Cousot proposed the Widening and Narrowing approach.

- Widening (using operator $\sqcup$)
  - Accelerate fixpoint computation by throwing away information.
  - e.g., $[1,1] \sqsubseteq [1,2] \sqsubseteq [1,3] \sqsubseteq \ldots$
    $\rightarrow [1,1] \sqcup [1,2] = [1,\infty]$

Widening example

$$x_1 = x_1 \sqcup (0 \sqcup (x_2 + 1))$$
$$x_2 = x_2 \sqcup (x_1 \sqcap [-\infty, 99])$$
$$x_3 = x_3 \sqcup (x_1 \sqcap [100, \infty])$$

produces

x_1	0
x_2	
x_3	

Widening example

$$x_1 = x_1 \sqcup (0 \sqcup (x_2 + 1))$$
$$x_2 = x_2 \sqcup (x_1 \sqcap [-\infty, 99])$$
$$x_3 = x_3 \sqcup (x_1 \sqcap [100, \infty])$$

produces

x_1	0	0
x_2		0
x_3		

Widening example

$$x_1 = x_1 \uplus (0 \sqcup (x_2 + 1))$$
$$x_2 = x_2 \uplus (x_1 \sqcap [-\infty, 99])$$
$$x_3 = x_3 \uplus (x_1 \sqcap [100, \infty])$$

produces

x_1	0	0	$[0, \infty]$
x_2		0	$[0, \infty]$
x_3			

Widening example

$$x_1 = x_1 \sqcup (0 \sqcup (x_2 + 1))$$
$$x_2 = x_2 \sqcup (x_1 \sqcap [-\infty, 99])$$
$$x_3 = x_3 \sqcup (x_1 \sqcap [100, \infty])$$

produces

x_1	0	0	$[0, \infty]$	$[0, \infty]$
x_2		0	$[0, \infty]$	$[0, \infty]$
x_3				$[100, \infty]$

The Classical Approach (cont.)

- Narrowing (using operator $\boxminus$)
  - Subsequently recover some information by decreasing iteration.
  - e.g., $[1, \infty] \sqsupseteq [1, 99] \sqsupseteq [1, 98] \sqsupseteq \ldots$

    $\rightarrow [1, \infty] \boxminus [1, 100] = [1, 100]$

    $[1, 100] \boxminus [1, 99] = [1, 100]$
  - Requires monotonic right-hand sides.

Narrowing example

$$
\begin{aligned}
x_1 &= x_1 \sqcap (0 \sqcup (x_2 + 1)) \\
x_2 &= x_2 \sqcap (x_1 \sqcap [-\infty, 99]) \\
x_3 &= x_3 \sqcap (x_1 \sqcap [100, \infty])
\end{aligned}
$$

produces

x_1	$[0, \infty]$
x_2	$[0, \infty]$
x_3	$[100, \infty]$

Narrowing example

$$x_1 = x_1 \sqcap (0 \sqcup (x_2 + 1))$$
$$x_2 = x_2 \sqcap (x_1 \sqcap [-\infty, 99])$$
$$x_3 = x_3 \sqcap (x_1 \sqcap [100, \infty])$$

produces

x_1	$[0, \infty]$	$[0, \infty]$
x_2	$[0, \infty]$	$[0, 99]$
x_3	$[100, \infty]$	$[100, \infty]$

Narrowing example

$$x_1 = x_1 \sqcap (0 \sqcup (x_2 + 1))$$
$$x_2 = x_2 \sqcap (x_1 \sqcap [-\infty, 99])$$
$$x_3 = x_3 \sqcap (x_1 \sqcap [100, \infty])$$

produces

x_1	$[0, \infty]$	$[0, \infty]$	$[0, 100]$
x_2	$[0, \infty]$	$[0, 99]$	$[0, 99]$
x_3	$[100, \infty]$	$[100, \infty]$	$[100, \infty]$

Narrowing example

$$x_1 = x_1 \sqcap (0 \sqcup (x_2 + 1))$$
$$x_2 = x_2 \sqcap (x_1 \sqcap [-\infty, 99])$$
$$x_3 = x_3 \sqcap (x_1 \sqcap [100, \infty])$$

produces

x_1	$[0, \infty]$	$[0, \infty]$	$[0, 100]$	$[0, 100]$
x_2	$[0, \infty]$	$[0, 99]$	$[0, 99]$	$[0, 99]$
x_3	$[100, \infty]$	$[100, \infty]$	$[100, \infty]$	100

Cousot and Cousot

Solved:

- returns a (post-)solution
- guarantees termination
- trade-off precision for speed

Cousot and Cousot

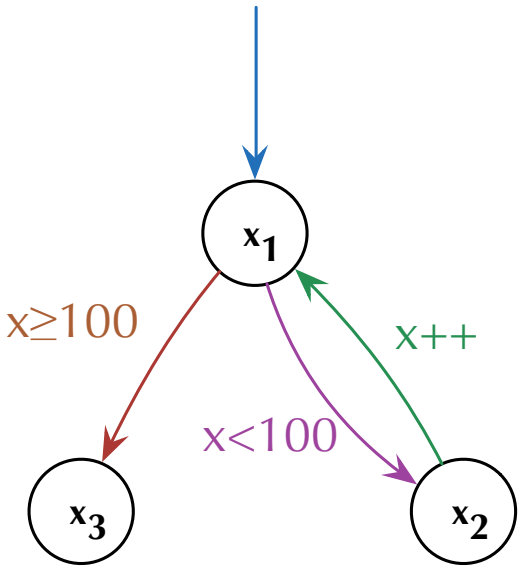
Solved:

- returns a (post-)solution
- guarantees termination
- trade-off precision for speed

Not Solved:

- precision loss due to separation into phases
- applicability for *local solving*

Context-sensitive example

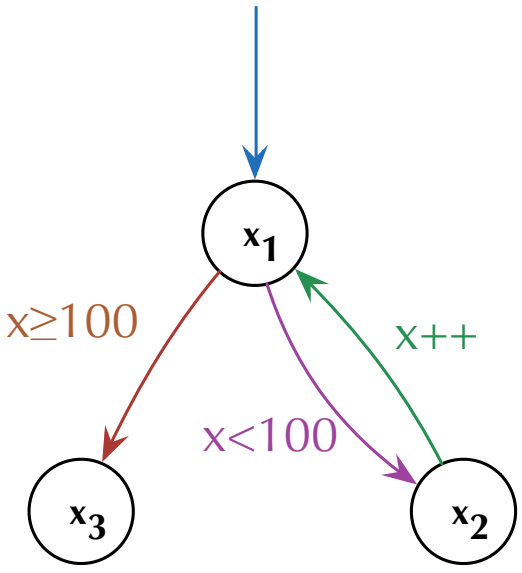


for all intervals c:

$$(x_1, c) = c \sqcup ((x_3, c) + [1, 1])$$
$$(x_2, c) = (x_1, c) \sqcap [100, \infty]$$
$$(x_3, c) = (x_1, c) \sqcap [-\infty, 99]$$

start by querying $(x_3, 0)$

Context-sensitive example



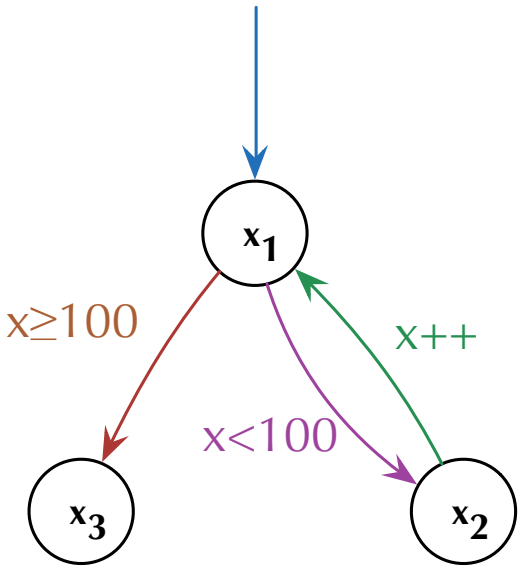
for all intervals c:

$$(x_1, c) = c \sqcup ((x_3, c) + [1, 1])$$
$$(x_2, c) = (x_1, c) \sqcap [100, \infty]$$
$$(x_3, c) = (x_1, c) \sqcap [-\infty, 99]$$

start by querying $(x_3, 0)$

- We may find *local* solutions.
  - Equation variables are solved on-demand.
  - Not well suited for 2-phased solving.

Context-sensitive example



for all intervals c:

$$(x_1, c) = c \sqcup ((x_3, c) + [1, 1])$$
$$(x_2, c) = (x_1, c) \sqcap [100, \infty]$$
$$(x_3, c) = (x_1, c) \sqcap [-\infty, 99]$$

start by querying $(x_3, 0)$

- We may find *local* solutions.
  - Equation variables are solved on-demand.
  - Not well suited for 2-phased solving.
- Right-hand sides not necessarily monotonic.
  - Narrowing not defined for non-decreasing iteration.

Our idea

Use given $\sqcup$ and $\sqcap$ operators and define

$$x \boxminus y = \begin{cases} x \sqcap y & \text{if } y \sqsubseteq x \\ x \sqcup y & \text{otherwise} \end{cases}$$

Our idea

Use given $\sqcup$ and $\sqcap$ operators and define

$$x \boxminus y = \begin{cases} x \sqcap y & \text{if } y \sqsubseteq x \\ x \sqcup y & \text{otherwise} \end{cases}$$

New equality system to solve:

$$x_i = x_i \boxminus f_i(x_1, x_2, \ldots, x_n)$$

Our idea

- Everything in one iteration.
- Monotonicity of r.h.s not an issue for the definition.
- Solution of

$$x_i = x_i \boxminus f_i(x_1, x_2, \ldots, x_n)$$

give a post-solution for

$$x_i = f_i(x_1, x_2, \ldots, x_n)$$

Solving

Good news: Generic "Chaotic fixpoint iteration" solvers return post-solutions whenever they terminate.

Solving

Good news: Generic "Chaotic fixpoint iteration" solvers return post-solutions whenever they terminate.

Bad news: Might not terminate even for trivial monotonic systems.

Solving

Good news: Generic "Chaotic fixpoint iteration" solvers return post-solutions whenever they terminate.

Bad news: Might not terminate even for trivial monotonic systems. E.g., Round-Robin iteration with $\sqsupseteq$ for

$$\begin{aligned} x_1 &= x_2 \\ x_2 &= x_3 + 1 \\ x_3 &= x_1 \end{aligned}$$

using the natural numbers, will not terminate

Example 2

$$x_1 = x_1 \boxminus x_2$$
$$x_2 = x_2 \boxminus (x_3 + 1)$$
$$x_3 = x_3 \boxminus x_1$$

produces

x_1	0
x_2	0
x_3	0

Example 2

$$x_1 = x_1 \boxminus x_2$$
$$x_2 = x_2 \boxminus (x_3 + 1)$$
$$x_3 = x_3 \boxminus x_1$$

produces

x_1	0	0
x_2	0	∞
x_3	0	0

Example 2

$$x_1 = x_1 \boxminus x_2$$
$$x_2 = x_2 \boxminus (x_3 + 1)$$
$$x_3 = x_3 \boxminus x_1$$

produces

x_1	0	0	∞
x_2	0	∞	∞
x_3	0	0	0

Example 2

$$x_1 = x_1 \boxminus x_2$$
$$x_2 = x_2 \boxminus (x_3 + 1)$$
$$x_3 = x_3 \boxminus x_1$$

produces

x_1	0	0	∞	∞
x_2	0	∞	∞	1
x_3	0	0	0	0

Example 2

$$x_1 = x_1 \boxminus x_2$$
$$x_2 = x_2 \boxminus (x_3 + 1)$$
$$x_3 = x_3 \boxminus x_1$$

produces

x_1	0	0	∞	∞	∞
x_2	0	∞	∞	1	1
x_3	0	0	0	0	∞

Example 2

$$x_1 = x_1 \boxminus x_2$$
$$x_2 = x_2 \boxminus (x_3 + 1)$$
$$x_3 = x_3 \boxminus x_1$$

produces

x_1	0	0	∞	∞	∞	1
x_2	0	∞	∞	1	1	1
x_3	0	0	0	0	∞	∞

Example 2

$$x_1 = x_1 \boxminus x_2$$
$$x_2 = x_2 \boxminus (x_3 + 1)$$
$$x_3 = x_3 \boxminus x_1$$

produces

x_1	0	0	∞	∞	∞	1	1
x_2	0	∞	∞	1	1	1	∞
x_3	0	0	0	0	∞	∞	∞

Example 2

$$x_1 = x_1 \boxminus x_2$$
$$x_2 = x_2 \boxminus (x_3 + 1)$$
$$x_3 = x_3 \boxminus x_1$$

produces

x_1	0	0	∞	∞	∞	1	1	1
x_2	0	∞	∞	1	1	1	∞	∞
x_3	0	0	0	0	∞	∞	∞	1

Our contribution

Adapted generic solvers.

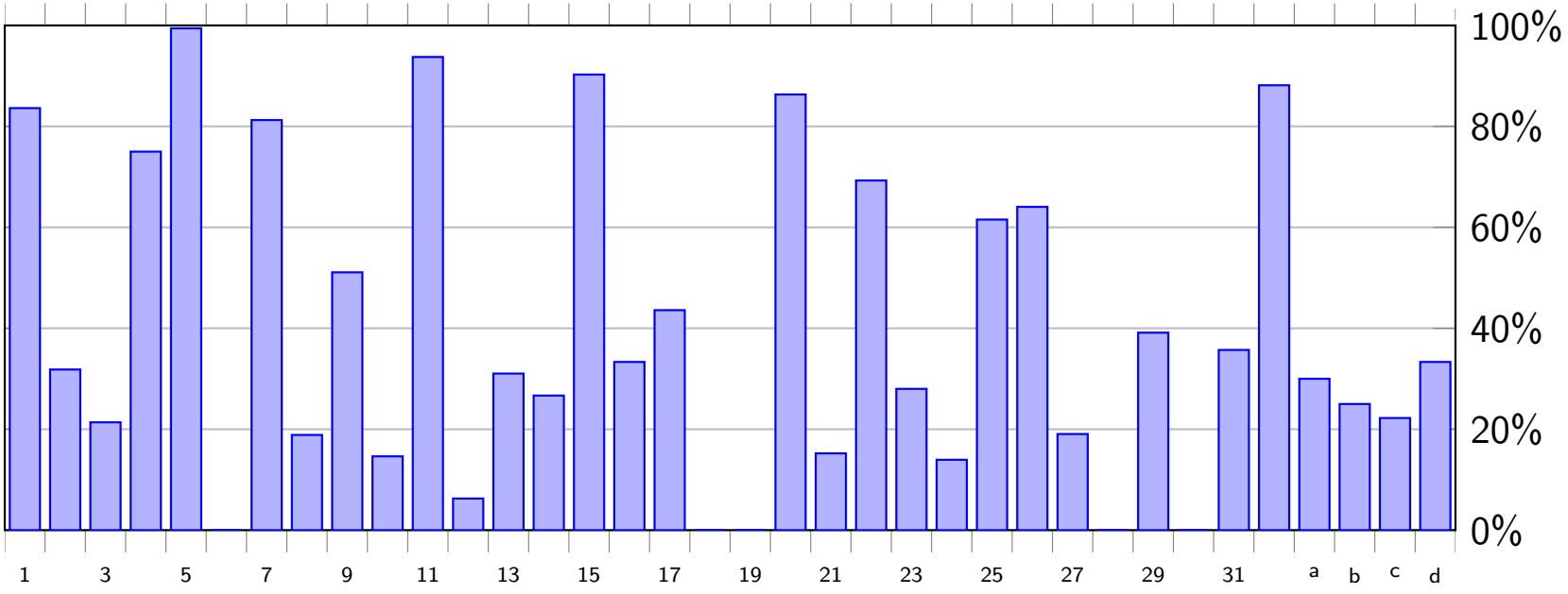
- Round-Robin Solver
- Worklist Solver
- Recursive Local Solver
- Side-effecting Local Solver

Which are correct and terminate for monotonic systems.

Key idea

- Introduce priorities on variables.
- Iterate till stabilization for lower priority variables first.

Relative improvement over two-phased solving



- % of program points where precision was gained
- The Mälardalen WCET (+ four hand-picked) benchmarks

Conclusion

- Possibility to use W/N in a local solving framework.
- Add robustness against non-monotonicity.
- Potentially increase in precision.