raSAT: SMT for Polynomial Inequality

To Van Khanh (UET/VNU-HN)
Vu Xuan Tung, Mizuhito Ogawa (JAIST)

2014.7.18

Polynomial constraints (QF_NRA)

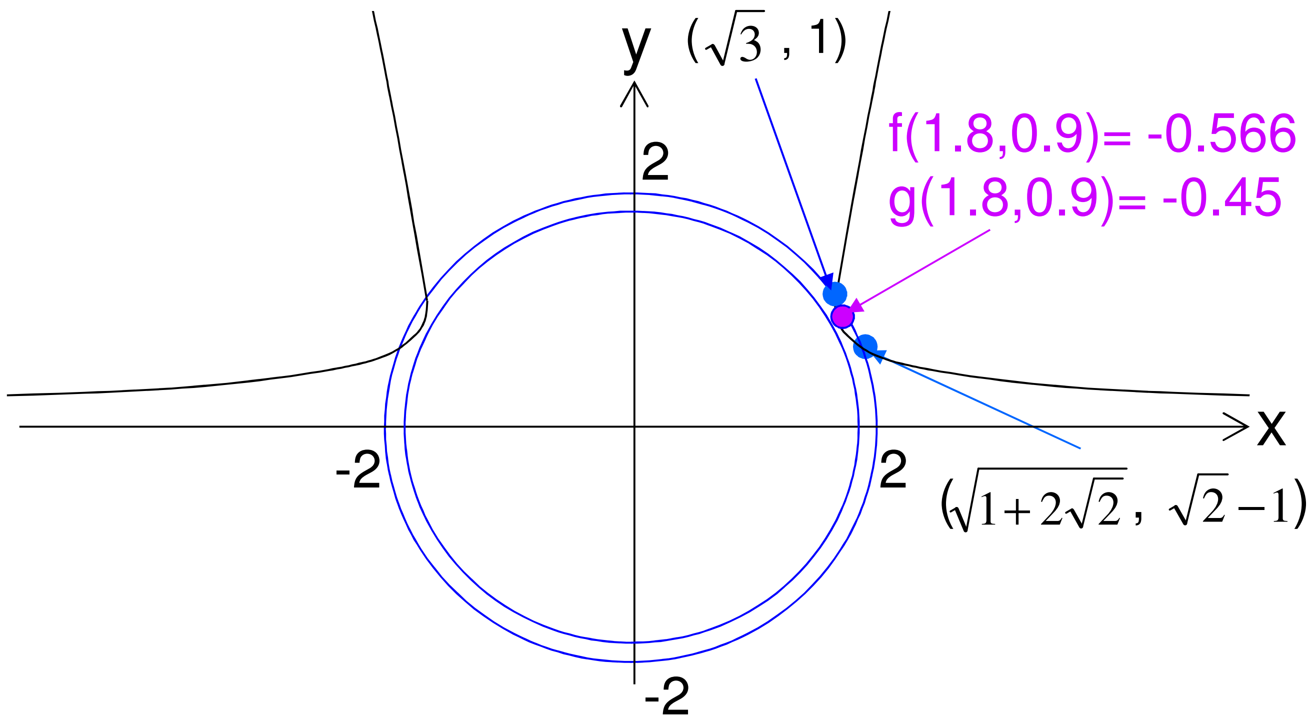
- Polynomial constraints (with integer coefficients) consist of
  - ✓Bounding inputs $x_i \in [l_i, h_i]$
  - ✓Polynomial equality/inequality $f_j > 0, f_i \geqq 0, f_i = 0$
  - ✓SAT if bounded quantification
    $\exists x_1 \in [l_1, h_1] \ldots x_n \in [l_n, h_n] . \wedge_j f_j \sim 0$ ($\sim = >, \geqq, =$)
    holds over real numbers; UNSAT otherwise.

- Motivated by
  - ✓Roundoff error analysis [Do Ogawa, 2009]

Polynomial constraints example

$\exists x\ y.\ f(x,y) < 0 \wedge g(x,y) < 0$?

where $\begin{cases} f(x,y) = y^2 - (x^2 - 1)y + 1 \\ g(x,y) = x^2 + y^2 - 4 \end{cases}$



raSAT for polynomial (strict) inequality

- Polynomial inequality (with bounded quantification)
  - ✓ $\exists x_1 \in (l_1, h_1) \ldots x_n \in (l_n, h_n) . \wedge_j f_j > 0$
- Strict inequality allows
  - ✓ approximation
  - ✓ open intervals only
  - ✓ SAT instances in rational numbers (if exists)
- raSAT web site (participated QF_NRA in SMTcomp)
  http://www.jaist.ac.jp/~mizuhito/tools/rasat.html
  - ✓ Current raSAT support ad-hoc equality
    (e.g., equality with integers)

By **raSAT** (previous example)

```
File  Edit  View  Search  Terminal  Help
tungvx@tungdeptrai ~/raSAT/development_ver/raSAT/solver $ ./raSAT sample.smt2 lb="0 10"
WARNING: for repeatability, setting FPU to use double precision

Start searching, please wait....

==========================[ Problem Statistic ]==================================

Input problem         : sample.smt2
Number of variables   : 2
Number of constraints : 2
Interval Arithmetic   : AF2
Unit searching box    : 0.1
Timeout setting       : 60 seconds

Total running time    : 0.008 seconds

IA time               : 0.004 seconds

Testing time          : 0 seconds

UNSAT Core time       : 0 seconds

Parsing time          : 0 seconds

Decomposition time    : 0 seconds

Ocaml time            : 0 seconds

MiniSAT time          : 0.004 seconds

MiniSAT vars          : 30

MiniSAT max clauses   : 46

MiniSAT calls         : 27

raSAT clauses         : 74

Decomposed clauses    : 56

UNSAT learned clauses : 18

UNKOWN learned clauses: 0
Result                : SAT

================================[ SAT instances ]================================

y = 0.687783209694
x = 1.875

=====================[ Detail SAT for each constraint ]==========================

y*y+y-x*x*y+1.=-0.25715889335 < 0.
y*y+x*x-4.=-0.0113292564623 < 0.
tungvx@tungdeptrai ~/raSAT/development_ver/raSAT/solver $
```

```
(set-logic QF_NRA)
(declare-fun x () Real)
(declare-fun y () Real)
(assert (< (+ (- (* y y) (* (- (* x x) 1.) y)) 1.) 0.))
(assert (< (- (+ (* x x) (* y y)) 4.) 0.))
(check-sat)
```

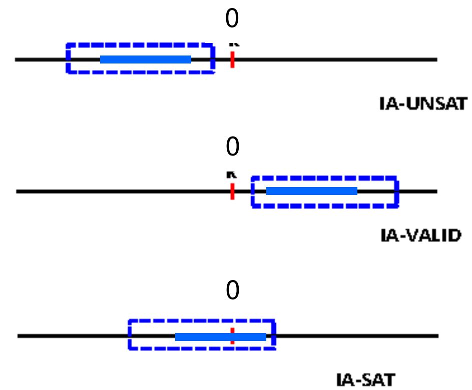
x=0.687783209694
y=1.875

Approximation methodology

- Over-approximation (O.T)

$x_1 \in I_1 \ldots x_n \in I_n . \wedge_j f_j > 0$ ⇨ **O.T** ⇨ *O.T-valid* (SAT), *O.T-UNSAT* (UNSAT), *O.T-SAT* (*unknown*)

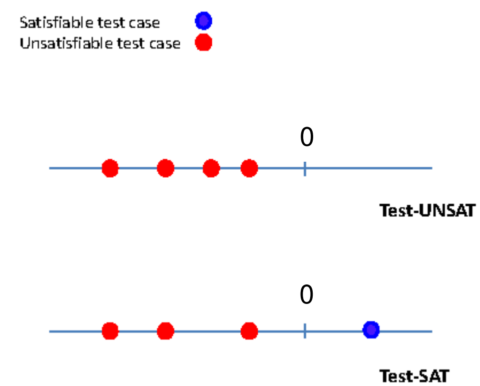
$f(x_1,\ldots,x_n) > 0$



✓Instance: Interval Arithmetic (IA)

- Under-approximation (U.T)

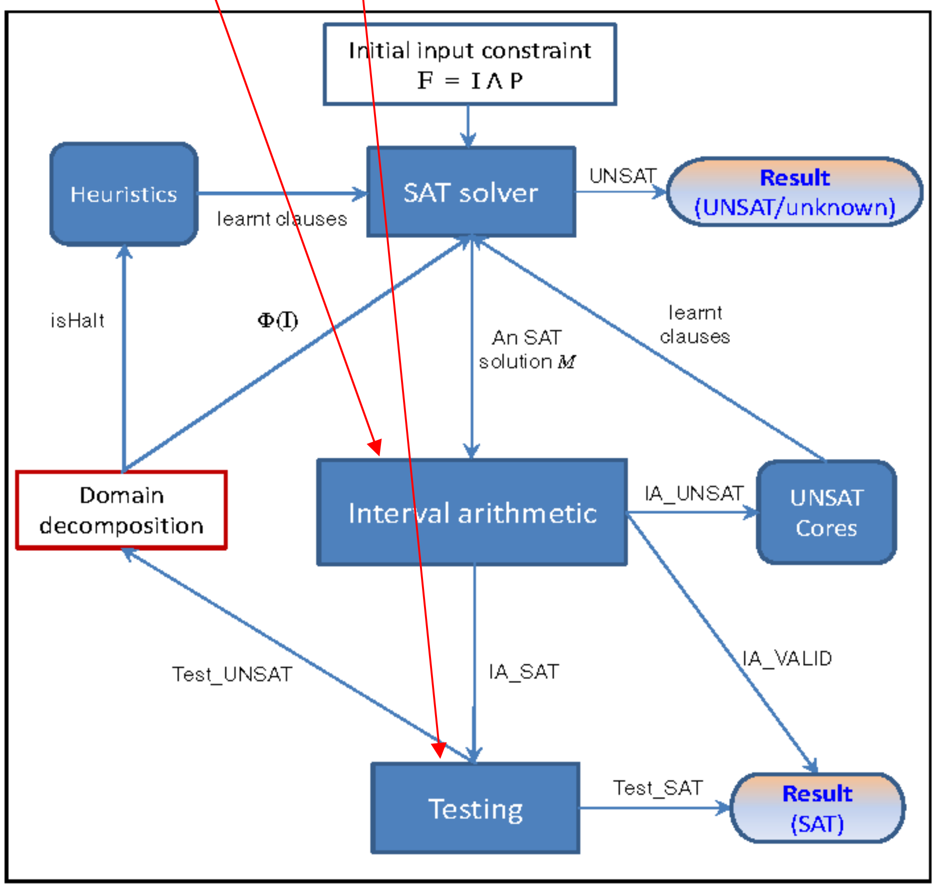
$x_1 \in I_1 \ldots x_n \in I_n . \wedge_j f_j > 0$ ⇨ **U.T** ⇨ *U.T-SAT* (SAT), *U.T-UNSAT* (*unknown*)



✓Instance: testing (to accelerate SAT)

raSAT loop

- Our idea : Instead of exact theory (QE-CAD), apply over/under approximations + **refinement**
- Refinement by box decomposition.

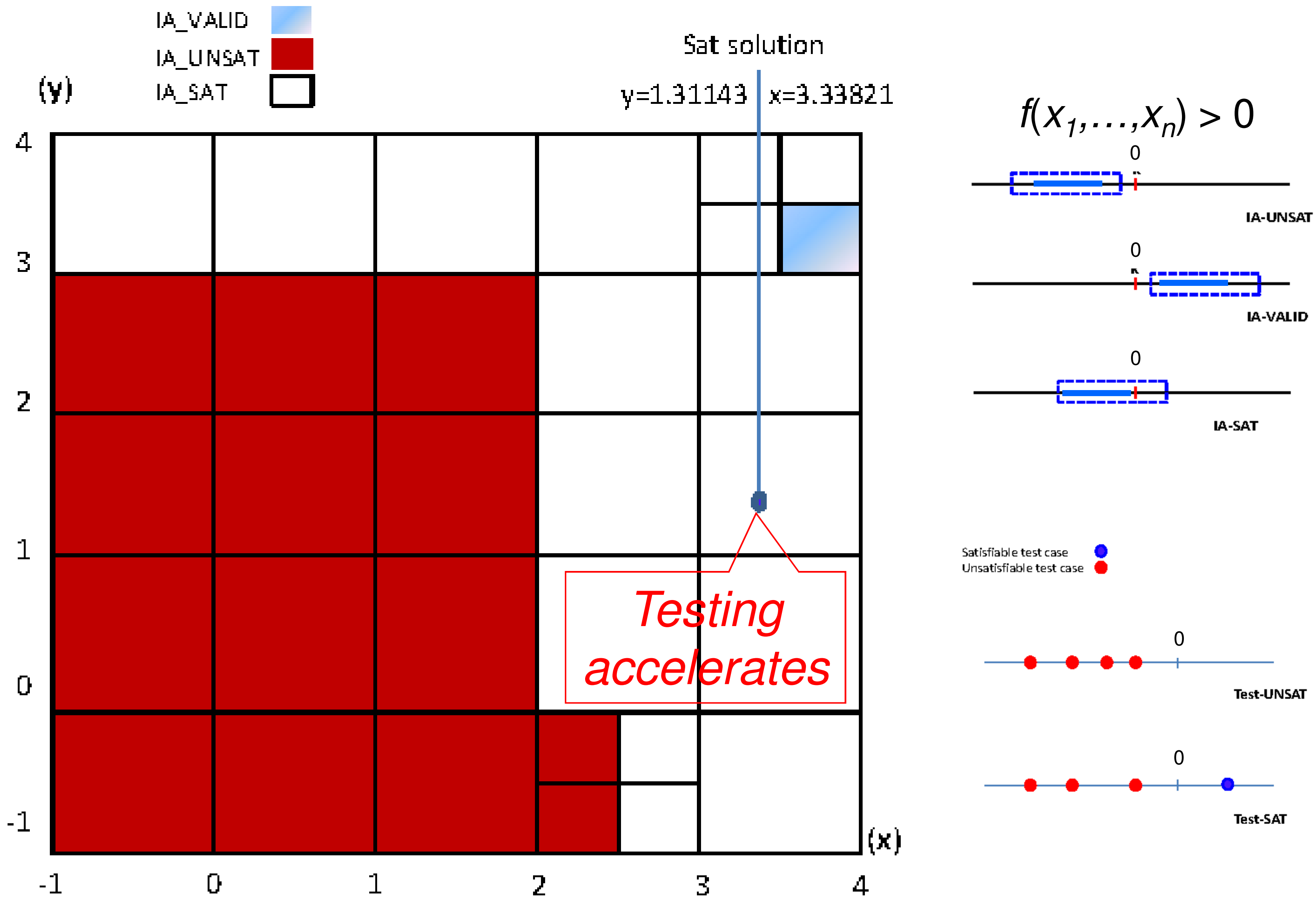


Over-approximation
Interval Arithmetic (IA)

Under-approximation
Testing

Refinement (Decomposition)
$x \in (l,h) \Rightarrow x \in (l,m) \vee x \in (m,h)$

Box decomposition (starting from 1 large box)



Soundness / (relative) completeness of raSAT

- **Th**. Let $\exists \underbrace{x_1 \in (l_1, h_1) \ldots x_n \in (l_n, h_n)}_{I_1, I_2, \ldots, I_n} . \wedge_j \underbrace{f_j > 0}_{P}$

  Let $D_j = \{ (x_1, \ldots, x_n) \mid f_j (x_1, \ldots, x_n) > 0 \}$

  ✓**Soundness**: If raSAT checks SAT (resp. UNSAT), it is really SAT (resp. UNSAT)

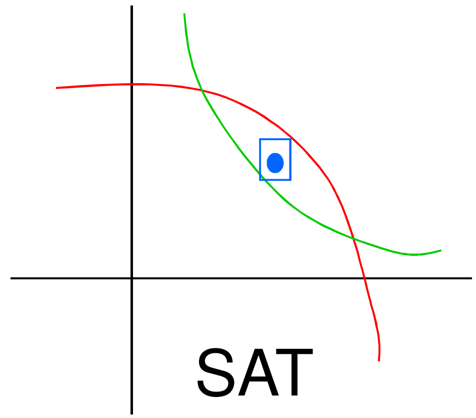
  ✓**Completeness**: Assume fair box decomposition

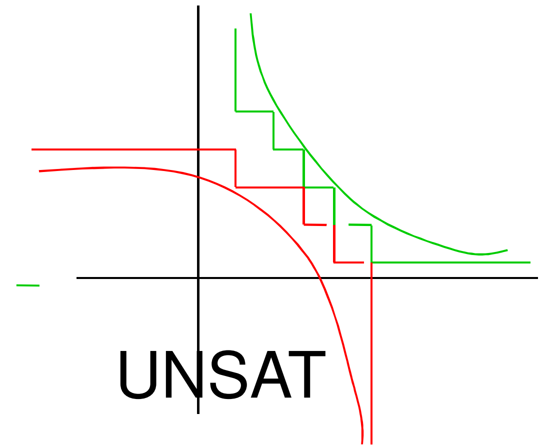
  – If SAT, raSAT eventually finds SAT-instance in $\mathbb{Q}$.

  – If closure(D_i)∩closure(D_j) =φ($i \neq j$) and closure(I_i) is compact, raSAT eventually detects UNSAT.

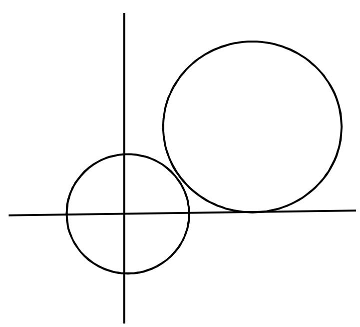
- **Alternative**: δ-equality ($x=0 \Rightarrow -\delta<x<\delta$) in **dReal.**

Completeness ideas

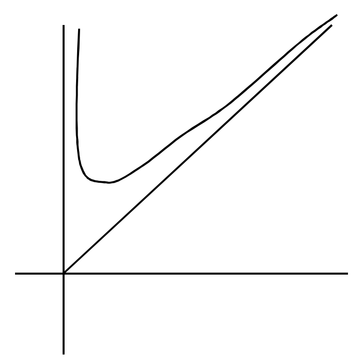




Failure to detect UNSAT



Toughing case
⇒ Groebner basis

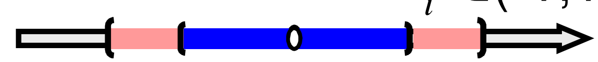


Converging case
(unbounded intervals)

raSAT implementation design

Interval arithmetic design

- Affine interval (AI) [Stolfi 1997]
  - ✓Use noise symbols ε, interpreted as ε∈(-1,1).
  - ✓Precision incomparable between CI and AI.
  - ✓AI fails for open-ended boxes; (∞+∞ε) as (0,∞)

	Classical interval (CI)(1)	Affine interval (AI)(2)
Def	$\bar{x} = [lo, hi]$ 	$x = x_0 + x_1 \varepsilon_1 + \ldots + x_n \varepsilon_n$ $\varepsilon_i \in (-1,1)$ 
Arithmetic (e.g., x – x, x × x)	$[1,3] - [1,3] = [-2,2]$	$(2+\varepsilon_1) - (2+\varepsilon_1) = 0$
	$[1,3] \times [1,3] = [1,9]$	$(2+\varepsilon_1) \times (2+\varepsilon_1) = 4 + 4\varepsilon_1 + \varepsilon_1 \varepsilon_1$ → ε_2

raSAT implementation design

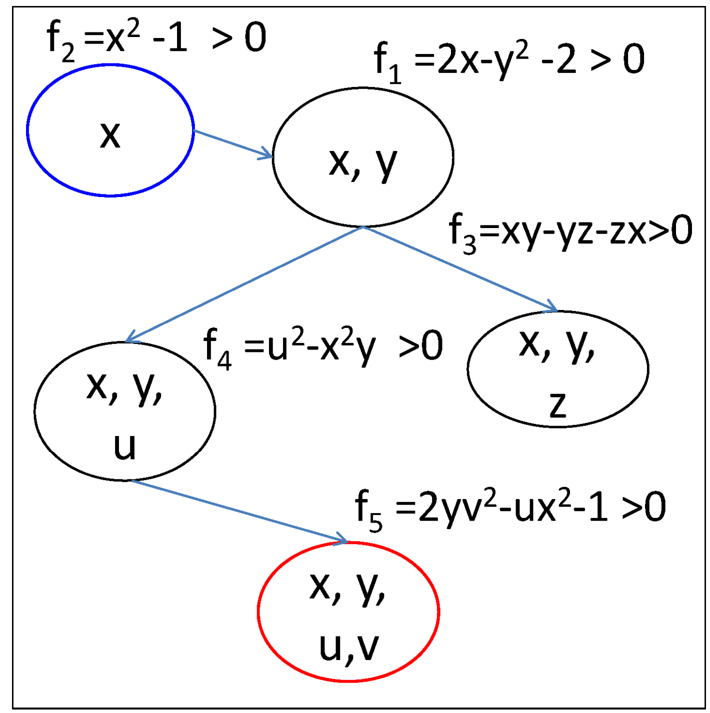
- **raSAT** procedure
  1. Starts with a bounded box, e.g., $(0,\infty) \Rightarrow (0,10)$, and compute with AI.
  2. If SAT, confirm it with an error bound guaranteed floating point library iRRAM (SAT confirmation)
  3. If UNSAT, check the whole box with CI.

- Not implemented
  - ✓ Equality handling (intermediate value theorem, Groebner basis)

    $\Rightarrow$ Adhoc equality with intergers.
  - ✓ UNSAT confirmation (related to UNSAT core)

Explosion by box decomposition

- If *n*-variables are decomposed
  - ✓ 2^n boxes to explore!



- Priority on variables.
  1. Choice of atomic polynomial inequality (API)
     ⇒ Dependency among unsatisfied APIs.
  2. Choice of variables in an API
     ⇒ Sensitivity, e.g. $x^3 - 2xy$ for $x = 1+\varepsilon_1$, $y = 2 + \varepsilon_2$

"x" is the most sensible

$$(-4, -\tfrac{11}{4}) + (-\tfrac{1}{4}, 0)\epsilon_1 - 2\epsilon_2 + 3|\epsilon_1| + (-2, 2)\epsilon_{\pm}$$

Greater-than-equal, equality handling

- Greater-than-equal ≧
  - ✓Strict-SAT: $f > \delta$ instead of $f \geqq 0$, for some $\delta > 0$.
  - ✓UNSAT: $f > -\delta$ instead of $f \geqq 0$

- Equality =
  - ✓Intermediate value theorem
    - –Currently, only for single equality

      $\exists x_1 \in (l_1, h_1)\ x_2 \in (l_2, h_2)\ .\wedge_j f_j > 0 \wedge g = 0\ )$
  - ✓Groebner basis
    - –Future work

Preliminary experiments on SMTlib

- Mostly focus on Zankl family (166 benchmarks)
  - ✓ Currently around 50 (depending on tuning), where
    - 89 by Z3 4.3, 50 by Mathematica, 46 by miniSMT.
  - ✓ Remarkable SAT examples (other tools fail)
    - matrix-2-all-8 (17vars, 25APIs, 56 max |API|)
    - matrix-5-all-7 (267vars, 384APIs, 822 max |API|)
  - ✓ Other benchmarks often contains ≧, =.

- Stronger than Z3 4.3
  - ✓ When the maximal degree of an API > 15
  - ✓ When the number of variables in an API > 15
  - ✓ When the maximal length of an API > 50

  Z3 4.3 has good strategy to choose a subset of APIs.

Related interval arithmetic-based tools

- iSAT3
  - ✓Classival interval
  - ✓No under approximation (testing)
    - –SAT by IA-valid only

- dReal
  - ✓Sharing approximation idea
  - ✓Only with interval arithmetic
  - ✓δ-SAT does not imply SAT (aim different)

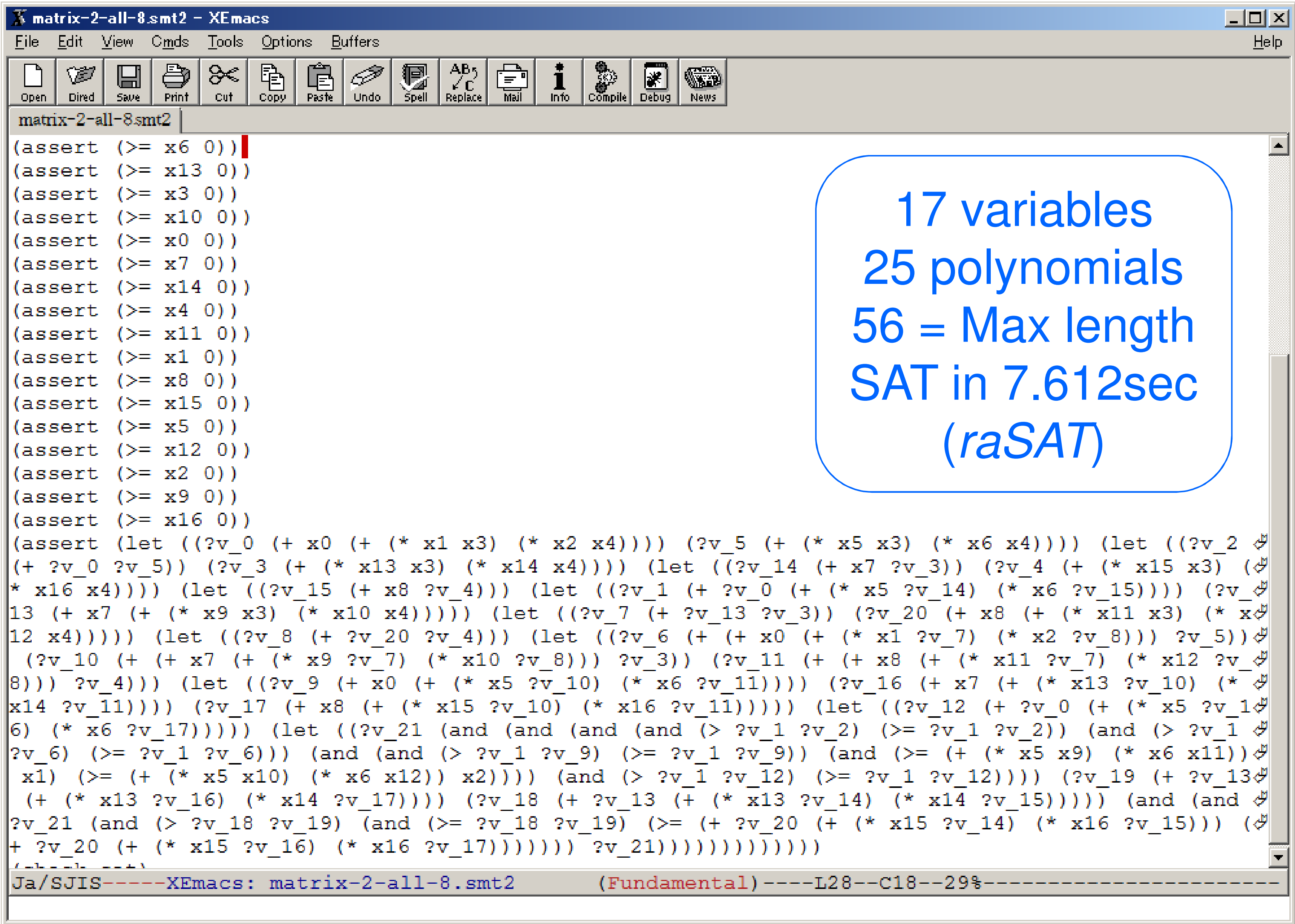
Conclusion and future works

- **raSAT** for QF_NRA is presented.
  - ✓With single methodology: **raSAT** loop
  - ✓Experiments are preliminary, some remarkable examples
  - ✓Participated SMTcomp 2014 (4^{th} among 4)

- ToDo
  - ✓Implementation revision (to accept disjunctive polynomial constraints), strategy tuning
  - ✓UNSAT core improvement
  - ✓Equality handling (Int. value Th., Groebner basis)
  - ✓Mixed integers.

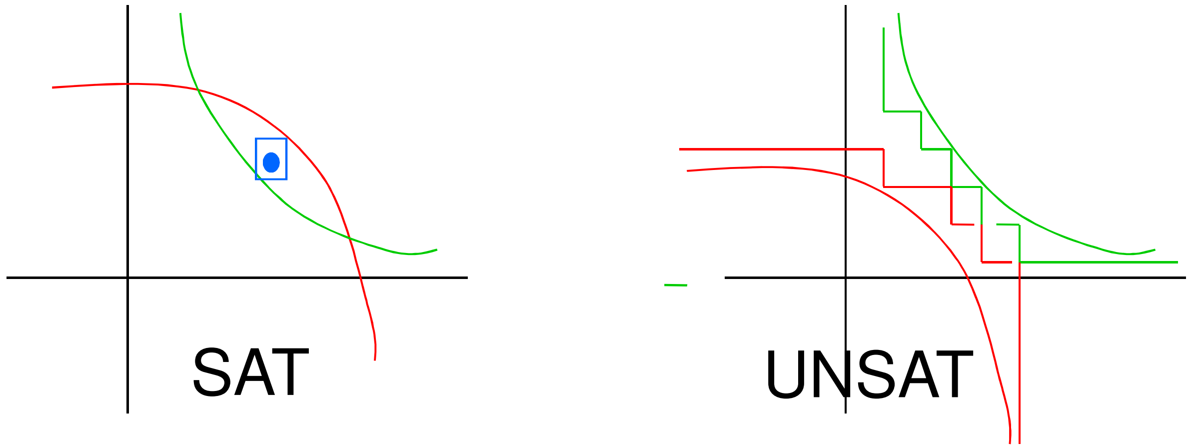
Thank you!

Benchmark example: zankl/matrix-2-all-8



17 variables
25 polynomials
56 = Max length
SAT in 7.612sec
(*raSAT*)

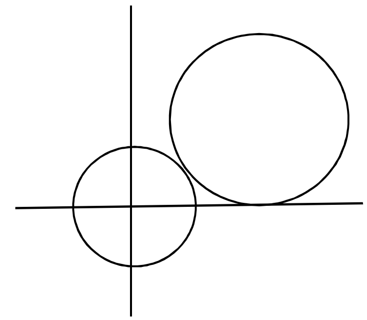
Completeness proof ideas



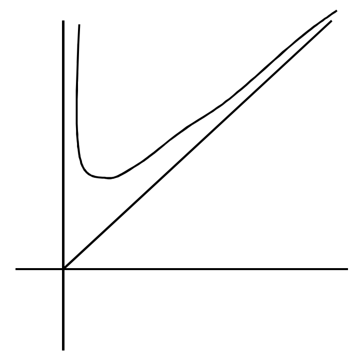
- SAT: if $f_1>0$ and $f_2>0$ have intersection, there must be a neighborhood of an internal point.
- UNSAT: if $f_1 \geqq 0$ and $f_2 \geqq 0$ are UNSAT and closure s of intervals are compact, we have lower bound of distance $\delta>0$ between D_1 and D_2.
  - ✓By induction on the number of refinement steps.

Where UNSAT limitation comes

- Boundary conditions (kissing situation)
  - ✓ $x^2+y^2 < 2^2 \wedge (x-4)^2+(y-3)^2 < 3^2$
  - $\Rightarrow$ two closures intersect at (1.6,1.2)

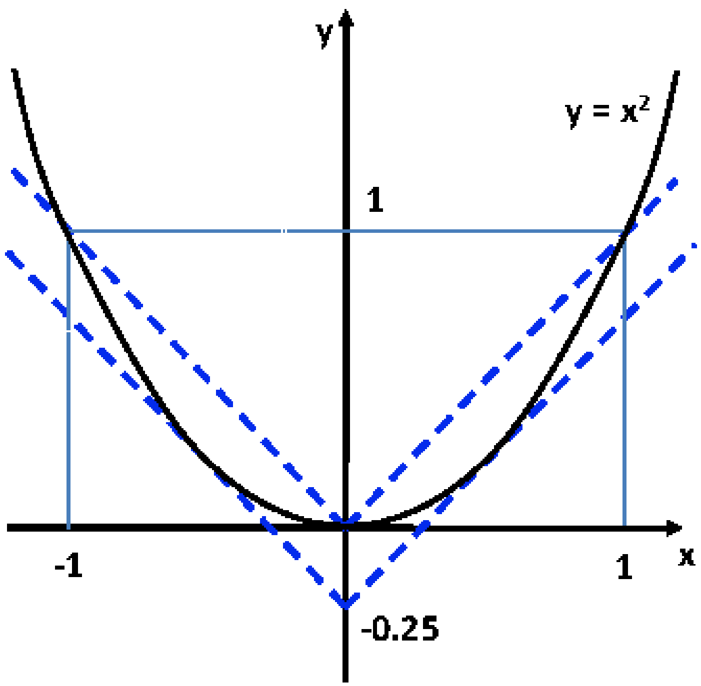


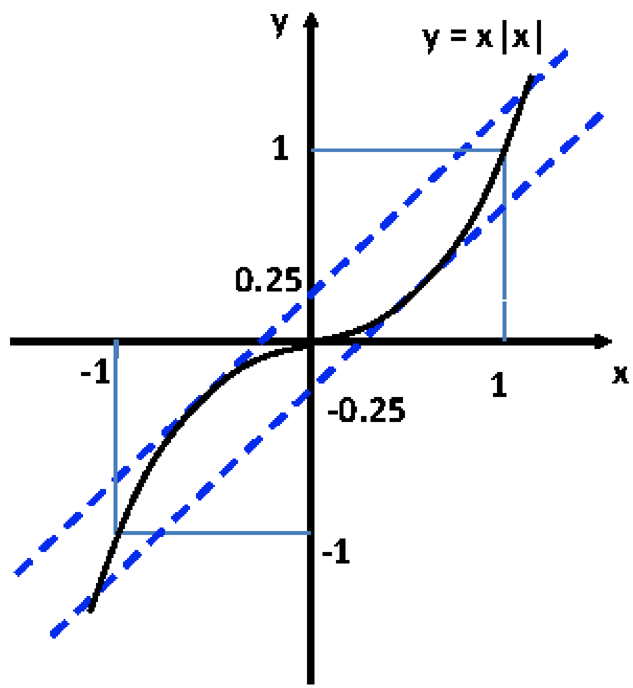
- Convergence
  - ✓ $y > x + 1/x \wedge y < x \wedge x > 0$
  - $\Rightarrow$ x needs an upper bound.



Chebyshev affine interval (Khanh-Ogawa 12)

- Focusing on precision of mulatiplications of the same noise symbol by linear approximations.





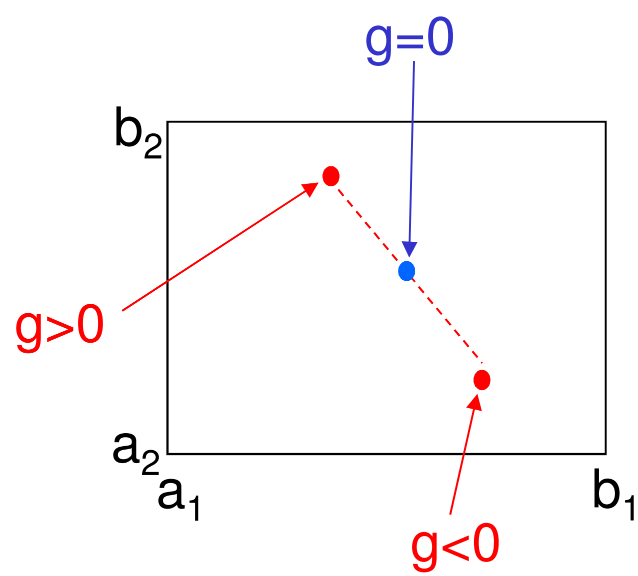
$|\varepsilon| - 1/4 \leqq \varepsilon^2 < |\varepsilon|$

$\varepsilon - 1/4 \leqq \varepsilon \cdot |\varepsilon| \leqq \varepsilon + 1/4$

Equality (=) handling by intermediate value th.

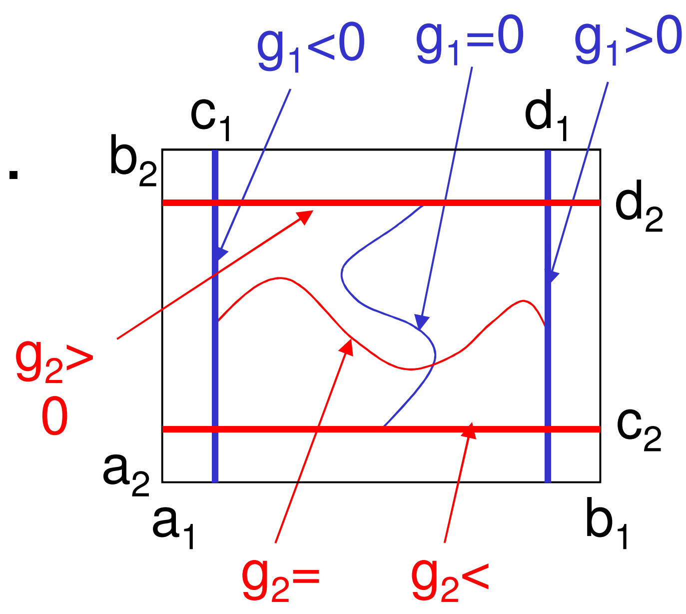
- Idea: Let $\exists x_1 \in (l_1,h_1)\ x_2 \in (l_2,h_2)\ .\wedge_j f_j > 0 \wedge g = 0$
  - ✓ Assume that $x_1 \in (a_1,b_1)\ x_2 \in (a_2,b_2)\ .\wedge_j f_j > 0$ is IA-valid.
  - ✓ We found two points in $(a_1,b_1) \times (a_2,b_2)$ such that $g<0$ and $g>0$.

- We see there are $g=0$. (SAT)
  (By intermediate value theorem)
  - ✓ UNSAT by $-\delta < g < \delta$ instead of $g = 0$



Equality handling : Multiple equality (idea)

- For $\exists x_1 \in (l_1,h_1)\ x_2 \in (l_2,h_2)\ .(\wedge_j f_j > 0) \wedge g_1=0 \wedge g_2=0$, assume that
  - ✓ $x_1 \in (a_1,b_1)\ x_2 \in (a_2,b_2)\ .\wedge_j f_j > 0$ is IA-valid.
  - ✓ c_1,d_1 with $g_1<0$ on $\{c_1\} \times (a_2,b_2)$, $g_1>0$ on $\{c_2\} \times (a_2,b_2)$
  - ✓ c_2,d_2 with $g_2<0$ on $(a_1,b_1) \times \{d_1\}$, $g_2>0$ on $(a_1,b_1) \times \{d_2\}$
- Then, we see there are $g_1=g_2=0$.



Requires
"|Vars| ≧ |equations|"

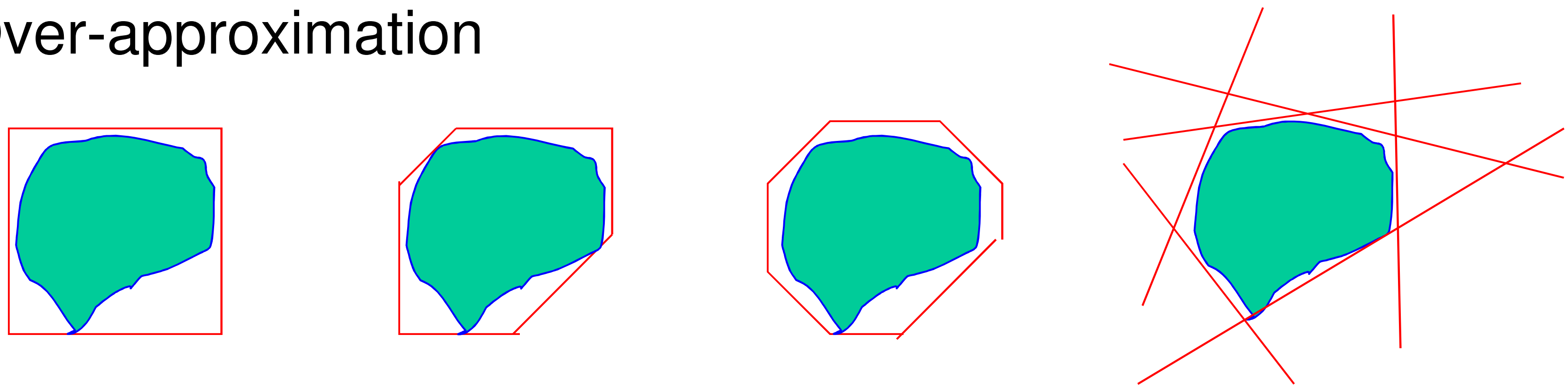
Groebner basis (Buchberger 65)

- Groebner basis is for computing quotient of ideals.
  - ✓Starting from given basis of ideals (with WFO on monomials).
  - ✓Completion for polynomials (in which variables are not substituted and completion always succeed).

- E.g., $\mathbb{Q}[z,w]/(z^2 - 3, zw^2 + 2w - 3z)$ with $w > z$.
  - →Regard them $z^2 \to 3$, $zw^2 \to -2w + 3z$
  - →Critical pair $(3w^2, -2zw + 3z^2)$
  - →New rule $3w^2 \to -2zw + 9$, …
  - →Finally, we obtain $z^2 \to 3$, $3w^2 \to -2zw + 9$ and $\mathbb{Q}[z,w]/(z^2 - 3, 3w^2 + 2zw - 9)$.

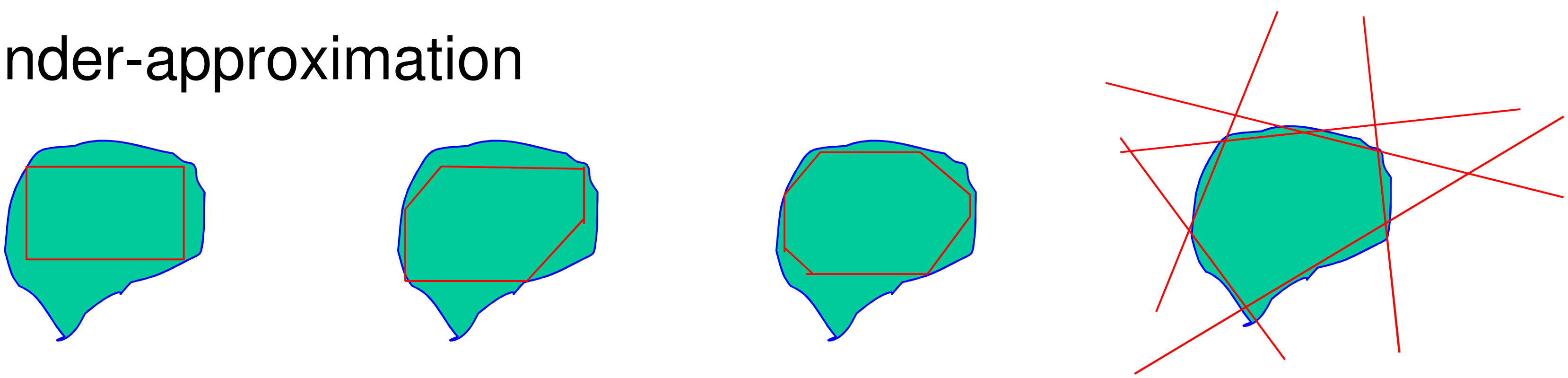
Linear approximations

Over-approximation



Interval *Zone* *Octagon* *Polyhedra*

Under-approximation



Interval *Zone* *Octagon* *Polyhedra*