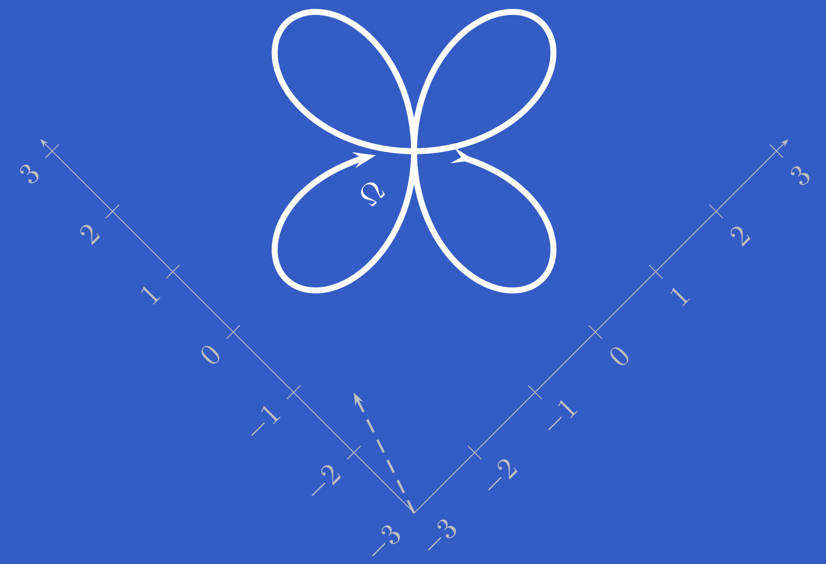
Universally Quantified Interval Constraint Solving

Frédéric Benhamou and Frédéric Goualard

Institut de Recherche en Informatique de Nantes, France

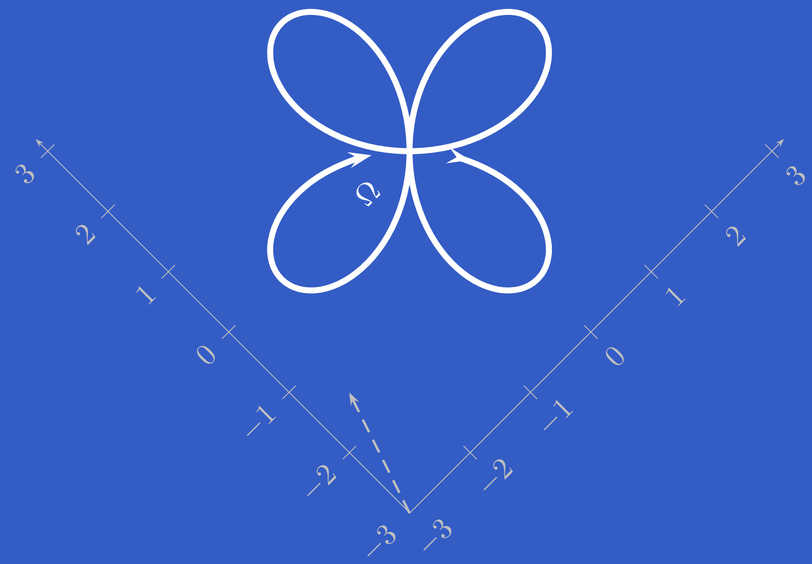
Motivations

Object Ω with trajectory: $\begin{cases} x(\theta) = 3\sin\theta\cos\theta(\sin\theta - \cos\theta) \\ y(\theta) = 3\sin\theta\cos\theta(\sin\theta + \cos\theta) \\ z(\theta) = 0 \end{cases}$



Motivations

Object Ω with trajectory: $\begin{cases} x(\theta) = 3\sin\theta\cos\theta(\sin\theta - \cos\theta) \\ y(\theta) = 3\sin\theta\cos\theta(\sin\theta + \cos\theta) \\ z(\theta) = 0 \end{cases}$

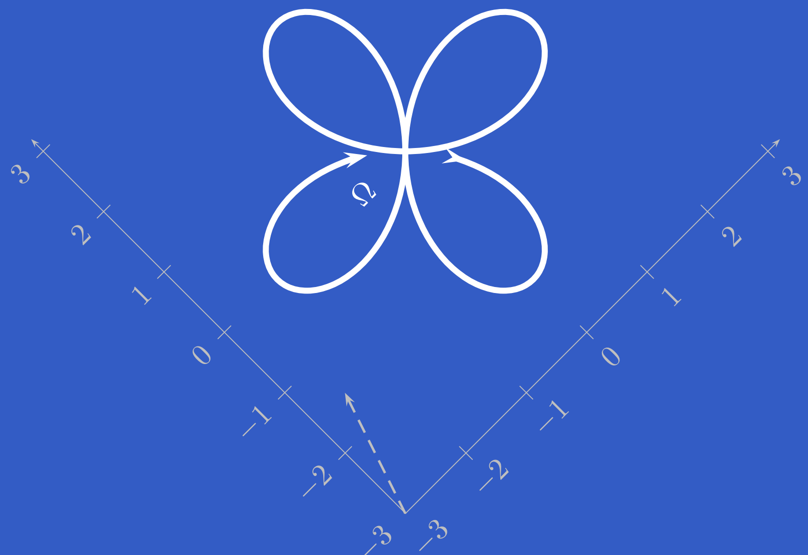


Positions of a camera Γ s.t. $|\Gamma, \Omega| \geqslant 0.5$ at any time?

Motivations

Object Ω with trajectory: $\begin{cases} x(\theta) = 3\sin\theta\cos\theta(\sin\theta - \cos\theta) \\ y(\theta) = 3\sin\theta\cos\theta(\sin\theta + \cos\theta) \\ z(\theta) = 0 \end{cases}$

Positions of a camera Γ s.t.
$|\Gamma, \Omega| \geqslant 0.5$ at any time?



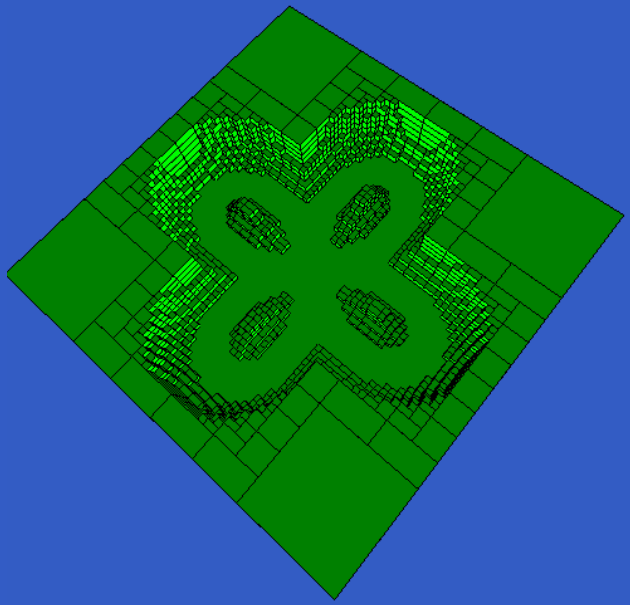
$\forall \theta \in [-\pi \,..\, \pi]:$

$$\sqrt{(3\sin\theta\cos\theta(\sin\theta - \cos\theta) - x)^2 + (3\sin\theta\cos\theta(\sin\theta + \cos\theta) - y)^2 + z^2} \geqslant 0.5$$

Motivations

Object Ω with trajectory: $\begin{cases} x(\theta) = 3\sin\theta\cos\theta(\sin\theta - \cos\theta) \\ y(\theta) = 3\sin\theta\cos\theta(\sin\theta + \cos\theta) \\ z(\theta) = 0 \end{cases}$

Positions of a camera Γ s.t.
$|\Gamma, \Omega| \geqslant 0.5$ at any time?



$\forall\theta \in [-\pi \mathinner{..} \pi]:$

$$\sqrt{(3\sin\theta\cos\theta(\sin\theta - \cos\theta) - x)^2 + (3\sin\theta\cos\theta(\sin\theta + \cos\theta) - y)^2 + z^2} \geqslant 0.5$$

Non-linear real constraint: compulsory to solve it *soundly*

Outline

- Sound techniques
  - Cylindrical Algebraic Decomposition
  - Interval arithmetic and inner-approximation computation
- Complete techniques
  - Interval constraint solving
- Interval constraint solving and soundness
- Solving constraints with universal quantifiers
- Conclusion and perspectives

Cylindrical Algebraic Decomposition

[Collins, 1973] (quantifier elimination)

- Powerful method (handling of universal/existential quantifiers)
- Sound solving of real constraints

Cylindrical Algebraic Decomposition

[Collins, 1973] (quantifier elimination)

- Powerful method (handling of universal/existential quantifiers)
- Sound solving of real constraints

But:

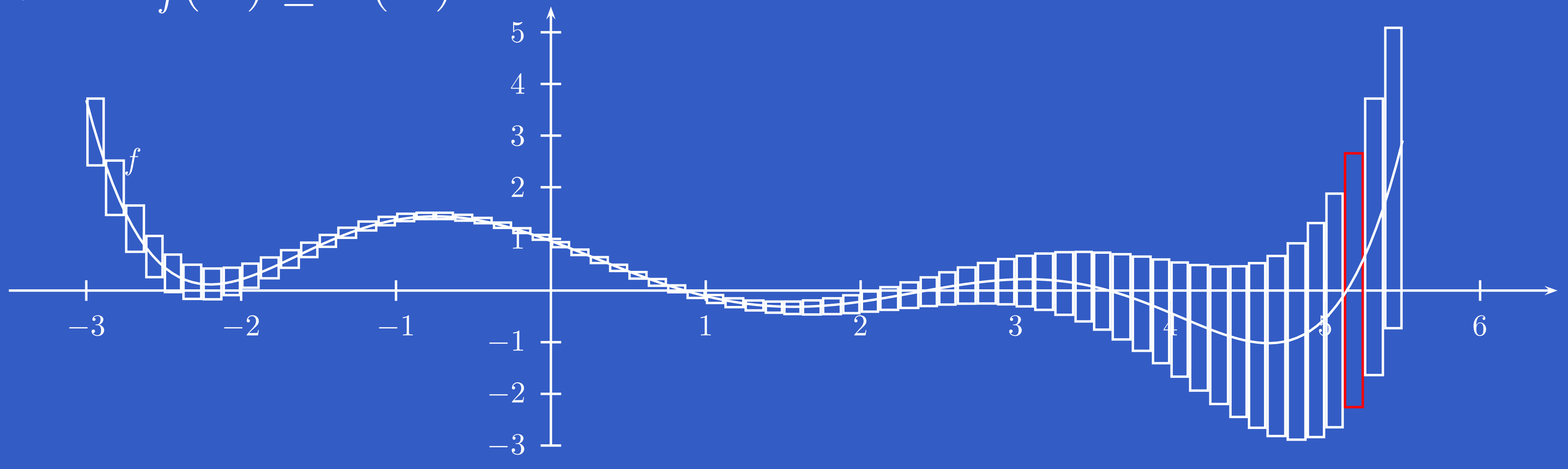
- Method restricted to polynomial constraints
- High complexity (time consuming)

Interval Arithmetic

[Moore, 1966] Interval set $\mathbb{I}$: $[a\,..\,b] = \{x \in \mathbb{R} \mid a \leqslant x \leqslant b\}$

Interval extension:

- *Extension of a real function.* $f\colon \mathbb{R}^n \longrightarrow \mathbb{R}$, $F\colon \mathbb{I}^n \longrightarrow \mathbb{I}$ s.t. $\forall \boldsymbol{B}\colon \mathcal{D}_f(\boldsymbol{B}) \subseteq F(\boldsymbol{B})$



- *Extension of a real relation* $\rho \subseteq \mathbb{R}^n$. Set of boxes $\mathcal{R}$ such that:

$$(a_1, \ldots, a_n) \in \rho \Rightarrow \exists (I_1, \ldots, I_n) \in \mathcal{R} \text{ s.t. } a_1 \in I_1, \ldots, a_n \in I_n$$

Approximation of a relation

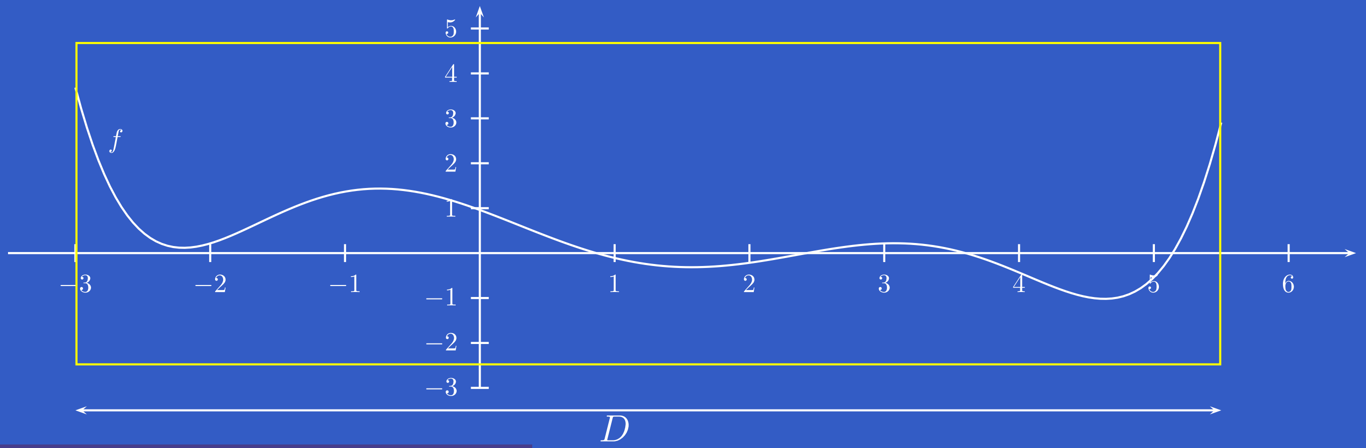
Outer approximation of ρ: $\text{Hull}(\rho) = \bigcap\{\boldsymbol{B} \in \mathbb{I}^n \mid \rho \subseteq \boldsymbol{B}\}$

Inner approximation of ρ:

$$\text{Inner}(\rho) = \{a \in \mathbb{R}^n \mid \text{Hull}(\{a\}) \subseteq \rho\}$$

Splitting/evaluation scheme (SES) for straightforward inner approximation computation:

Example. $c\colon f(x) \leqslant 0$?



Approximation of a relation

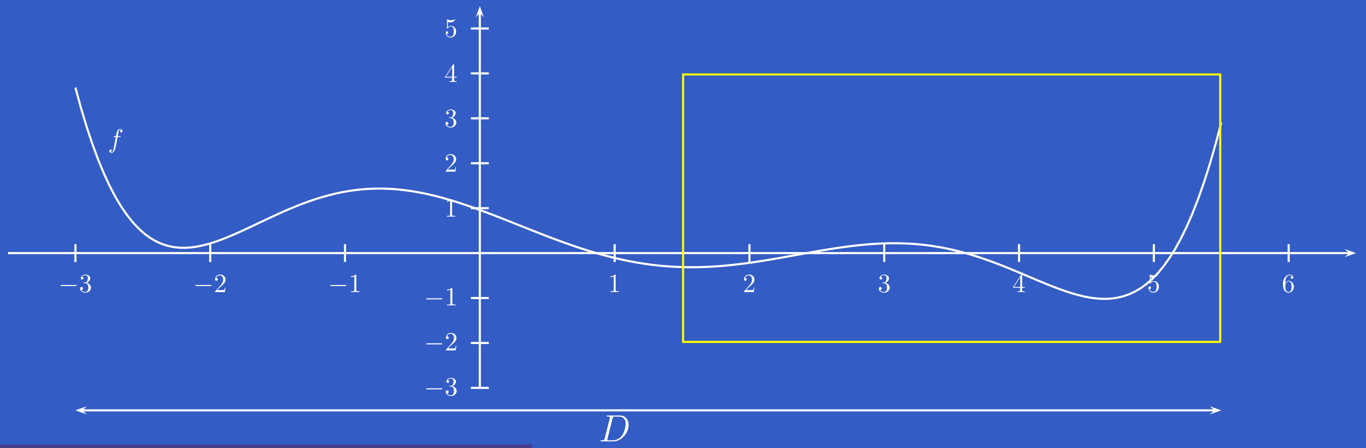
Outer approximation of ρ: $\mathsf{Hull}(\rho) = \bigcap \{\boldsymbol{B} \in \mathbb{I}^n \mid \rho \subseteq \boldsymbol{B}\}$

Inner approximation of ρ:

$$\mathsf{Inner}(\rho) = \{a \in \mathbb{R}^n \mid \mathsf{Hull}(\{a\}) \subseteq \rho\}$$

Splitting/evaluation scheme (SES) for straightforward inner approximation computation:

Example. $c\colon\ f(x) \leqslant 0$?



Approximation of a relation

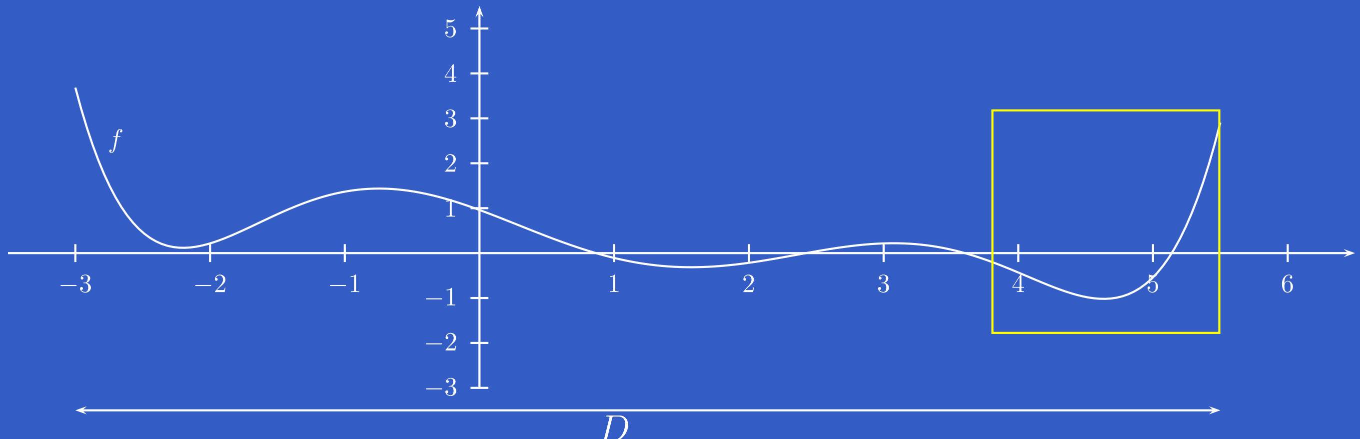
Outer approximation of ρ: $\mathsf{Hull}(\rho) = \bigcap\{\boldsymbol{B} \in \mathbb{I}^n \mid \rho \subseteq \boldsymbol{B}\}$

Inner approximation of ρ:

$$\mathsf{Inner}(\rho) = \{a \in \mathbb{R}^n \mid \mathsf{Hull}(\{a\}) \subseteq \rho\}$$

Splitting/evaluation scheme (SES) for straightforward inner approximation computation:

Example. $c\colon\ f(x) \leqslant 0$?



Approximation of a relation

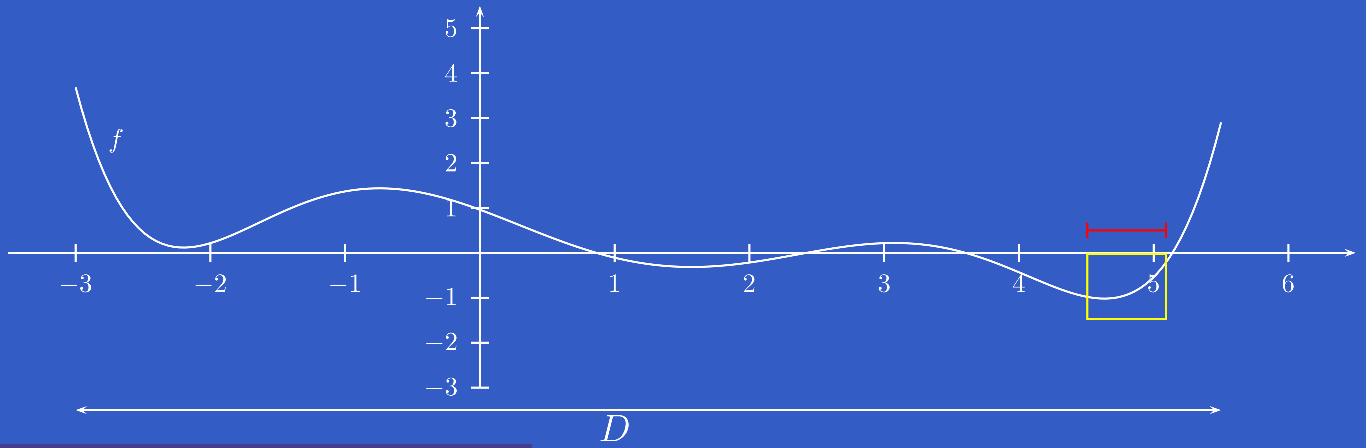
Outer approximation of ρ: $\text{Hull}(\rho) = \bigcap\{\boldsymbol{B} \in \mathbb{I}^n \mid \rho \subseteq \boldsymbol{B}\}$

Inner approximation of ρ:

$$\text{Inner}(\rho) = \{a \in \mathbb{R}^n \mid \text{Hull}(\{a\}) \subseteq \rho\}$$

Splitting/evaluation scheme (SES) for straightforward inner approximation computation:

Example. $c\colon f(x) \leqslant 0$?



Approximation of a relation

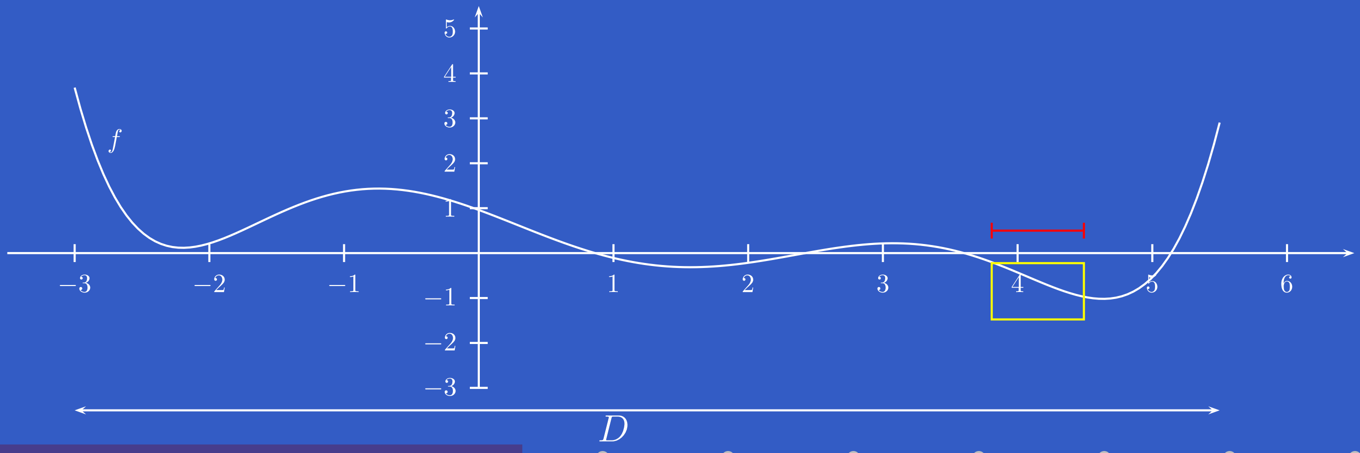
Outer approximation of ρ: $\text{Hull}(\rho) = \bigcap\{\boldsymbol{B} \in \mathbb{I}^n \mid \rho \subseteq \boldsymbol{B}\}$

Inner approximation of ρ:

$$\text{Inner}(\rho) = \{a \in \mathbb{R}^n \mid \text{Hull}(\{a\}) \subseteq \rho\}$$

Splitting/evaluation scheme (SES) for straightforward inner approximation computation:

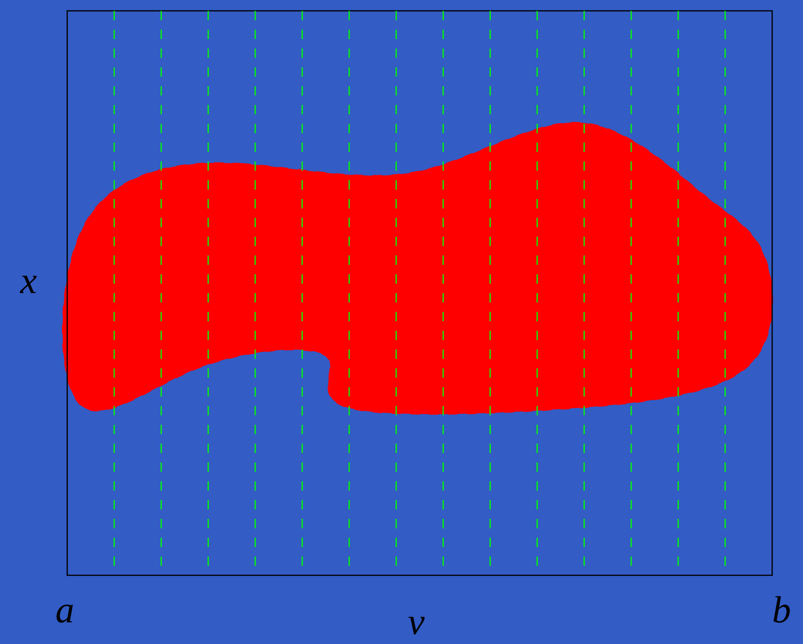
Example. $c\colon f(x) \leqslant 0$?



EIA4

[Jardillier & Languénou, 1998] (Modelling of camera movements)

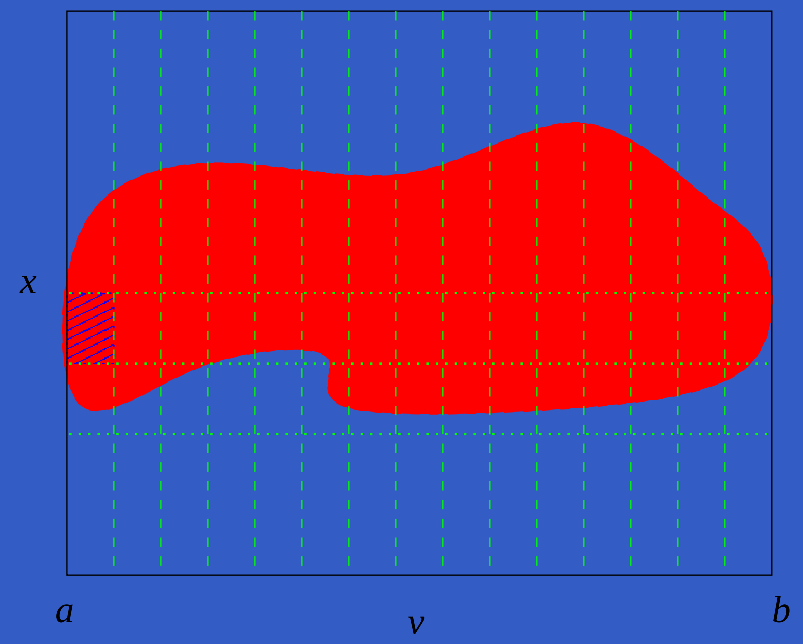
Solving $\forall v \in [a\,..\,b] \colon c(x, v)$ with the SES:



EIA4

[Jardillier & Languénou, 1998] (Modelling of camera movements)

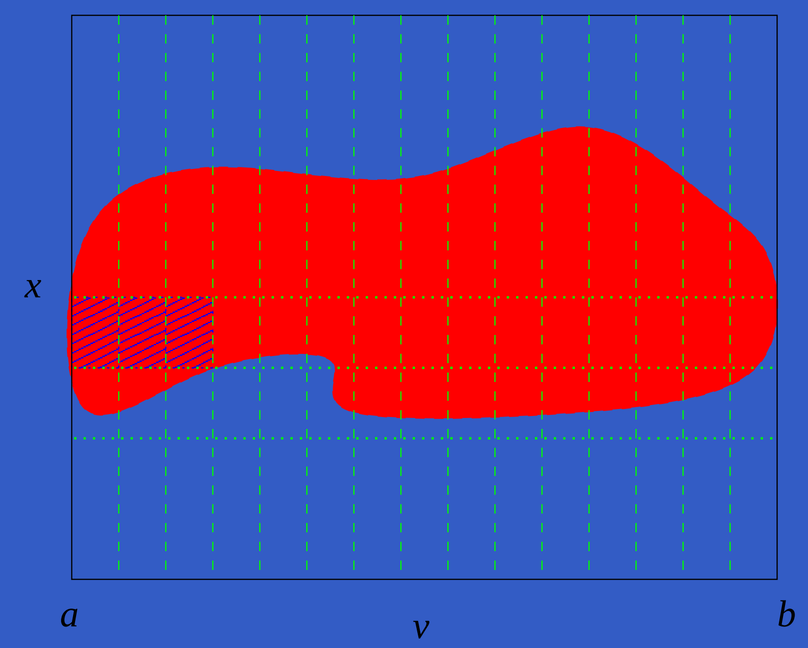
Solving $\forall v \in [a \mathinner{..} b]\colon c(x, v)$ with the SES:



EIA4

[Jardillier & Languénou, 1998] (Modelling of camera movements)

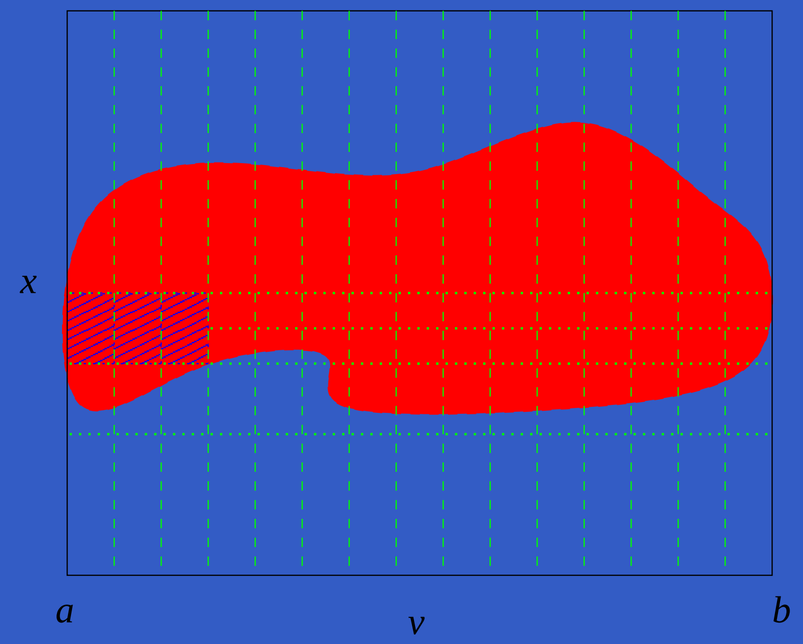
Solving $\forall v \in [a \mathinner{..} b] \colon c(x, v)$ with the SES:



EIA4

[Jardillier & Languénou, 1998] (Modelling of camera movements)

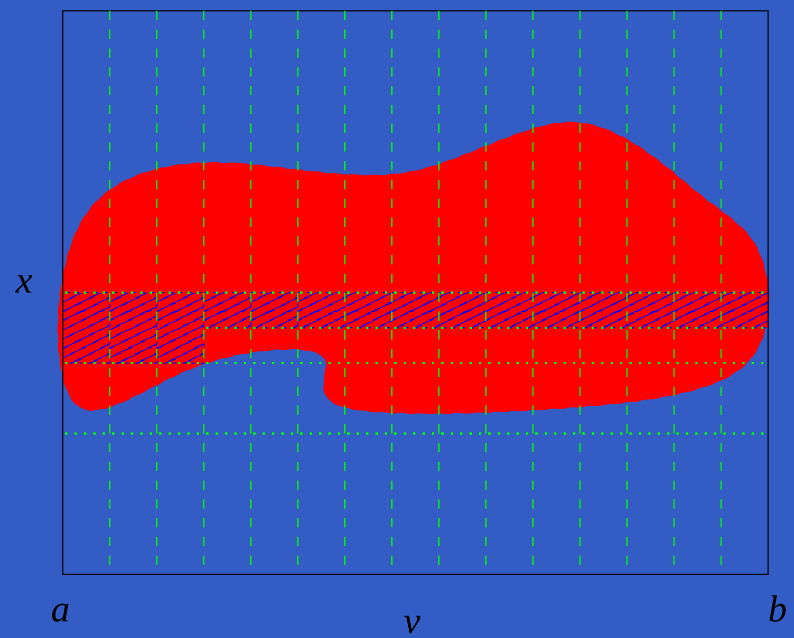
Solving $\forall v \in [a\,..\,b] \colon c(x, v)$ with the SES:



EIA4

[Jardillier & Languénou, 1998] (Modelling of camera movements)

Solving $\forall v \in [a \,..\, b] \colon c(x, v)$ with the SES:



$\Rightarrow$ algorithm EIA4 (very costly)

Interval constraint solving

- Variables $x_1, \ldots, x_n \rightarrow$ domains $D_1, \ldots, D_n$
- Constraint $c(x_1, \ldots, x_n)$

  $\rightarrow$ *contracting operator* $N[c]$

  [BENHAMOU & OLDER, APT, . . .]

$\boldsymbol{D} = D_1 \times \cdots \times D_n$ ⟶ N[c] ⟶ $\boldsymbol{D'} \subseteq \boldsymbol{D}$

- Properties: contractance, monotony, *completeness* ($\rho_c \cap \boldsymbol{D} \subseteq N[c](\boldsymbol{D})$)
- Constraint system $c_1, \ldots, c_m$: domain propagation algorithm (AC3 [MACKWORTH,1977])
- Discarding values: *local consistency* $+$ splitting

Box consistency

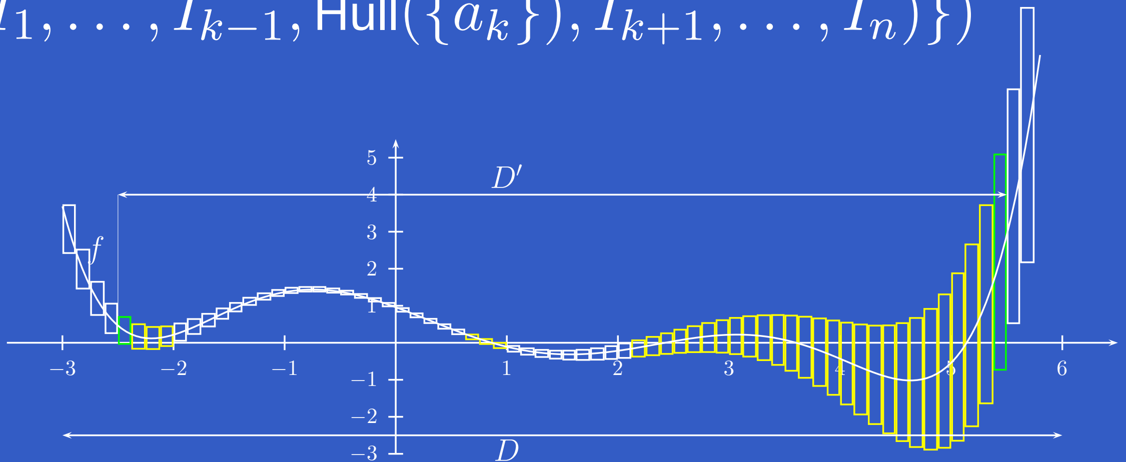
[Benhamou, McAllester, Van Hentenryck, 1994]

Given c a real constraint, C an interval extension for c,

$\boldsymbol{B} = I_1 \times \cdots \times I_n$ a box

c is *box-consistent w.r.t.* $\boldsymbol{B}$ iff $\forall k \in \{1, \ldots, n\}$:

$$I_k = \mathsf{Hull}(I_k \cap \{a_k \in \mathbb{R} \mid C(I_1, \ldots, I_{k-1}, \mathsf{Hull}(\{a_k\}), I_{k+1}, \ldots, I_n)\})$$



Box consistency operator:

$$\boldsymbol{B} \cap \rho_c \subseteq \mathsf{OCb}_c(\boldsymbol{B}) = \max\{\boldsymbol{B'} \mid \boldsymbol{B'} \subseteq \boldsymbol{B} \wedge c \text{ is box-consistent w.r.t. } \boldsymbol{B'}\}$$

Interval constraint solving

- Efficiency (systems of non-linear real equations with punctual solutions)
- No restriction to polynomials
- Completeness: no solution lost

Interval constraint solving

- Efficiency (systems of non-linear real equations with punctual solutions)
- No restriction to polynomials
- Completeness: no solution lost

But:

- Continuum of solutions? (systems of inequations)
- Soundness ?
- Quantified variables?

Achieving soundness

Computing with interval constraint prog. techniques on $\neg c$:

$\Rightarrow$ completeness for $\neg c$

$\Rightarrow$ *correctness* for c

For discarded box $\boldsymbol{B} = I_1 \times \cdots \times I_n \times I_v$:

$\neg(\exists x_1 \ldots \exists x_n \exists v \colon x_1 \in I_1 \ldots x_n \in I_n v \in I_v \wedge \neg c(x_1, \ldots, x_n, v))$

Achieving soundness

Computing with interval constraint prog. techniques on $\neg c$:

$\Rightarrow$ completeness for $\neg c$

$\Rightarrow$ *correctness* for c

For discarded box $B = I_1 \times \cdots \times I_n \times I_v$:

$\neg(\exists x_1 \ldots \exists x_n \exists v : x_1 \in I_1 \ldots x_n \in I_n v \in I_v \wedge \neg c(x_1, \ldots, x_n, v))$

That is:

$\forall x_1 \ldots \forall x_n \forall v : x_1 \in I_1 \ldots x_n \in I_n v \in I_v \Rightarrow c(x_1, \ldots, x_n, v)$

(computation of inner approximation for ρ_c)

Negation of c: easy for inequations

Handling quantifiers

Solving $\forall v \in I_v : c$

- Computing box consistency on $\neg c$
- Saving discarded boxes $B' = I'_1 \times \cdots \times I'_n \times I'_v$ with $I_v = I'_v$
- Splitting undiscarded boxes and iterating

Handling quantifiers

Solving $\forall v \in I_v : c$

- Computing box consistency on $\neg c$
- Saving discarded boxes $B' = I'_1 \times \cdots \times I'_n \times I'_v$ with $I_v = I'_v$
- Splitting undiscarded boxes and iterating

Solving $\forall v \in I^1 : c_1 \wedge \cdots \wedge \forall v \in I^m : c_m$

- Applying the previous scheme on $c_1, \ldots, c_n$ in sequence once
- Using the output after c_i as input for c_{i+1}

Solving $\forall v \in I_v \colon c$

$\text{ICAb3}_c(\textbf{in}\colon \boldsymbol{B} \in \mathbb{I}^n_\circ, v \in \mathcal{V}_\mathbb{R};\ \textbf{out}\colon \mathcal{W} \in \mathcal{P}(\mathbb{I}^n_\circ))$

begin

  $\boldsymbol{B'} \leftarrow \text{OCb}_c(\boldsymbol{B})$

  if $(\text{Dom}_{\boldsymbol{B'}}(v) = \text{Dom}_{\boldsymbol{B}}(v))$ **then**

    $\boldsymbol{D} \leftarrow \text{OCb}_{\overline{c}}(\boldsymbol{B'})$

    $\mathcal{W} \leftarrow \boldsymbol{B'} \setminus \boldsymbol{D}|_{v,\boldsymbol{B'}}$

    if $(\boldsymbol{D} \neq \varnothing$ **and** $\neg\text{Canonical}_v(\boldsymbol{D}))$ **then**

      $(\boldsymbol{D_1}, \boldsymbol{D_2}) \leftarrow \text{Split}_v(\boldsymbol{D}|_{v,\boldsymbol{B'}})$

      $\mathcal{W} \leftarrow \mathcal{W} \cup \text{ICAb3}_c(\boldsymbol{D_1}, v) \cup \text{ICAb3}_c(\boldsymbol{D_2}, v)$

    endif

    return $(\mathcal{W})$

  else

    return $(\varnothing)$

end

Constraint systems and $\forall$

Solving $\forall v \in I^1 \colon c_1 \wedge \cdots \wedge \forall v \in I^m \colon c_m$

ICAb5(**in**: $\mathcal{S} = \{(c_1, I^1), \ldots, (c_m, I^m)\}$,
    $\mathcal{A} \in \mathcal{P}(\mathbb{I}^n_\circ)$, $v \in \mathcal{V}_\mathbb{R}$; **out**: $\mathcal{W} \in \mathcal{P}(\mathbb{I}^n_\circ)$)
begin
    if $(\mathcal{S} \neq \varnothing)$ **then**
        $\mathcal{B} \leftarrow \varnothing$
        foreach $D \in \mathcal{A}$ **do**
            $\mathcal{B} \leftarrow \mathcal{B} \cup$ ICAb3$_{c_1}(\boldsymbol{D}|_{v,I^1}, v,)$
        endforeach
        if $(\mathcal{B} = \varnothing)$ **then**
            return $(\varnothing)$
        else
            return (ICAb5$(\mathcal{S} \setminus \{(c_1, I^1)\}, \mathcal{B}, v)$)
        endif
    else
        return $(\mathcal{A})$
    endif
end

Properties

Given $\tilde{\rho}_i$, relation associated with $\forall v \in I_v \colon c_i$

Soundness:

$$\text{ICAb3}_c(\boldsymbol{B}, v) \subseteq \text{Inner}(\boldsymbol{B} \cap \tilde{\rho}_c)$$

$$\text{ICAb5}(\mathcal{S}, \{\boldsymbol{B}\}, v) \subseteq \text{Inner}(\boldsymbol{B} \cap \tilde{\rho}_1 \cap \cdots \cap \tilde{\rho}_m)$$

Comparison with EIA4:

$$\text{EIA4}(\{c_1, \ldots, c_m\}, \boldsymbol{B}, v) \subseteq \text{ICAb5}(\mathcal{S}, \{\boldsymbol{B}\}, v)$$

ICAb5 vs. EIA4

All solutions:

Benchmark	EIA4	ICAb5	EIA4 / ICAb5
$\text{Projection}_{3,5}$	783	68	11
$\text{Projection}_{3,10}$	$>$9000	3634	$>$2
$\text{Projection}_{5,5}$	$>$9000	3612	$>$2
$\text{School Problem}_{3,1}$	156	13	12
$\text{Flying Saucer}_{4,1}$	1459	1078	1
$\text{Simple Circle}_{2,1}$	12789	651	19
$\text{Simple Circle}_{2,2}$	1579	56	28

Time in s. on a SUN UltraSparc 1/167 MHz.

N.B.: first solution only $\Rightarrow$ speed-up much more important

Conclusion and perspectives

Alg. ICAb5:

- solving of non-linear real constraints (polynomial or not) with one universally quantified variable per constraint
- Speed-up compared to splitting/evaluation (EIA4)
- Still costly

Solutions?

- More precise interval extensions (Bernstein)
- Other interval arithmetics (Kaucher/Markov, modal arithmetic)
- Cooperation of symbolic/numerical methods (CAD?)

Handling of constraints as full first-order expressions?

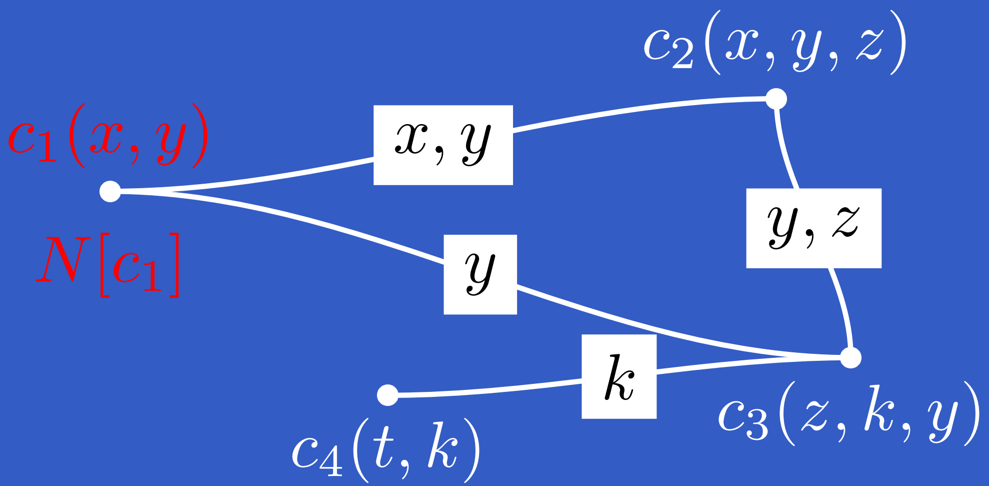
Universally Quantified Interval Constraint Solving

Frédéric Benhamou and Frédéric Goualard

Institut de Recherche en Informatique de Nantes, France

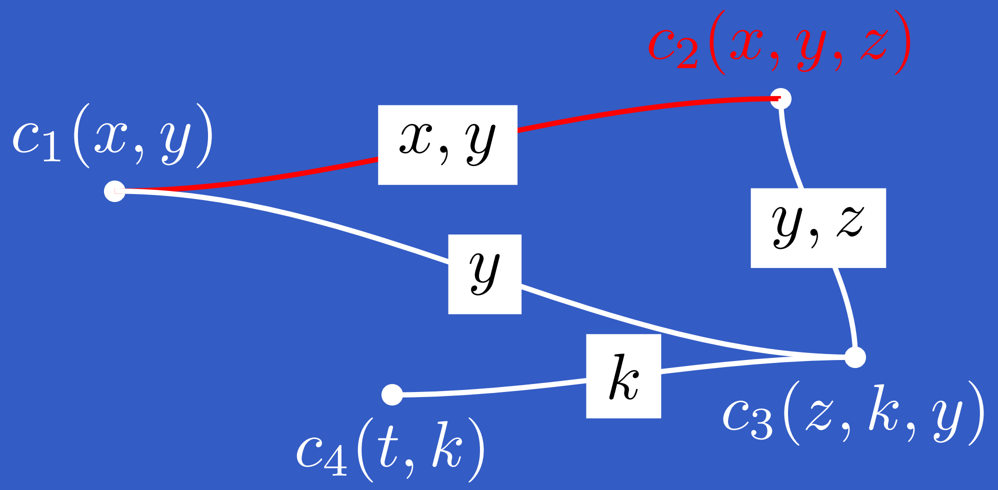
Interval constraint solving (2)

- Constraint system $c_1, \ldots, c_m$: domain propagation algorithm (AC3 [MACKWORTH,1977])



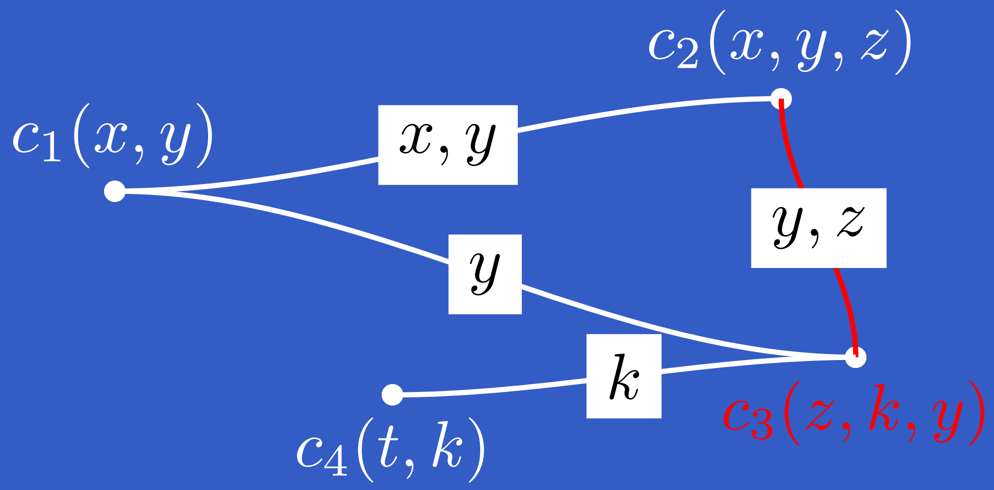
Interval constraint solving (2)

- Constraint system $c_1, \ldots, c_m$: domain propagation algorithm (AC3 [MACKWORTH,1977])



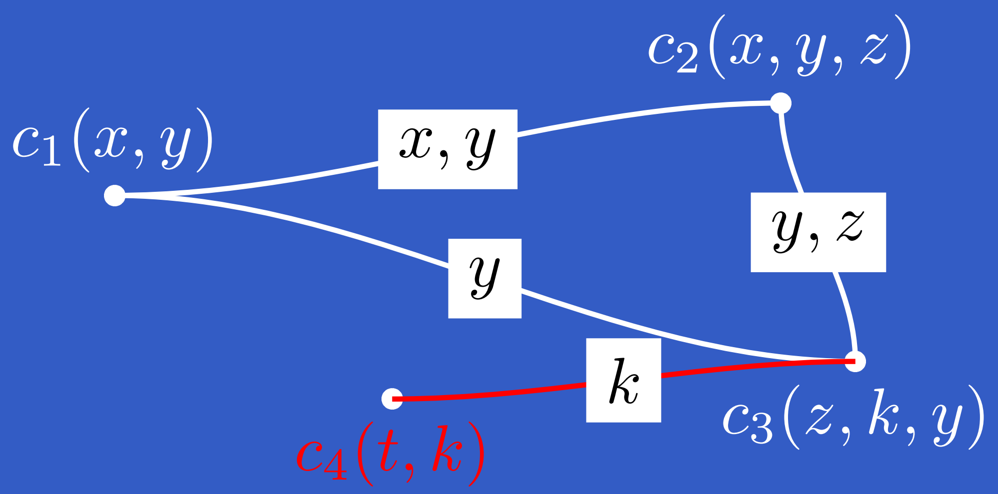
Interval constraint solving (2)

- Constraint system $c_1, \ldots, c_m$: domain propagation algorithm (AC3 [MACKWORTH,1977])



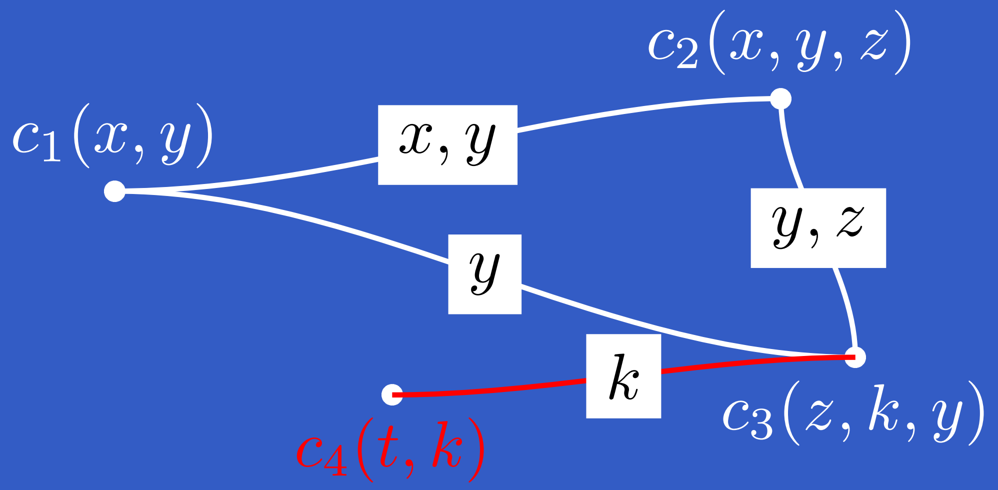
Interval constraint solving (2)

- Constraint system $c_1, \ldots, c_m$: domain propagation algorithm (AC3 [MACKWORTH,1977])



Interval constraint solving (2)

- Constraint system $c_1, \ldots, c_m$: domain propagation algorithm (AC3 [Mackworth,1977])



Properties:

- Termination
- Contractance
- Confluency
- Completeness