Constructing single cylindrical algebraic cells

Bath

Jasper Nalbach, Erika Ábrahám, Philippe Specht, Christopher W. Brown, James H. Davenport, Matthew England

August 2023

Introduction

Introduction

- Satisfiability-modulo-theories is a technique for checking the existential fragment of first-order logic

Introduction

- Satisfiability-modulo-theories is a technique for checking the existential fragment of first-order logic
- Non-linear arithmetic formulas are Boolean combinations of polynomial constraints

Introduction

- Satisfiability-modulo-theories is a technique for checking the existential fragment of first-order logic
- Non-linear arithmetic formulas are Boolean combinations of polynomial constraints
- Recent developments

Introduction

- Satisfiability-modulo-theories is a technique for checking the existential fragment of first-order logic
- Non-linear arithmetic formulas are Boolean combinations of polynomial constraints
- Recent developments
  - SMT solving: Sample-based conflict-driven algorithms based on the single cell construction are the state-of-the-art for NRA solving

Introduction

- Satisfiability-modulo-theories is a technique for checking the existential fragment of first-order logic
- Non-linear arithmetic formulas are Boolean combinations of polynomial constraints
- Recent developments
  - SMT solving: Sample-based conflict-driven algorithms based on the single cell construction are the state-of-the-art for NRA solving
    - NLSAT [Jovanović, De Moura 2012], Cylindrical algebraic coverings [Ábrahám et al 2021], Refinement-based single cell construction [Brown, Košta 2015]

Introduction

- Satisfiability-modulo-theories is a technique for checking the existential fragment of first-order logic
- Non-linear arithmetic formulas are Boolean combinations of polynomial constraints
- Recent developments
  - SMT solving: Sample-based conflict-driven algorithms based on the single cell construction are the state-of-the-art for NRA solving
    - NLSAT [Jovanović, De Moura 2012], Cylindrical algebraic coverings [Ábrahám et al 2021], Refinement-based single cell construction [Brown, Košta 2015]
  - Computer algebra: Various statements for speeding up different cases from CAD theory

Introduction

- Satisfiability-modulo-theories is a technique for checking the existential fragment of first-order logic
- Non-linear arithmetic formulas are Boolean combinations of polynomial constraints
- Recent developments
  - SMT solving: Sample-based conflict-driven algorithms based on the single cell construction are the state-of-the-art for NRA solving
    - NLSAT [Jovanović, De Moura 2012], Cylindrical algebraic coverings [Ábrahám et al 2021], Refinement-based single cell construction [Brown, Košta 2015]
  - Computer algebra: Various statements for speeding up different cases from CAD theory
    - McCallum, Brown-McCallum, Equational constraints, Lazard, ...

Non-linear arithmetic

Is a given Boolean combination of polynomial constraints satisfiable?

Non-linear arithmetic

Is a given Boolean combination of polynomial constraints satisfiable?

In this talk: only conjunctions!

$$0.5x_1 + 0.5 - x_2 > 0 \wedge x_1^2 + x_2^2 - 1 < 0 \wedge 0.5x_1 - 0.5 - x_2 < 0$$

Non-linear arithmetic

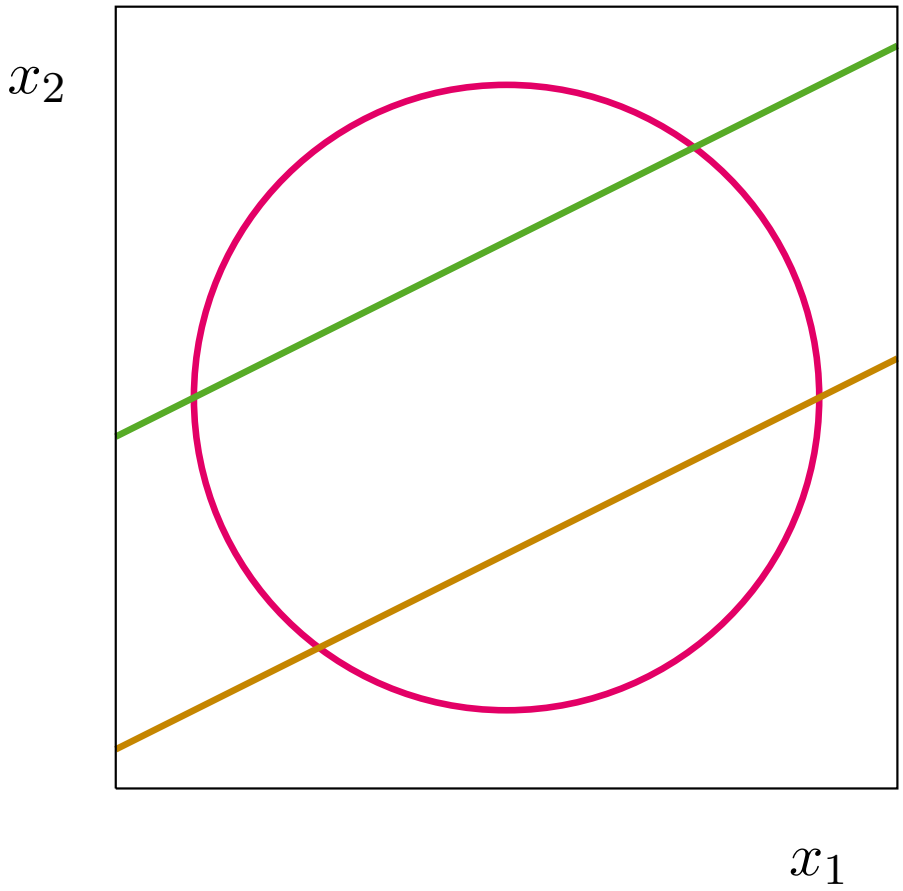
Is a given Boolean combination of polynomial constraints satisfiable?

In this talk: only conjunctions!

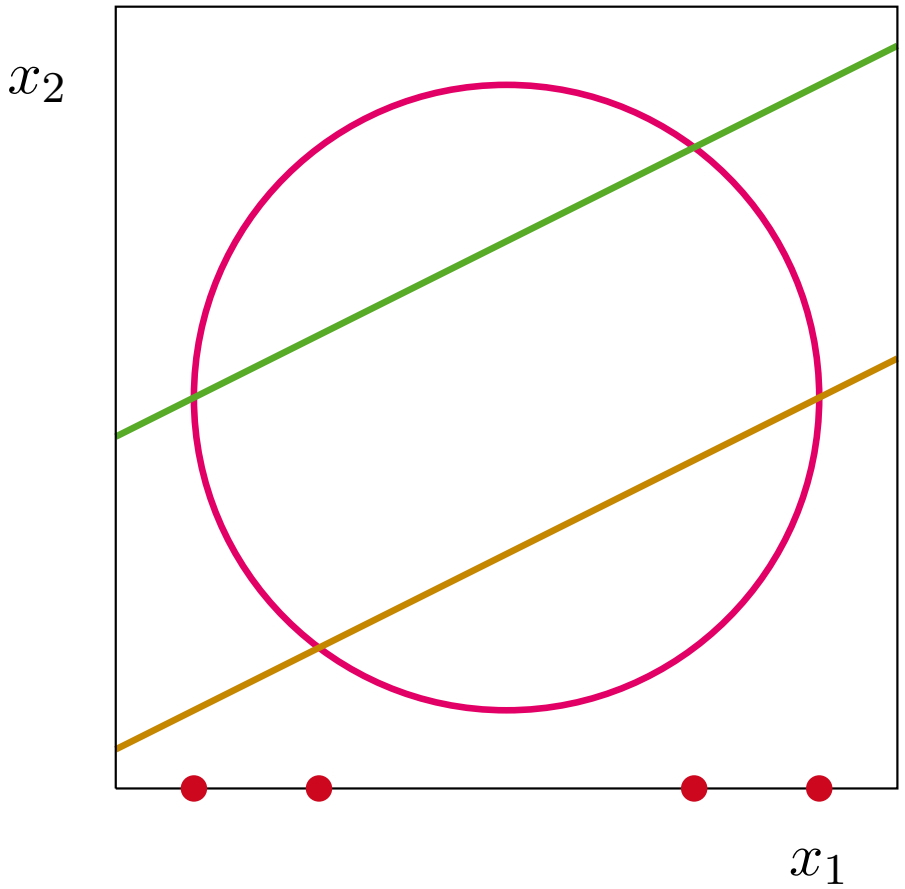
$$0.5x_1 + 0.5 - x_2 > 0 \wedge x_1^2 + x_2^2 - 1 < 0 \wedge 0.5x_1 - 0.5 - x_2 < 0$$

Traditional idea: Generate sign invariant regions of defining polynomials and check whether they satisfy the sign condition

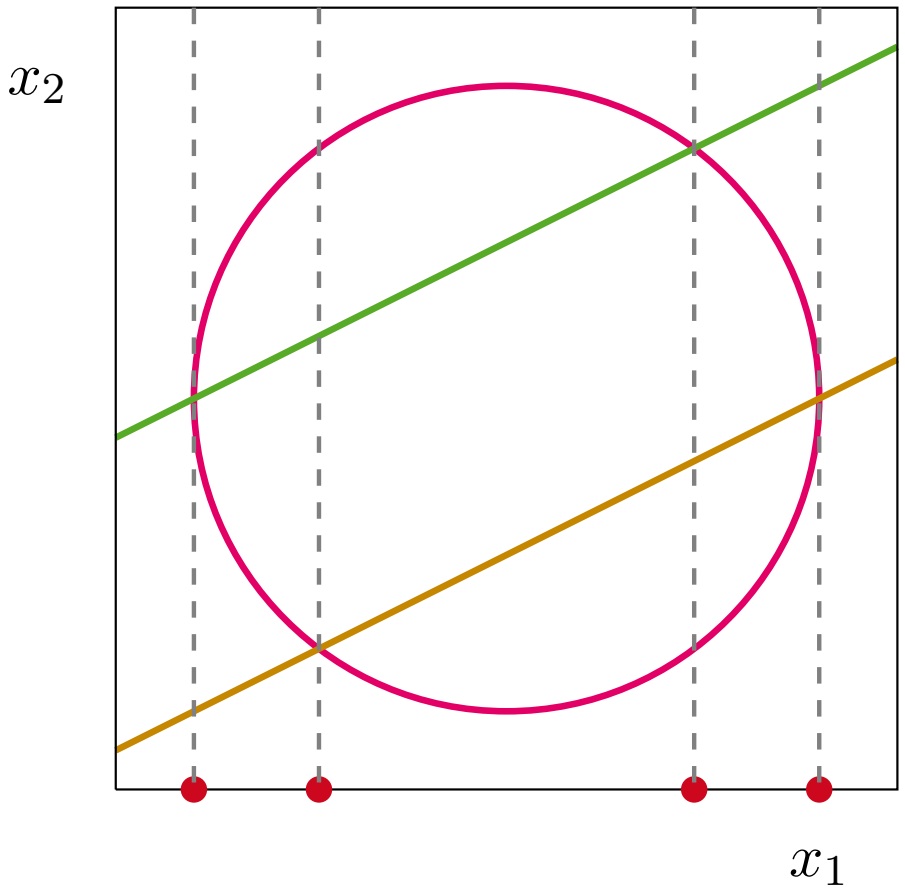
Cylindrical algebraic decomposition (CAD) [Collins 1975]



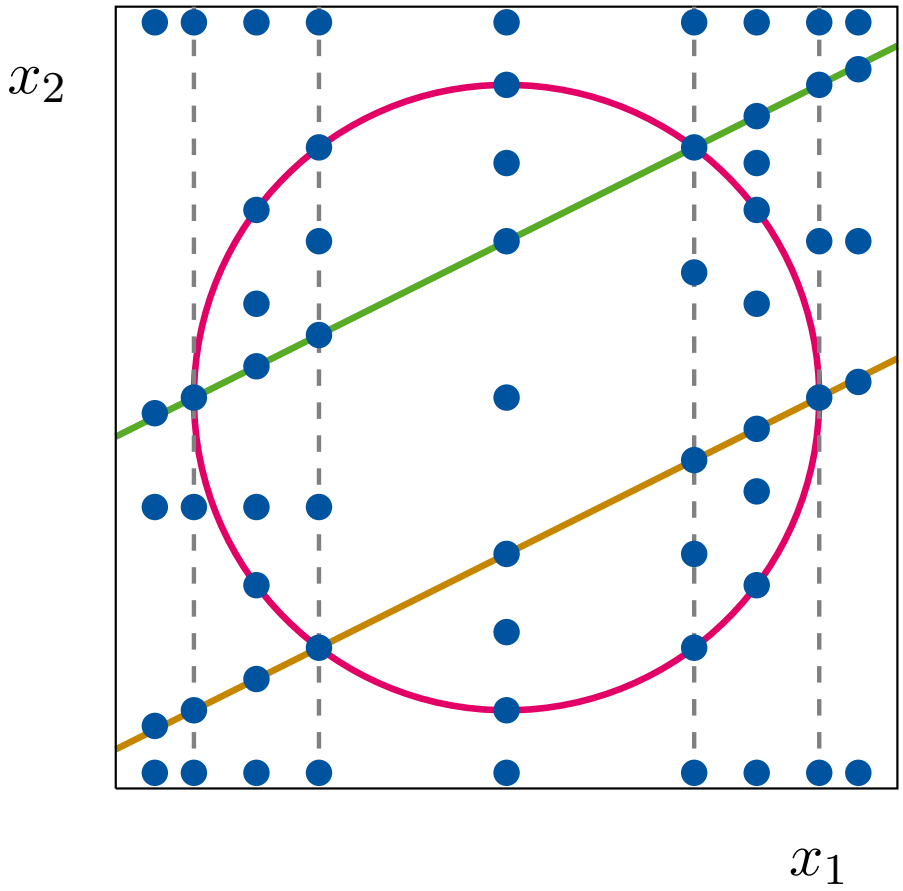
Cylindrical algebraic decomposition (CAD) [Collins 1975]



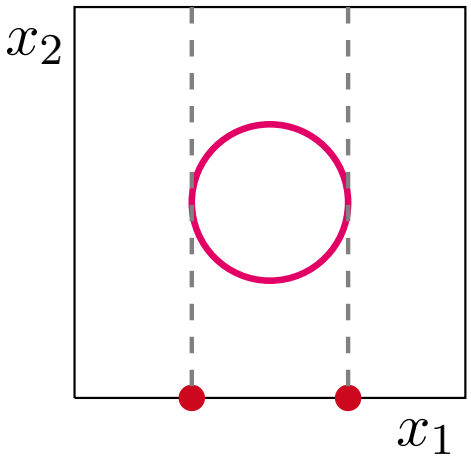
Cylindrical algebraic decomposition (CAD) [Collins 1975]



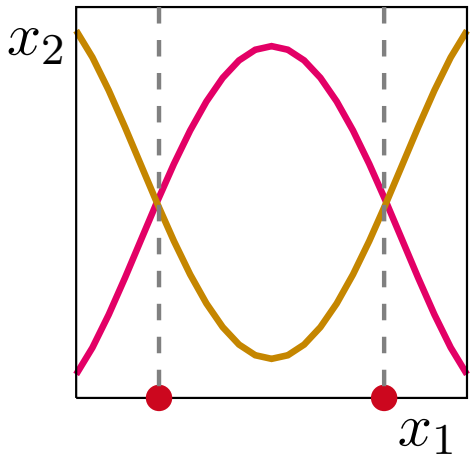
Cylindrical algebraic decomposition (CAD) [Collins 1975]



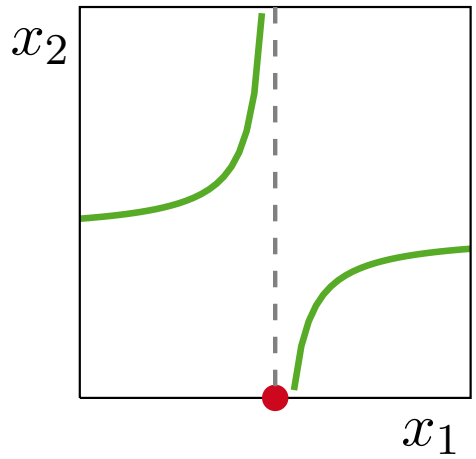
CAD projection tools



Discriminant



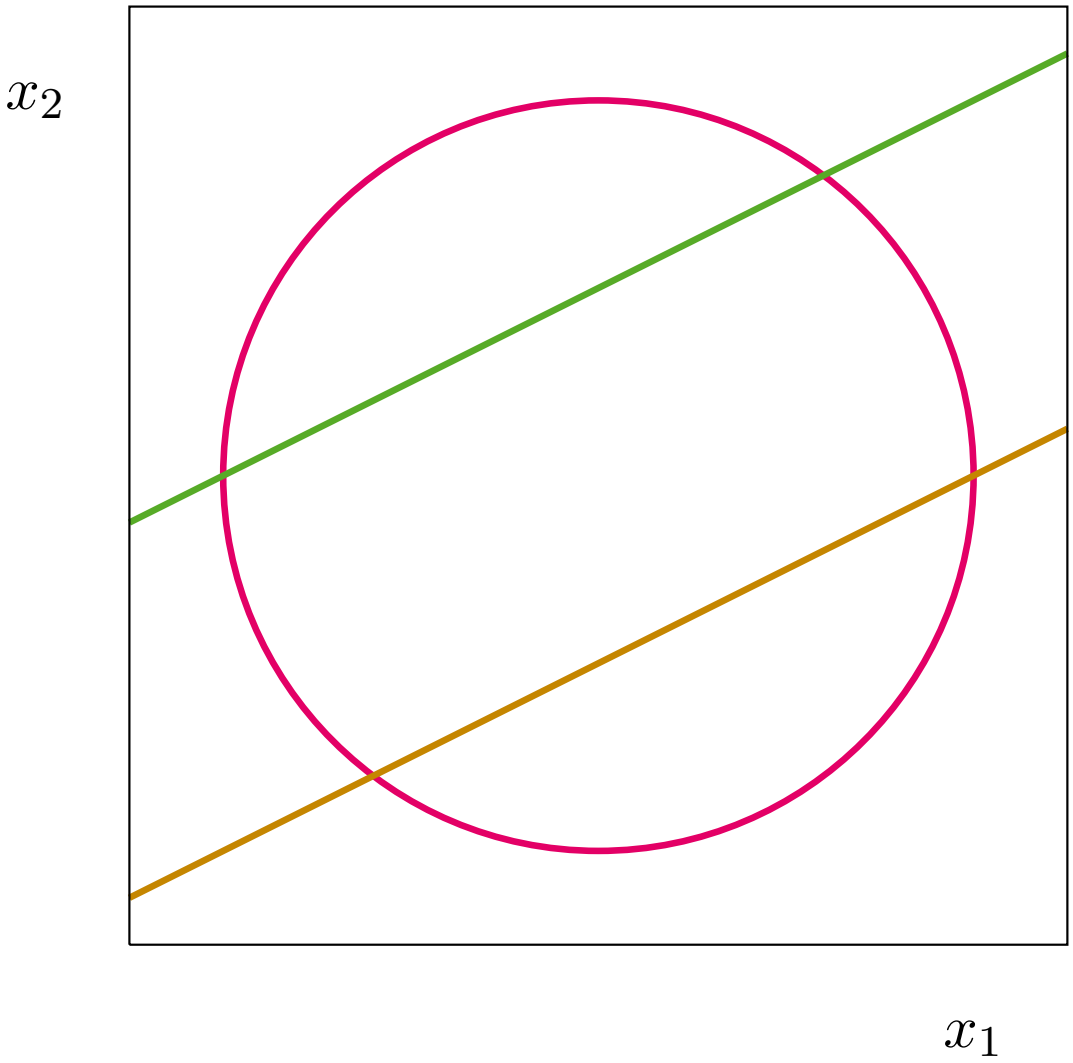
Resultant



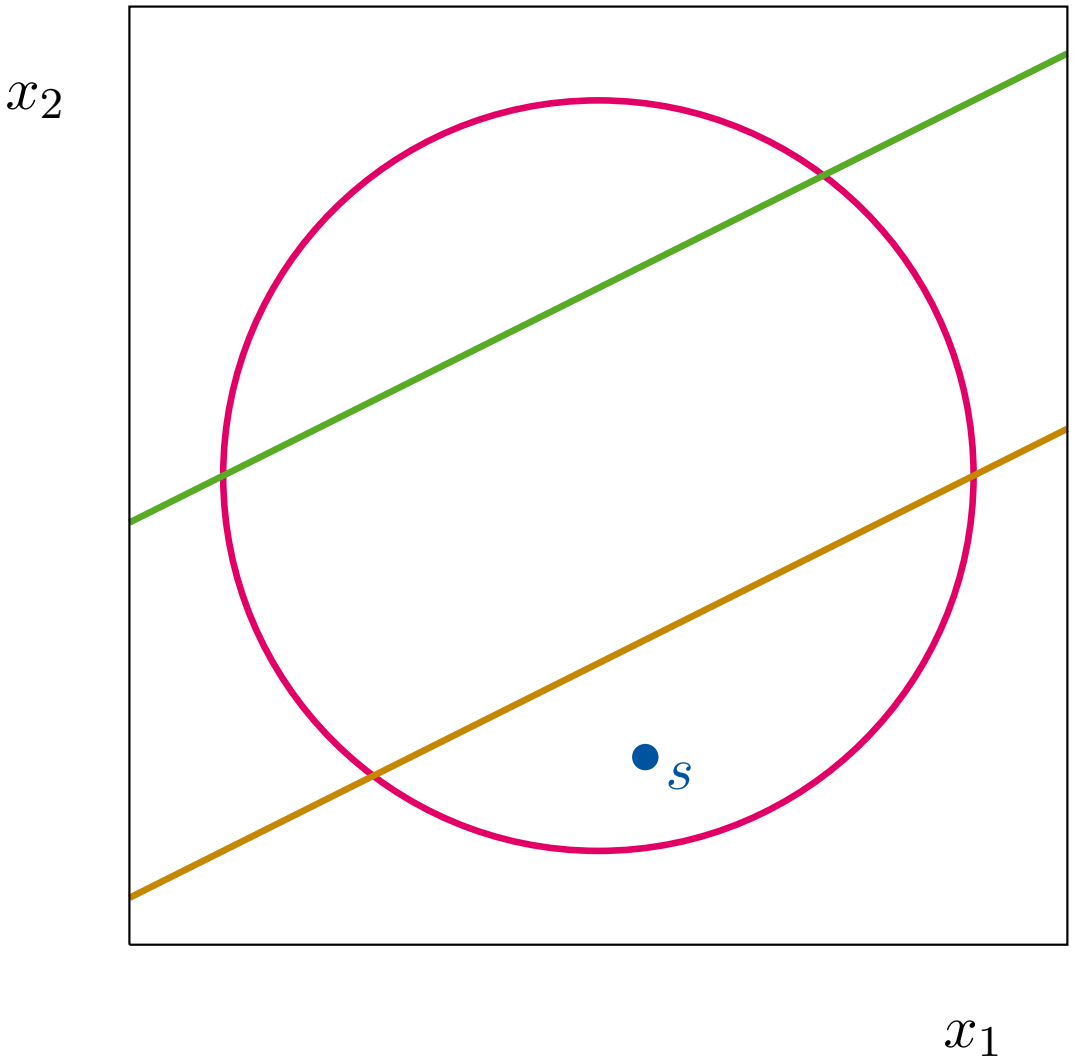
Leading coefficients

... and further coefficients

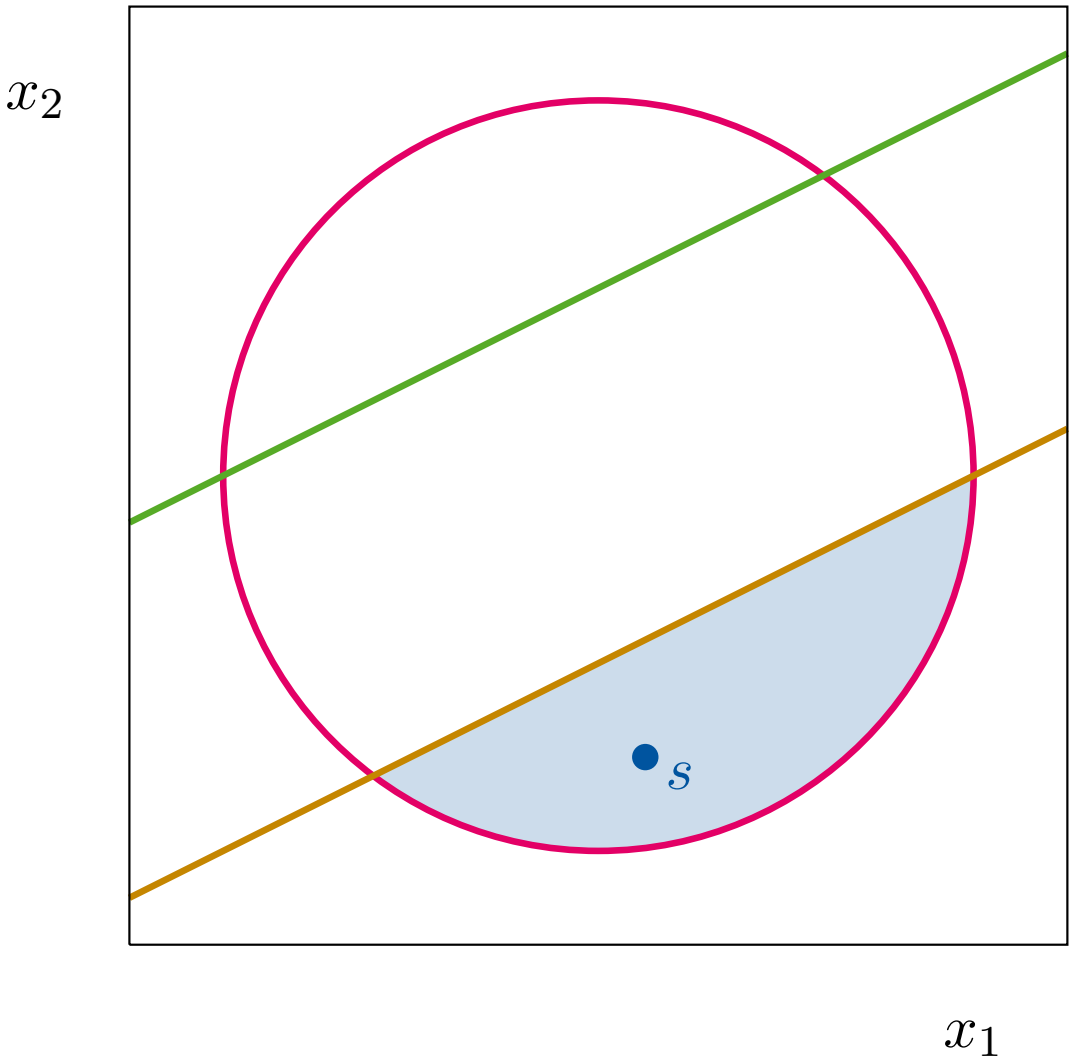
Intuition of sample-based algorithms



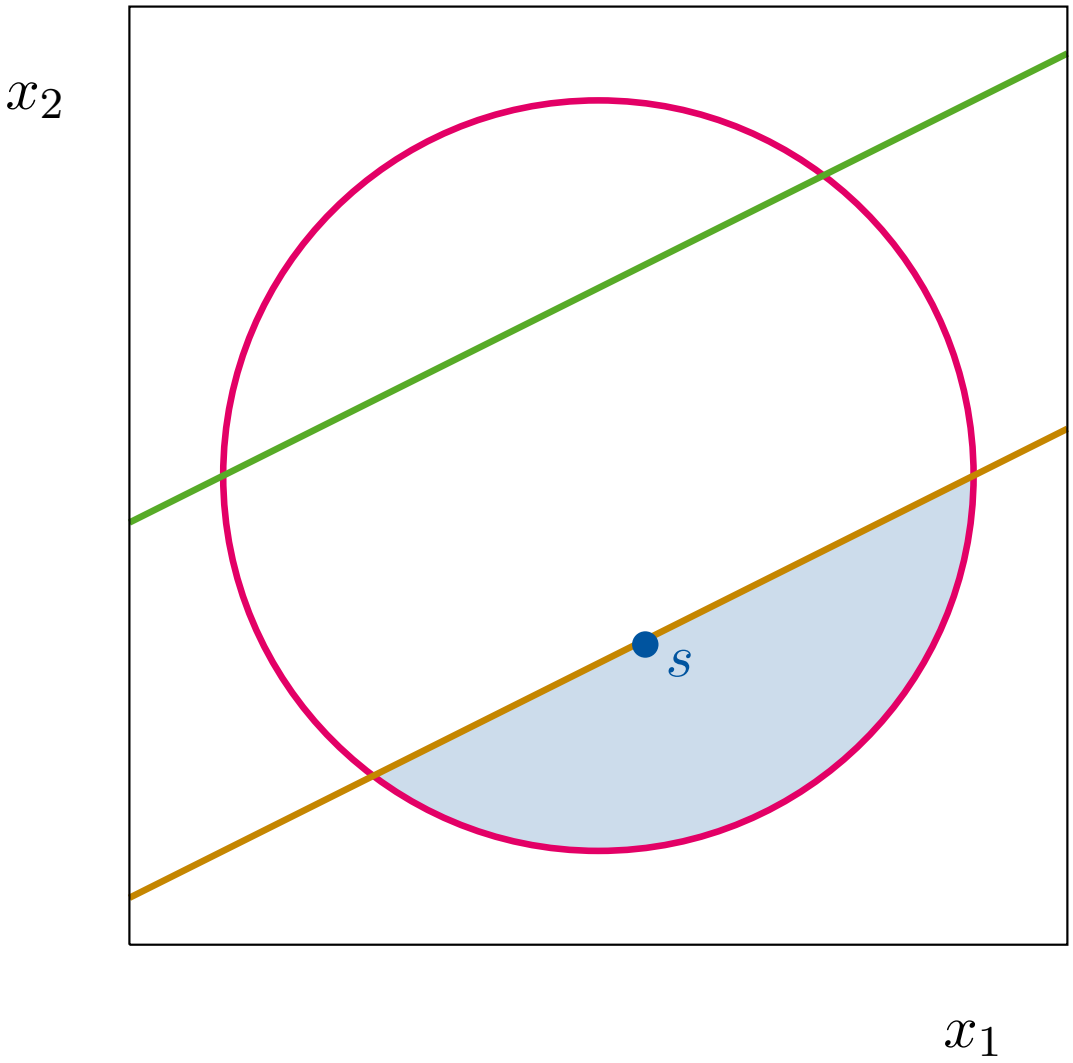
Intuition of sample-based algorithms



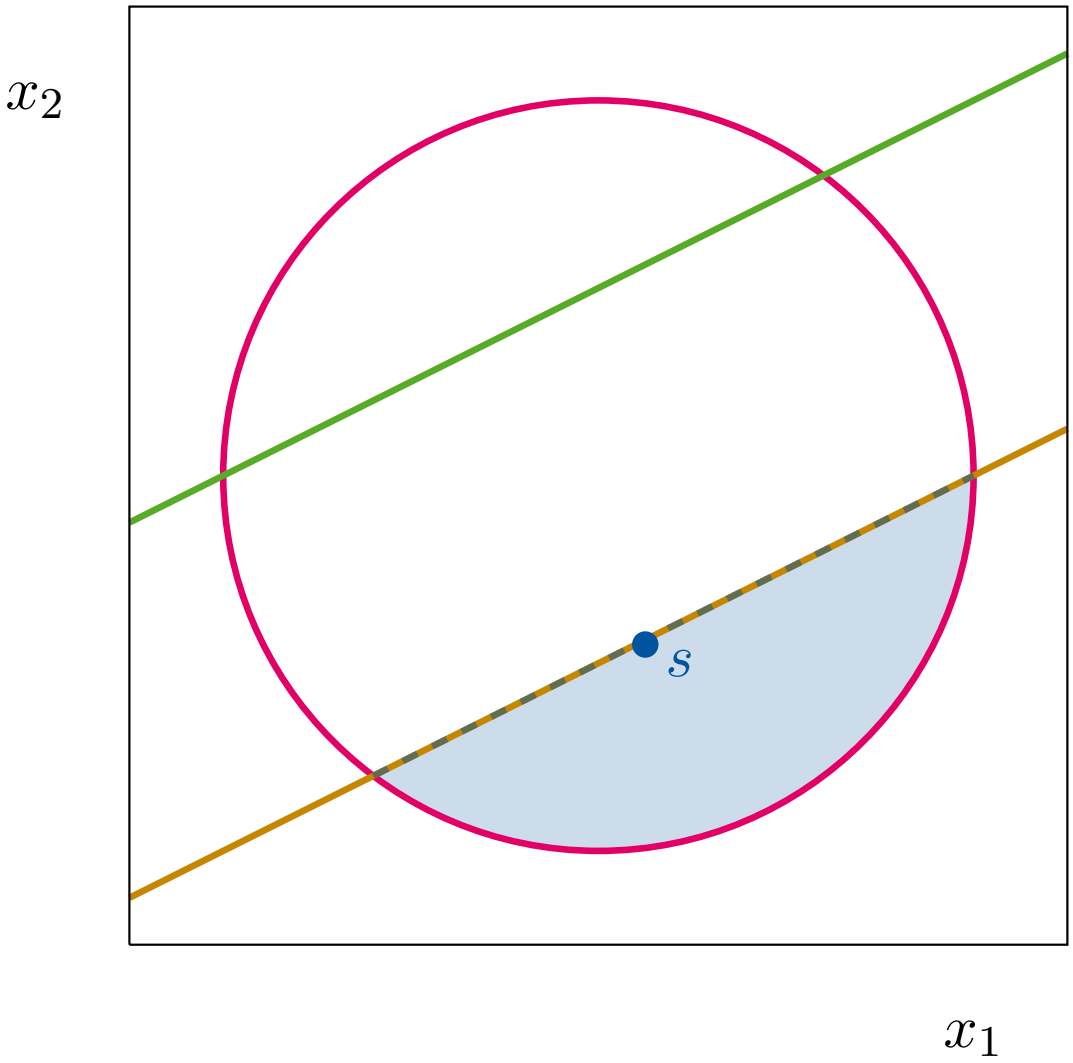
Intuition of sample-based algorithms



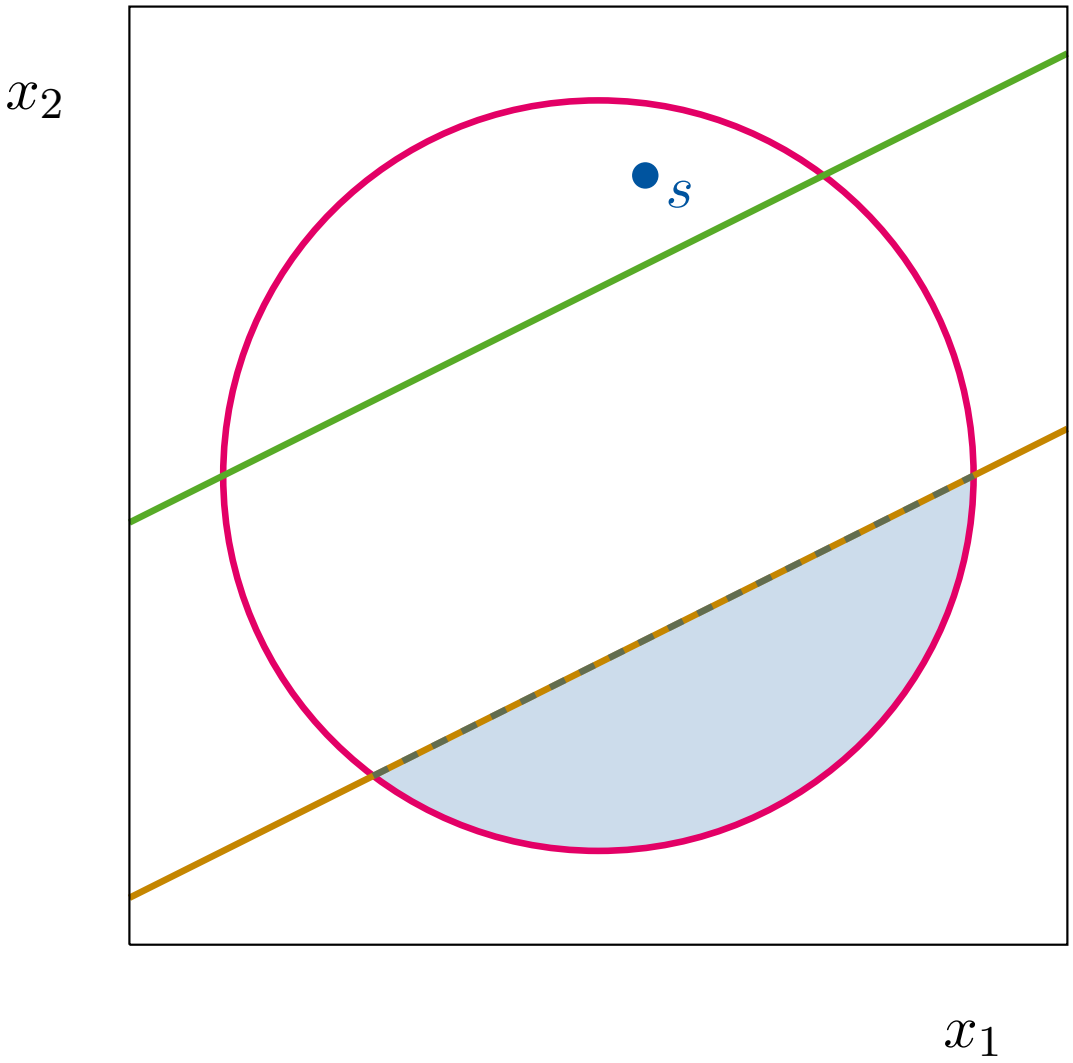
Intuition of sample-based algorithms



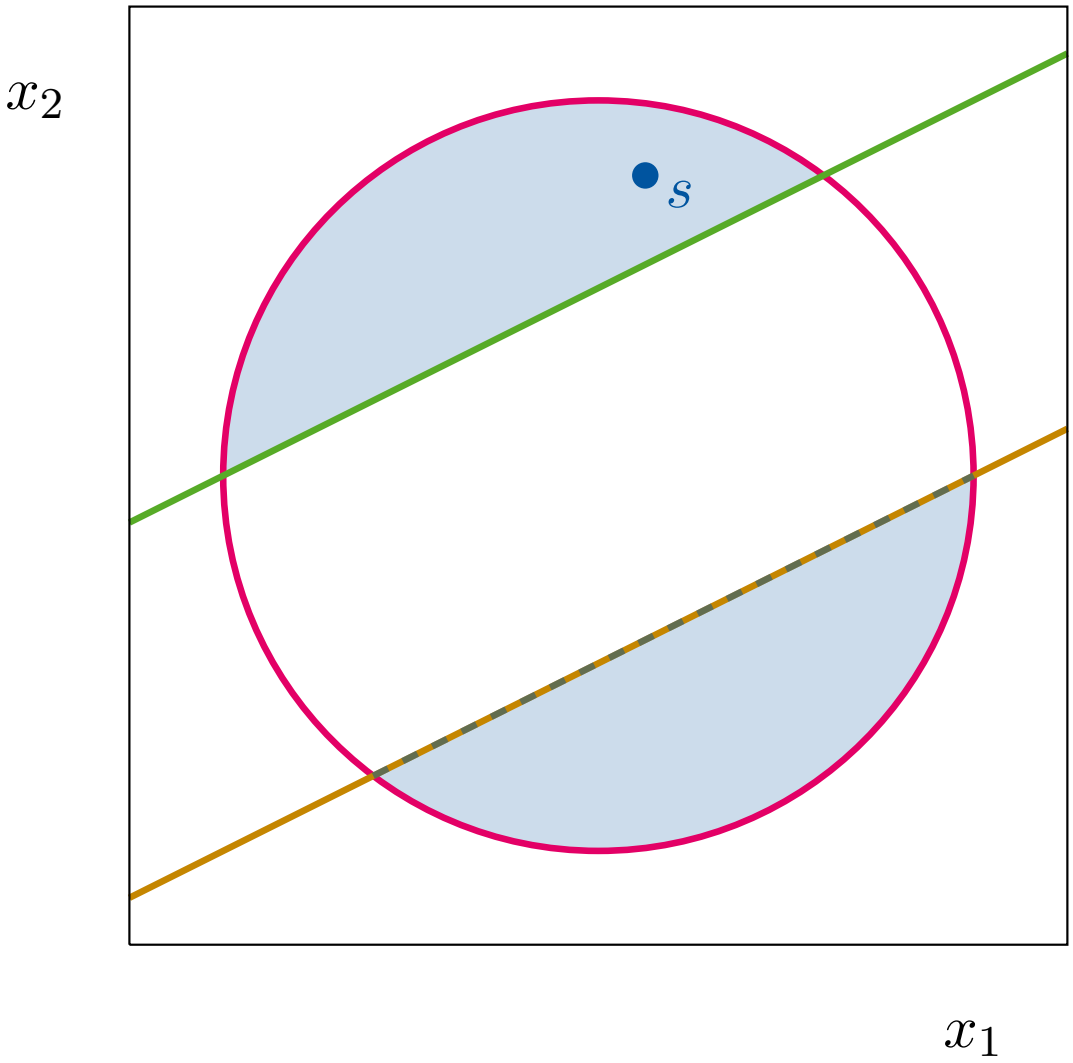
Intuition of sample-based algorithms



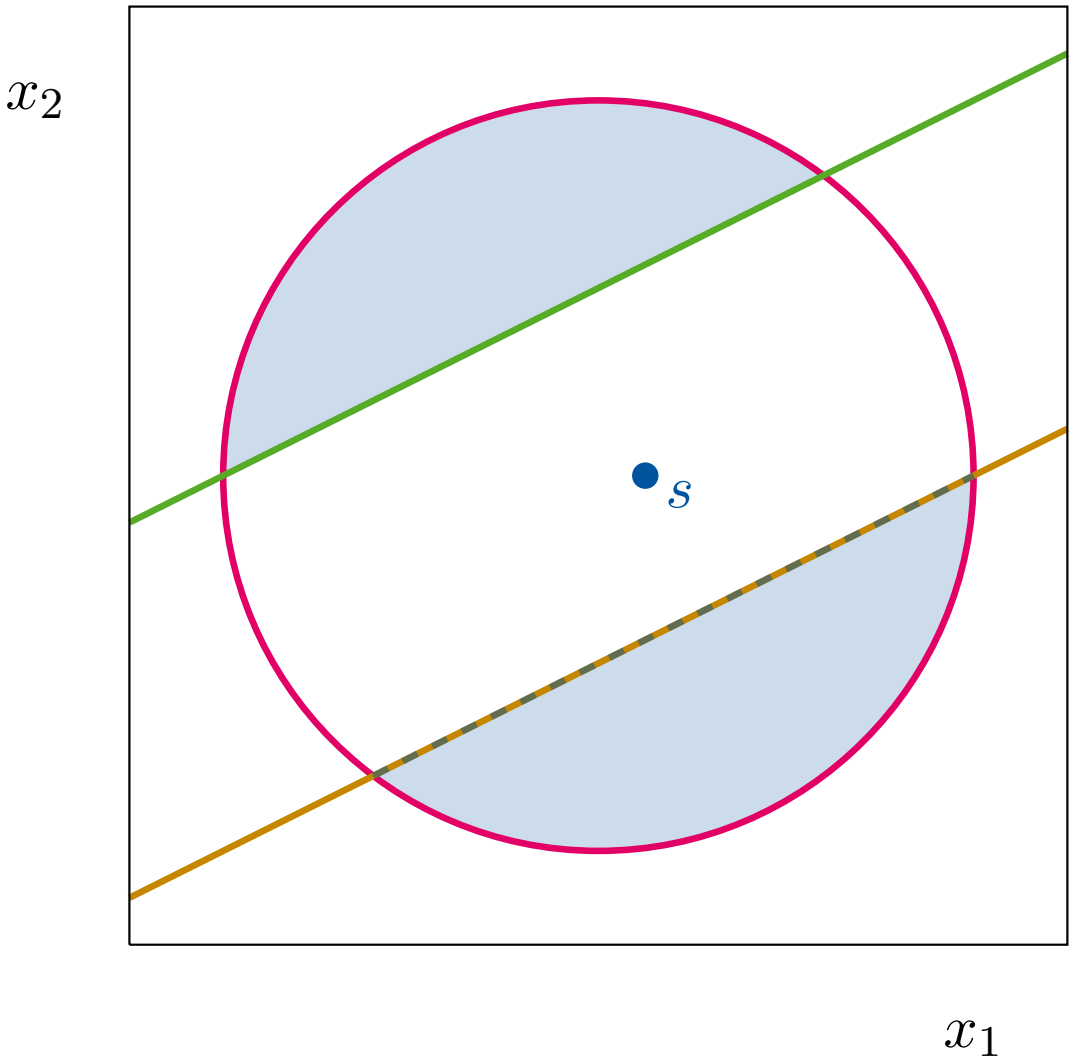
Intuition of sample-based algorithms



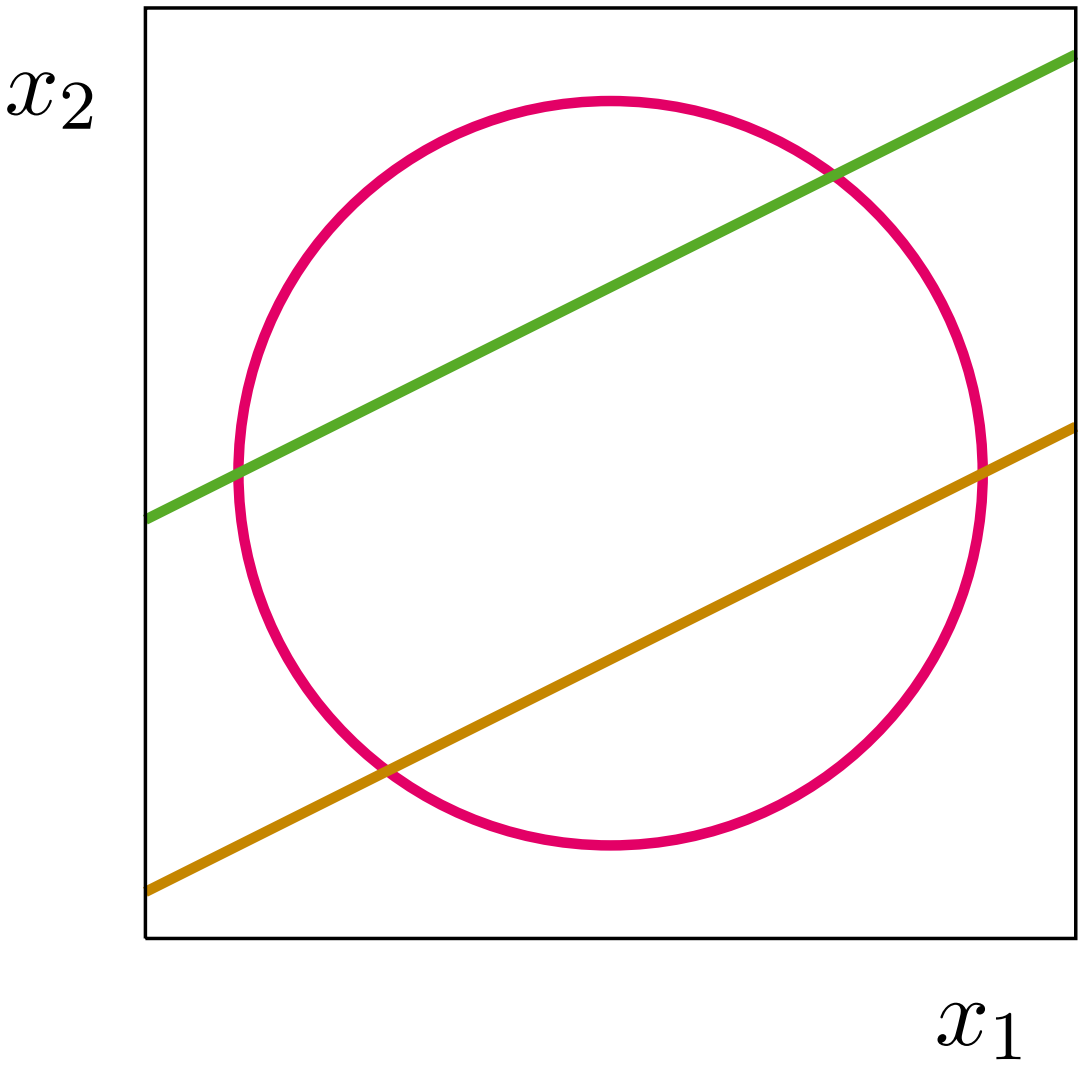
Intuition of sample-based algorithms



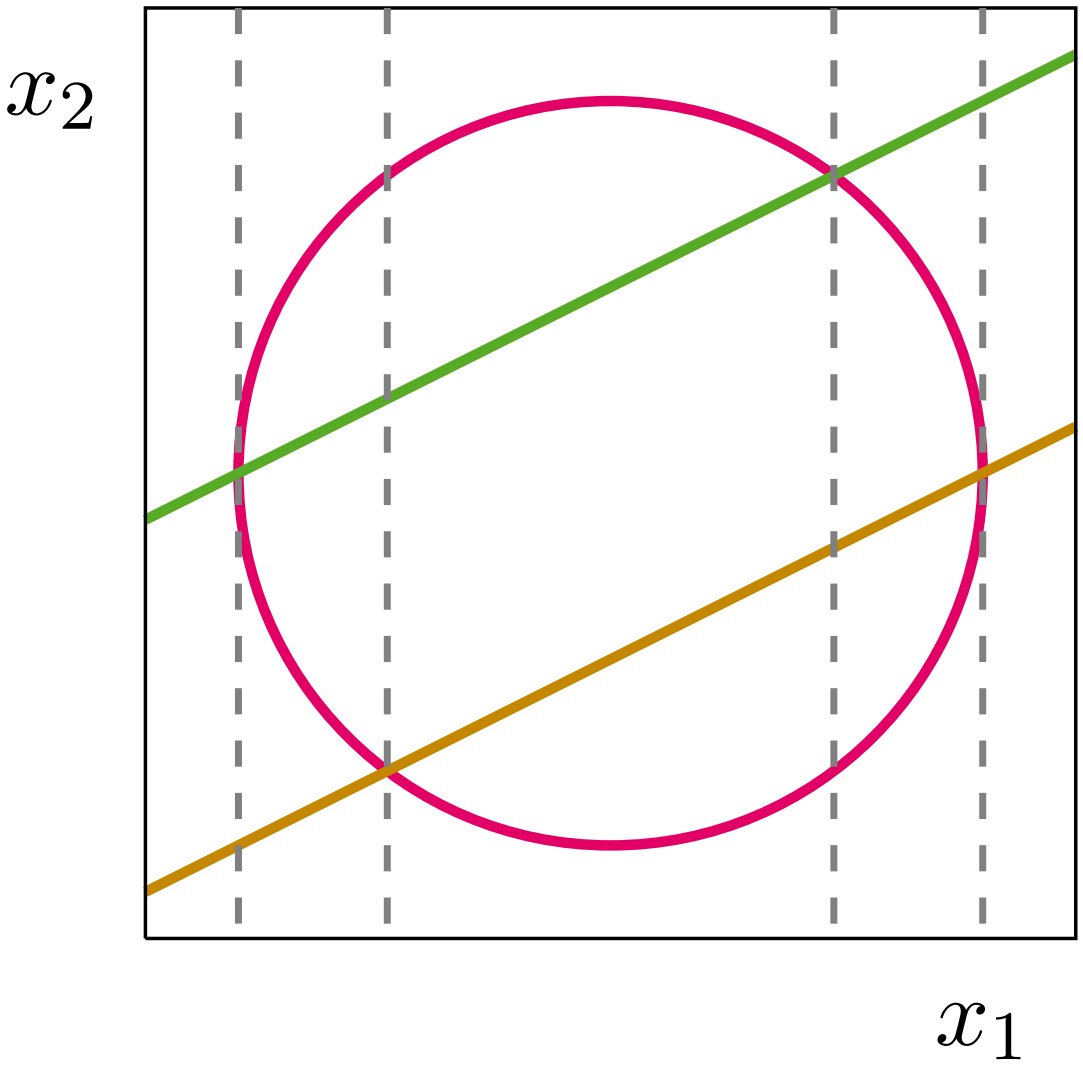
Intuition of sample-based algorithms



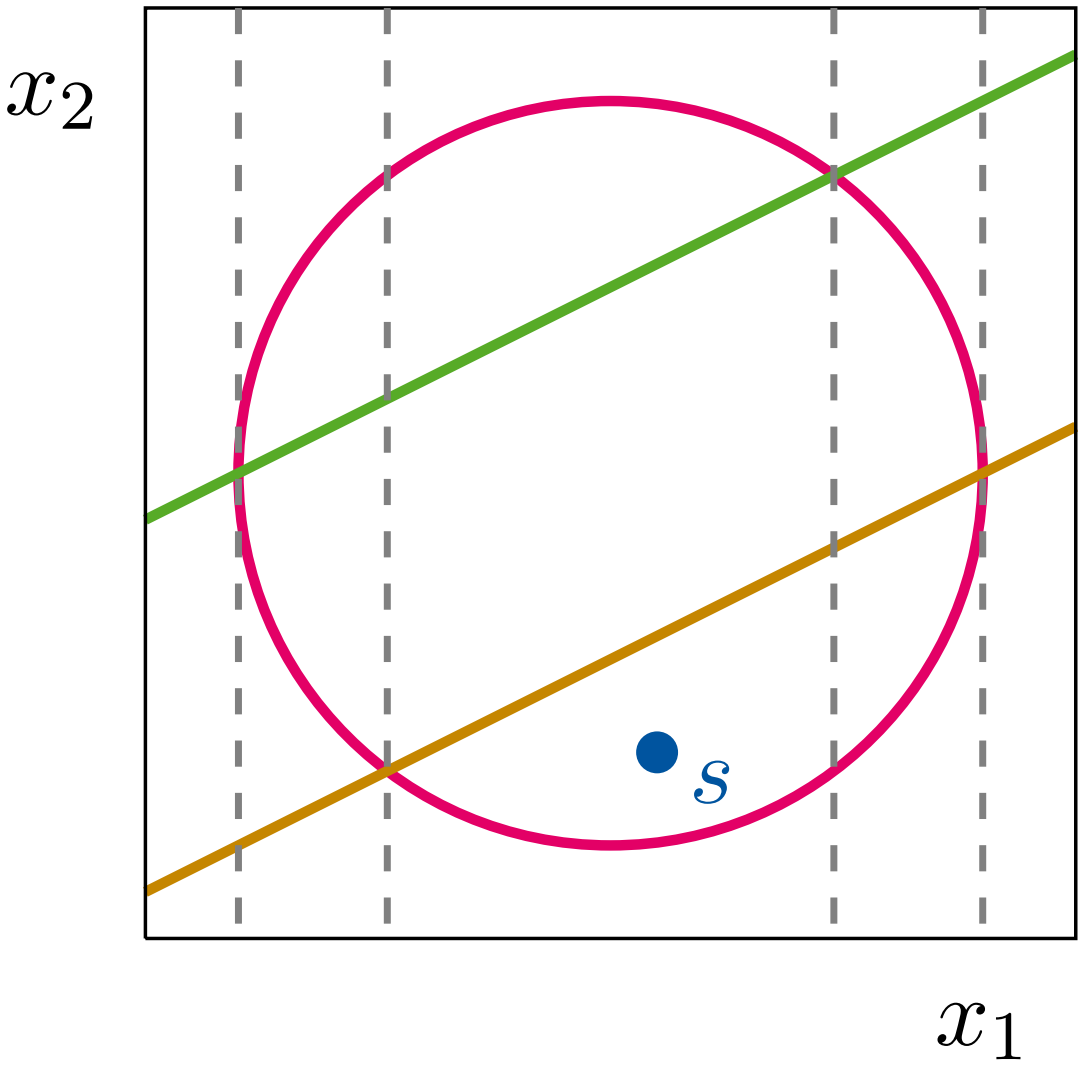
From CAD to single cells



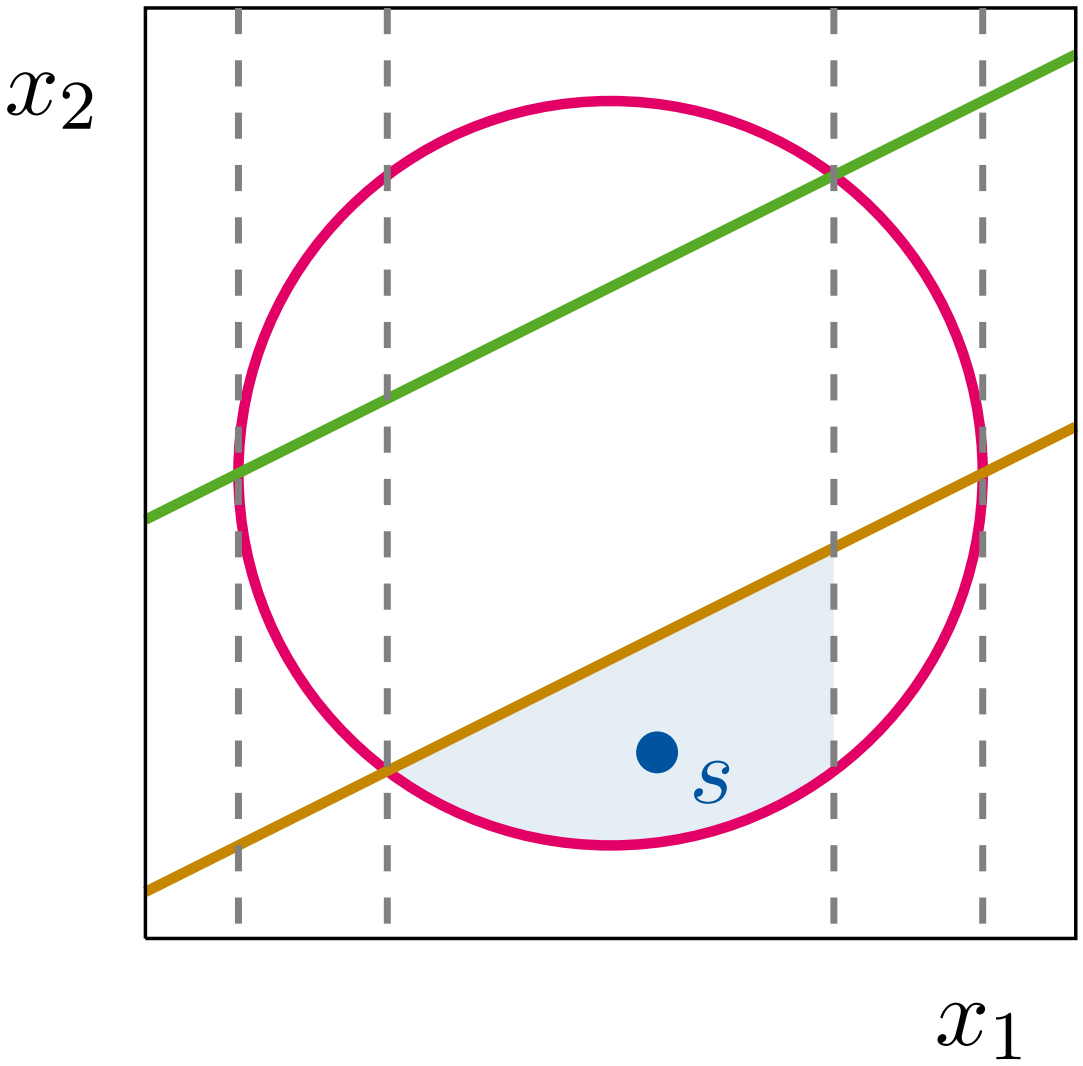
From CAD to single cells



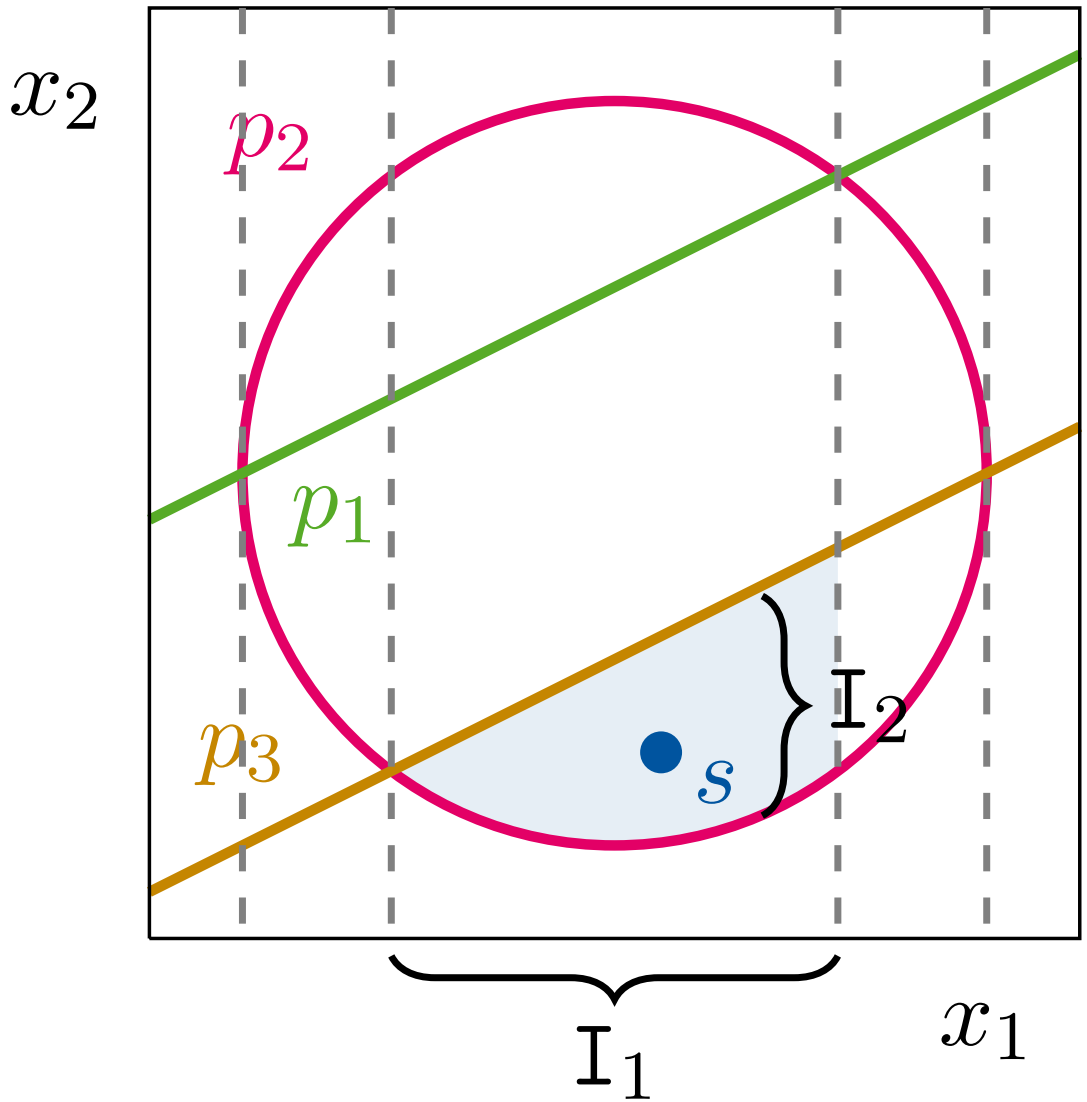
From CAD to single cells



From CAD to single cells



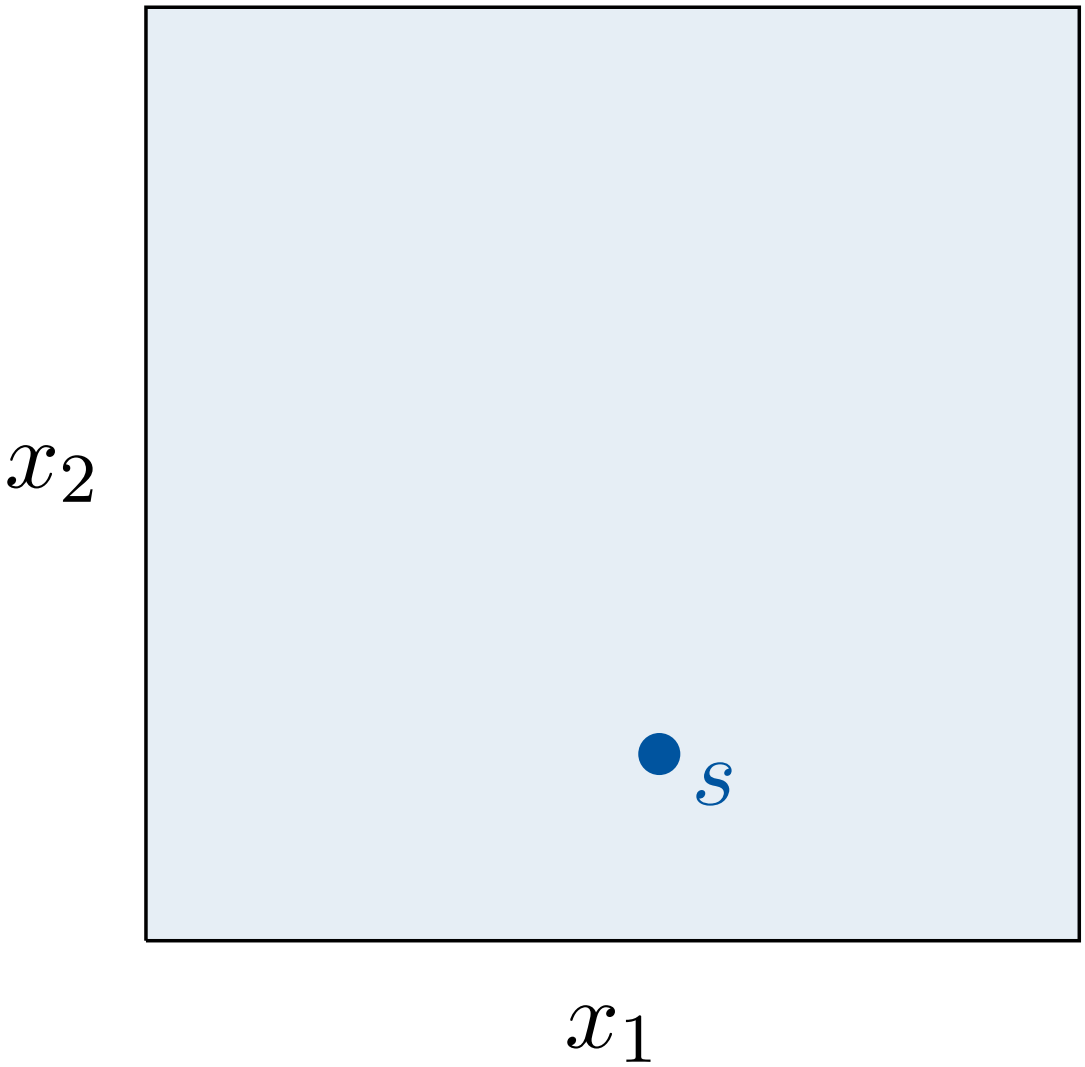
From CAD to single cells



$$\mathtt{I}_2 = (\mathrm{root}_{x_2}[p_2, 1], \mathrm{root}_{x_2}[p_3, 1])$$

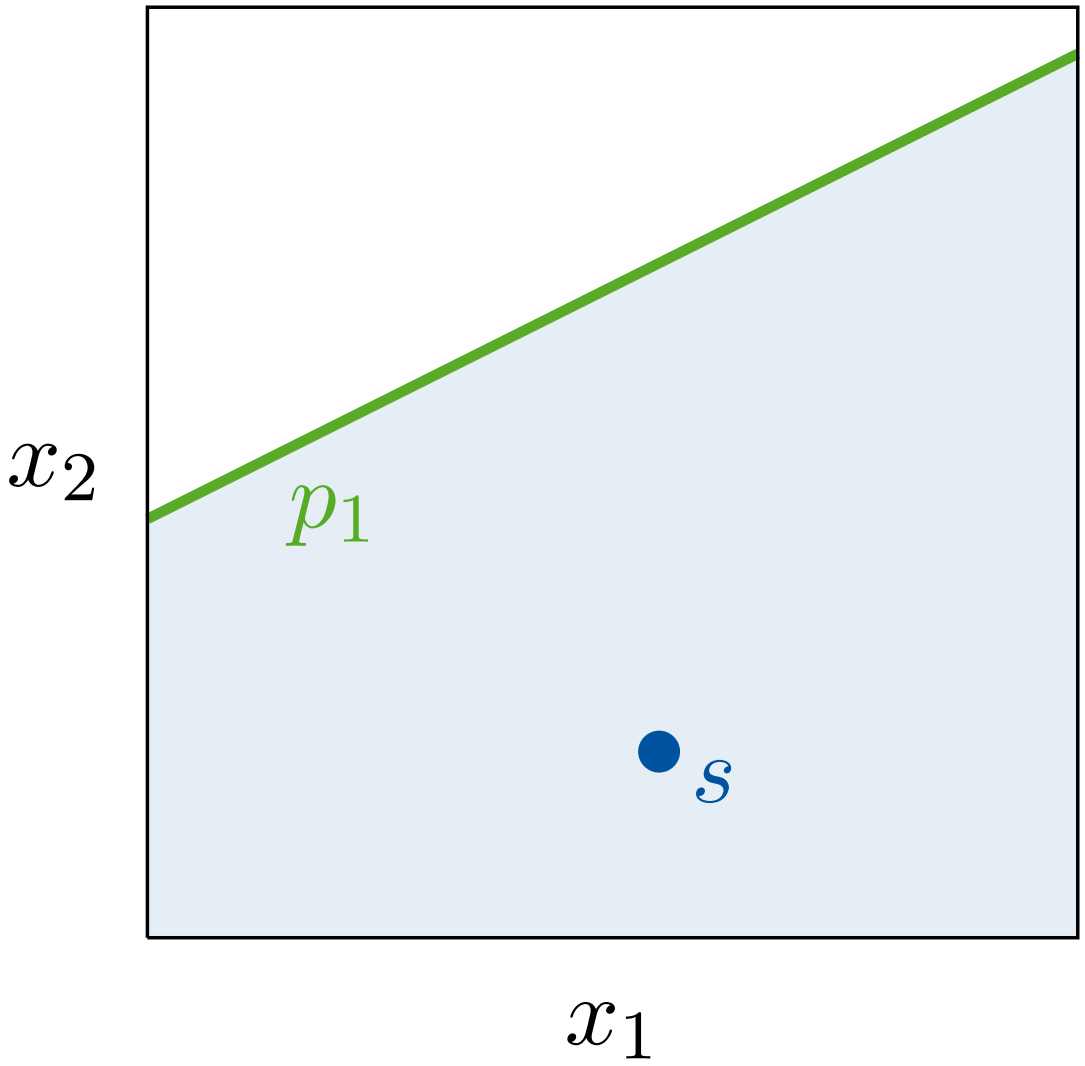
$$\mathtt{I}_1 = (\mathrm{root}_{x_2}[\mathrm{res}_{x_2}(p_3, p_2), 1], \mathrm{root}_{x_2}[\mathrm{res}_{x_2}(p_2, p_3), 2])$$

Refinement-based single cell [Brown, Košta 2015]



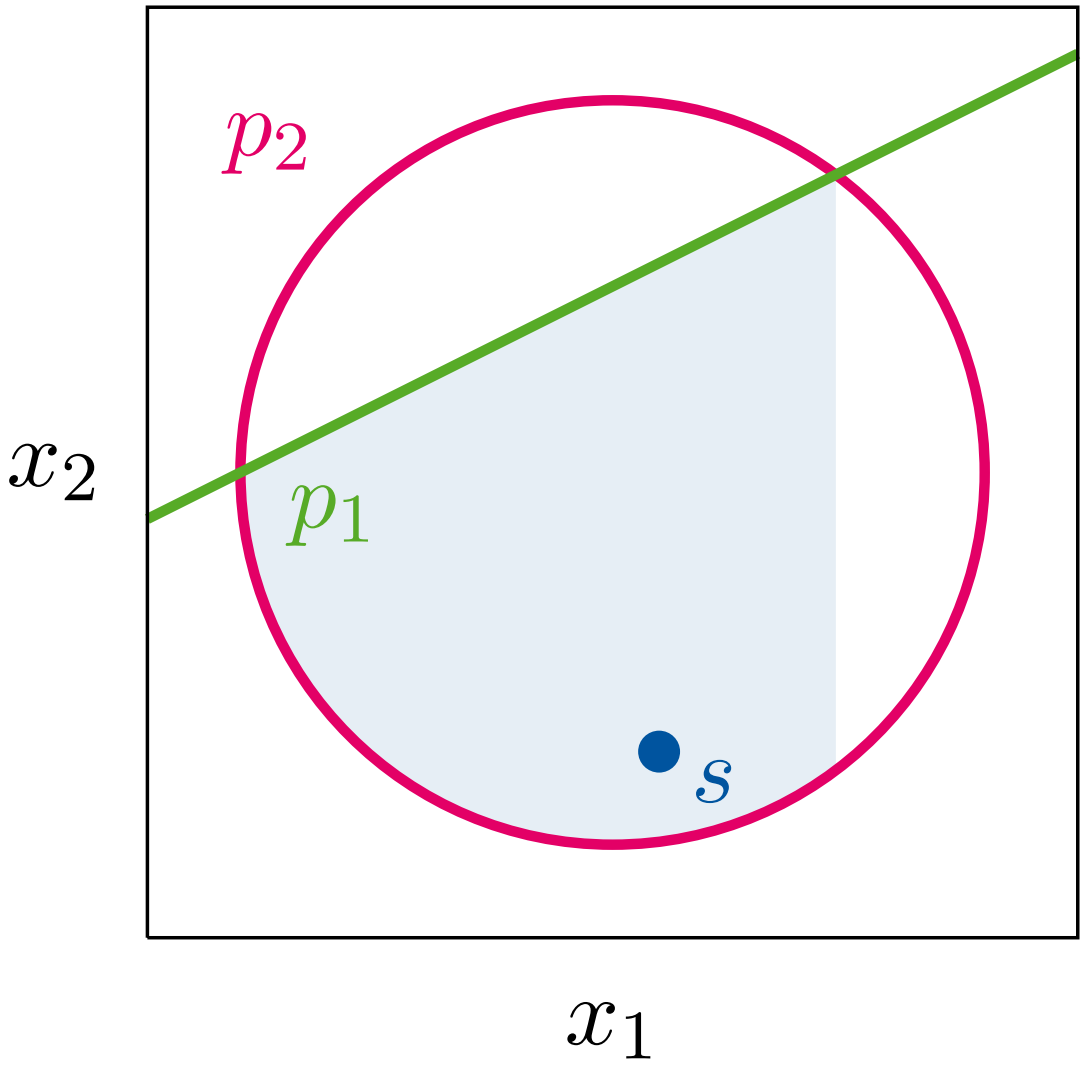
$$\mathtt{I}_2 = (-\infty, \infty)$$
$$\mathtt{I}_1 = (-\infty, \infty)$$

Refinement-based single cell [Brown, Košta 2015]



$$\mathtt{I}_2 = (-\infty, \mathrm{root}_{x_2}[p_1, 1])$$
$$\mathtt{I}_1 = (-\infty, \infty)$$

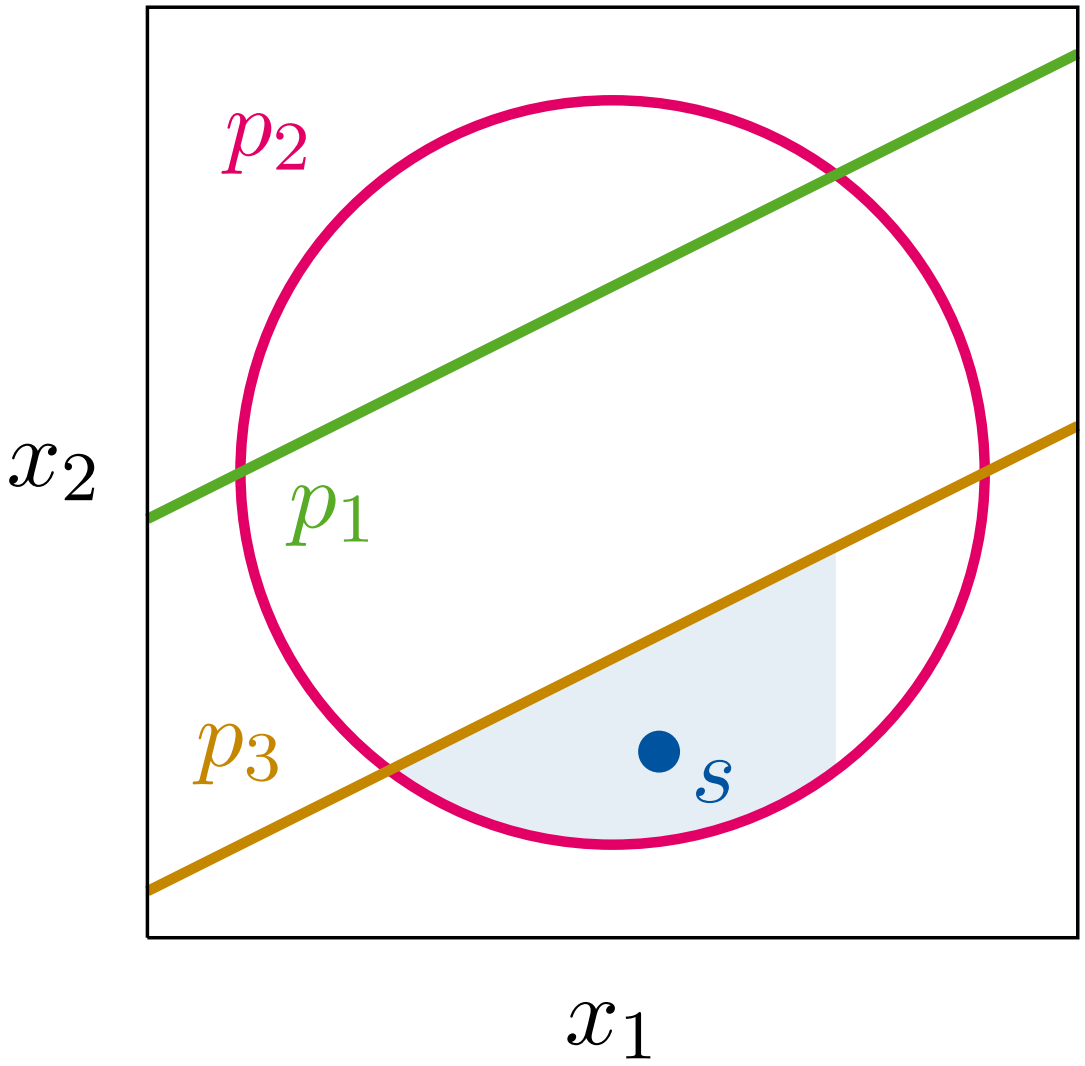
Refinement-based single cell [Brown, Košta 2015]



$$\mathtt{I}_2 = (\mathrm{root}_{x_2}[p_2, 1], \mathrm{root}_{x_2}[p_1, 1])$$

$$\mathtt{I}_1 = (\mathrm{root}_{x_2}[\mathrm{res}_{x_2}(p_1, p_2), 1], \mathrm{root}_{x_2}[\mathrm{res}_{x_2}(p_1, p_2), 2])$$

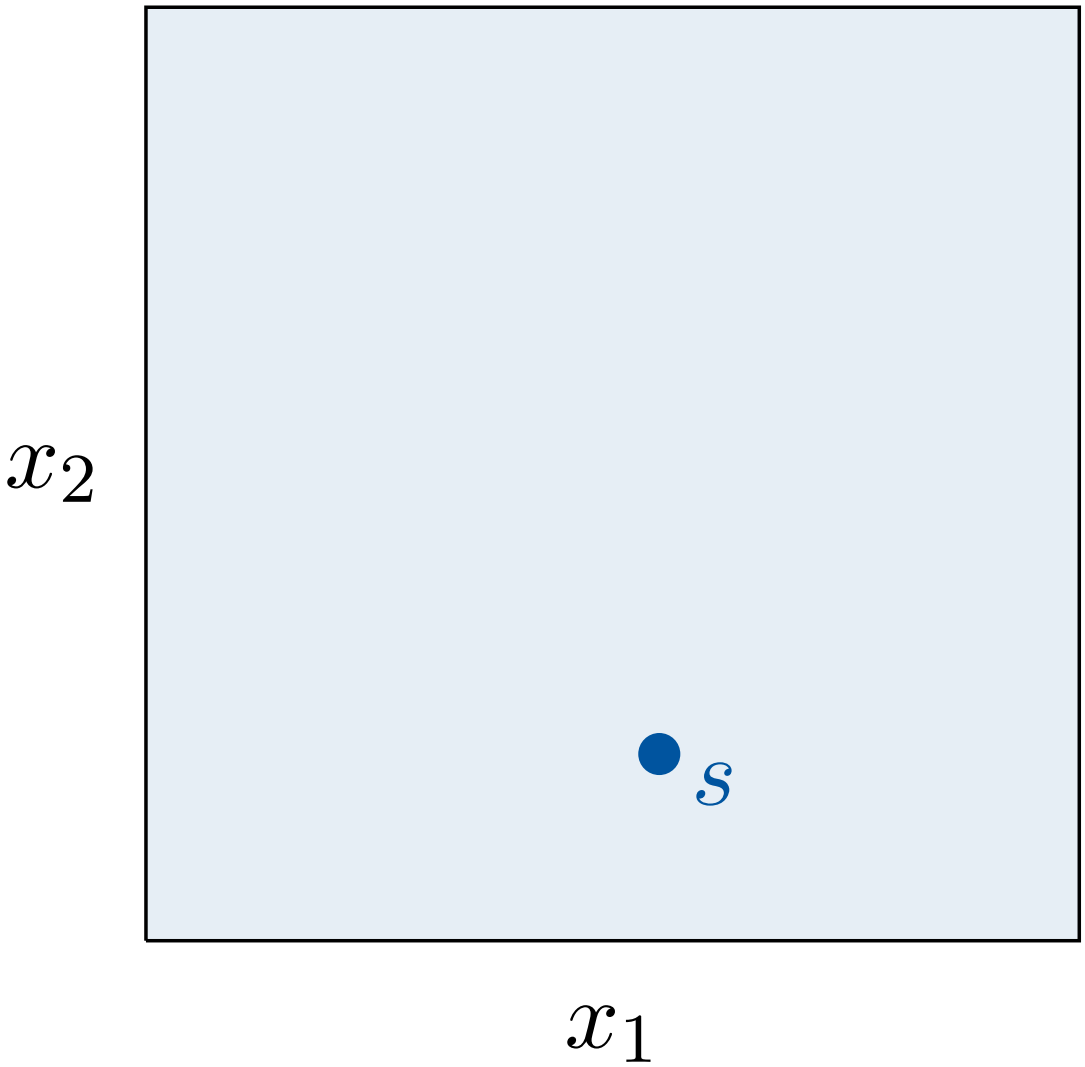
Refinement-based single cell [Brown, Košta 2015]



$$\mathtt{I}_2 = (\mathrm{root}_{x_2}[p_2, 1], \mathrm{root}_{x_2}[p_3, 1])$$

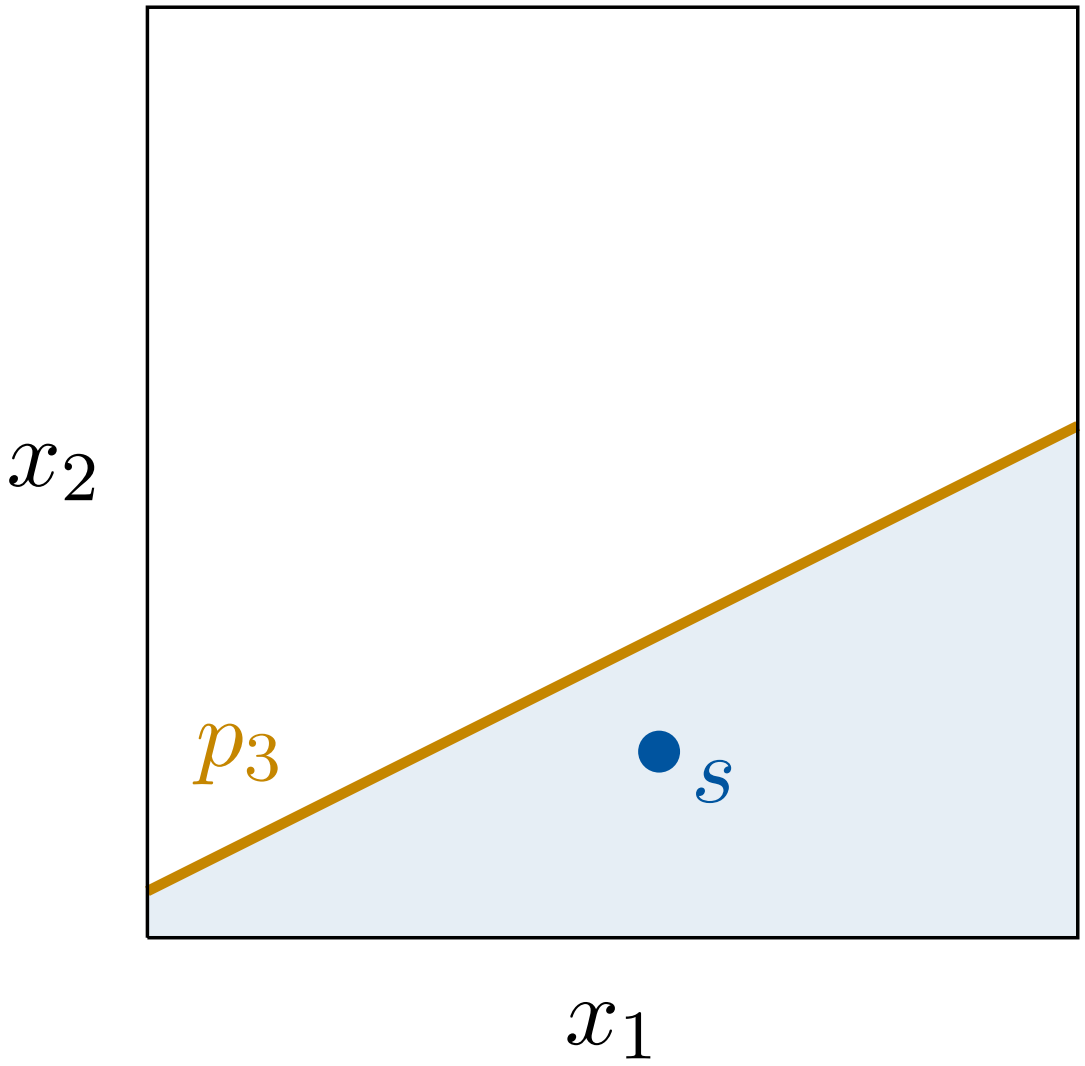
$$\mathtt{I}_1 = (\mathrm{root}_{x_2}[\mathrm{res}_{x_2}(p_3, p_2), 1], \mathrm{root}_{x_2}[\mathrm{res}_{x_2}(p_1, p_2), 2])$$

Refinement-based single cell [Brown, Košta 2015]



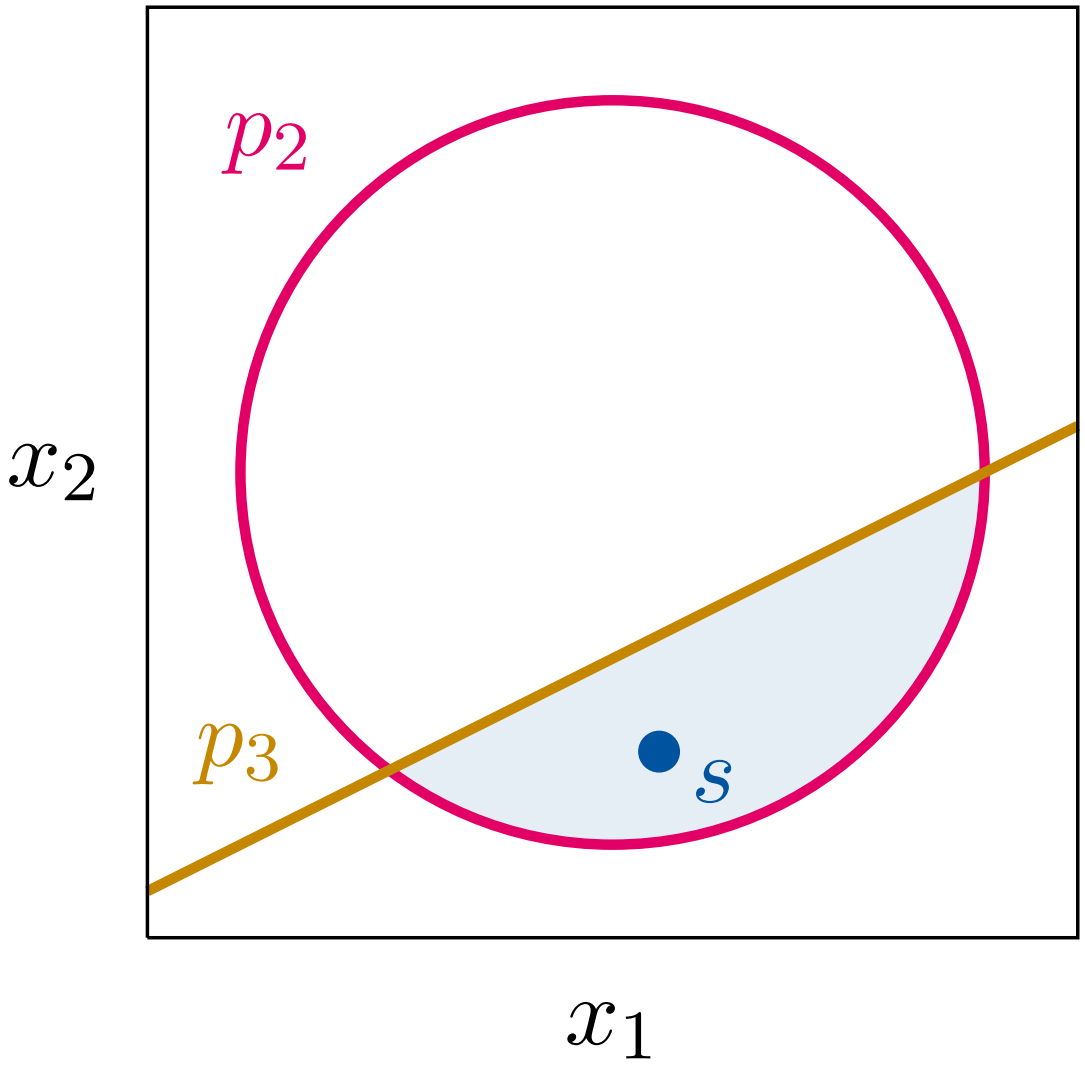
$$\mathtt{I}_2 = (-\infty, \infty)$$
$$\mathtt{I}_1 = (-\infty, \infty)$$

Refinement-based single cell [Brown, Košta 2015]



$$\mathtt{I}_2 = (-\infty, \mathrm{root}_{x_2}[p_3, 1])$$
$$\mathtt{I}_1 = (-\infty, \infty)$$

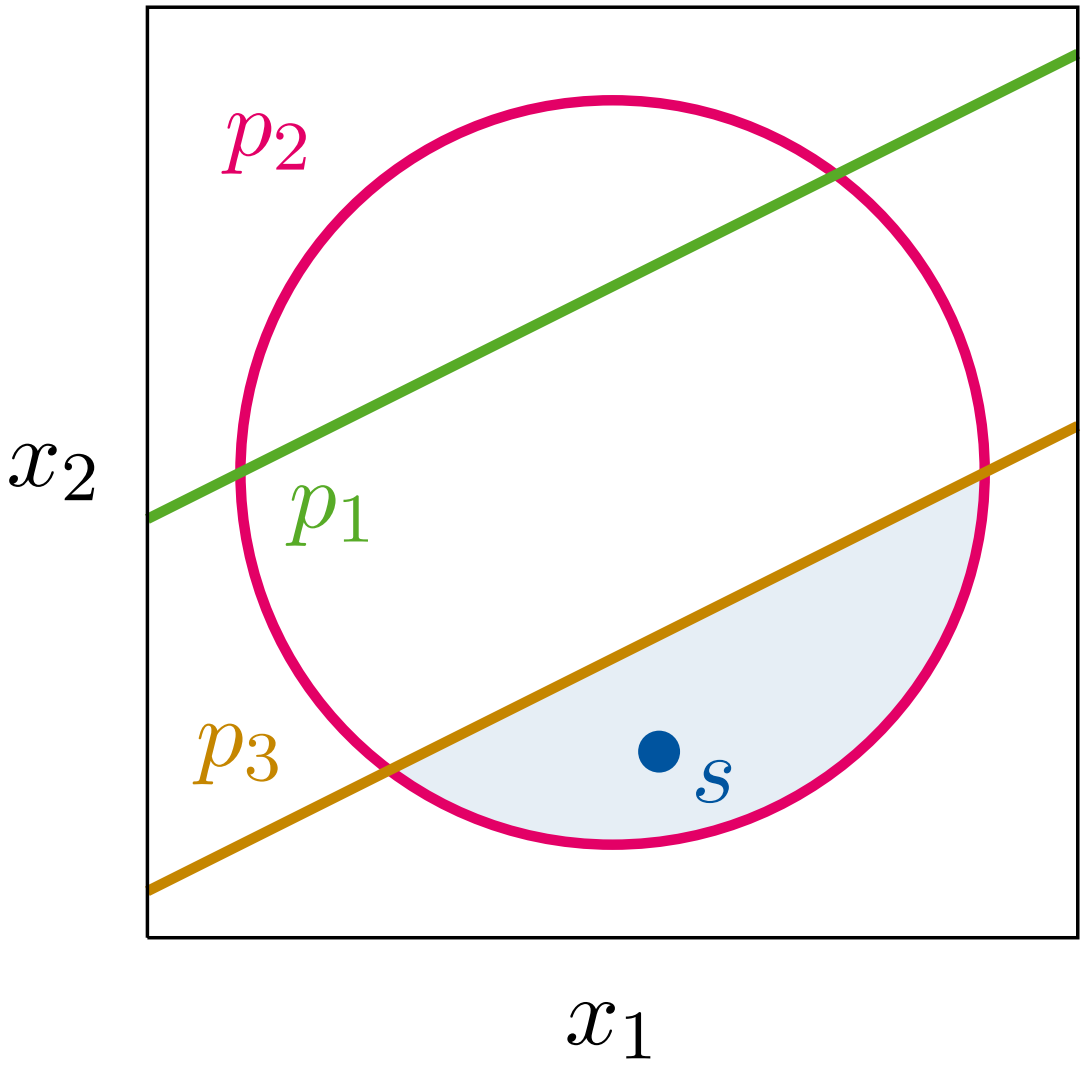
Refinement-based single cell [Brown, Košta 2015]



$$\mathtt{I}_2 = (\mathrm{root}_{x_2}[p_2, 1], \mathrm{root}_{x_2}[p_3, 1])$$

$$\mathtt{I}_1 = (\mathrm{root}_{x_2}[\mathrm{res}_{x_2}(p_3, p_2), 1], \mathrm{root}_{x_2}[\mathrm{res}_{x_2}(p_2, p_3), 2])$$

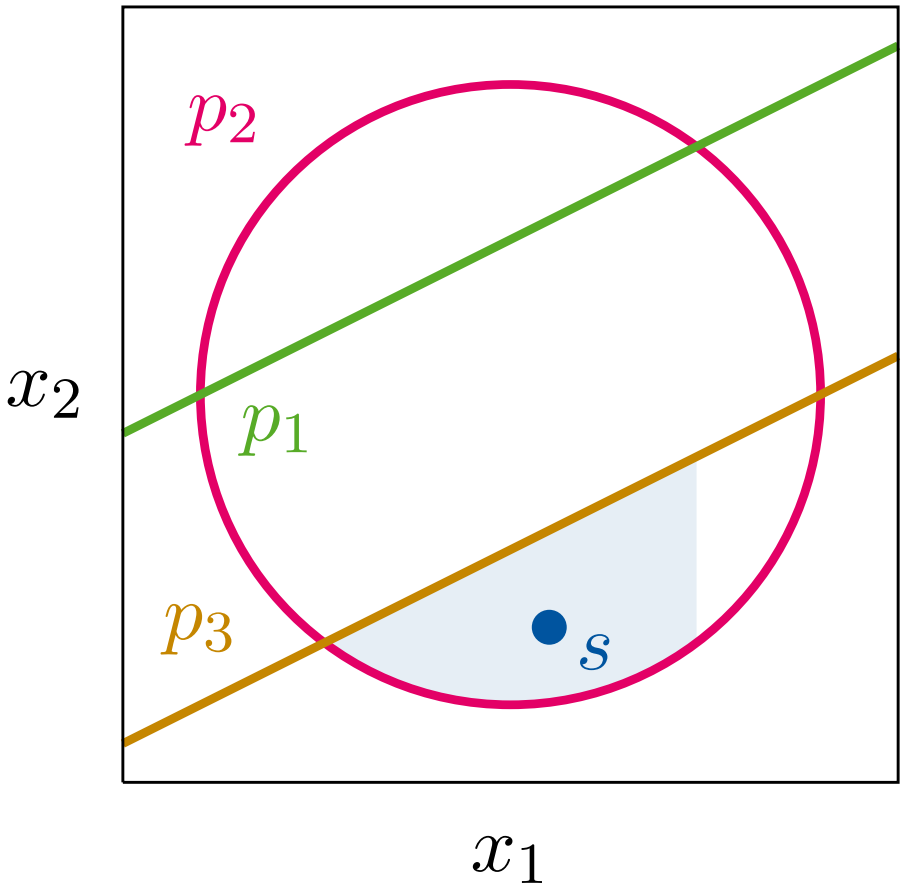
Refinement-based single cell [Brown, Košta 2015]



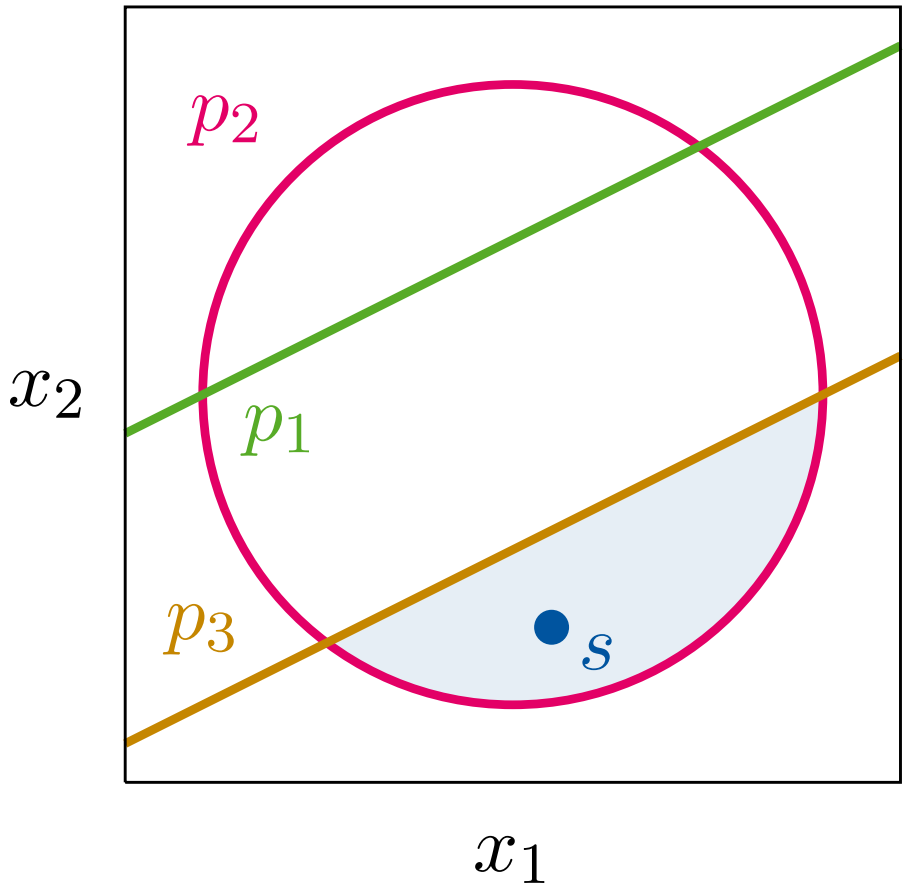
$$\mathtt{I}_2 = (\mathrm{root}_{x_2}[p_2, 1], \mathrm{root}_{x_2}[p_3, 1])$$

$$\mathtt{I}_1 = (\mathrm{root}_{x_2}[\mathrm{res}_{x_2}(p_3, p_2), 1], \mathrm{root}_{x_2}[\mathrm{res}_{x_2}(p_2, p_3), 2])$$

Refinement-based single cell [Brown, Košta 2015]

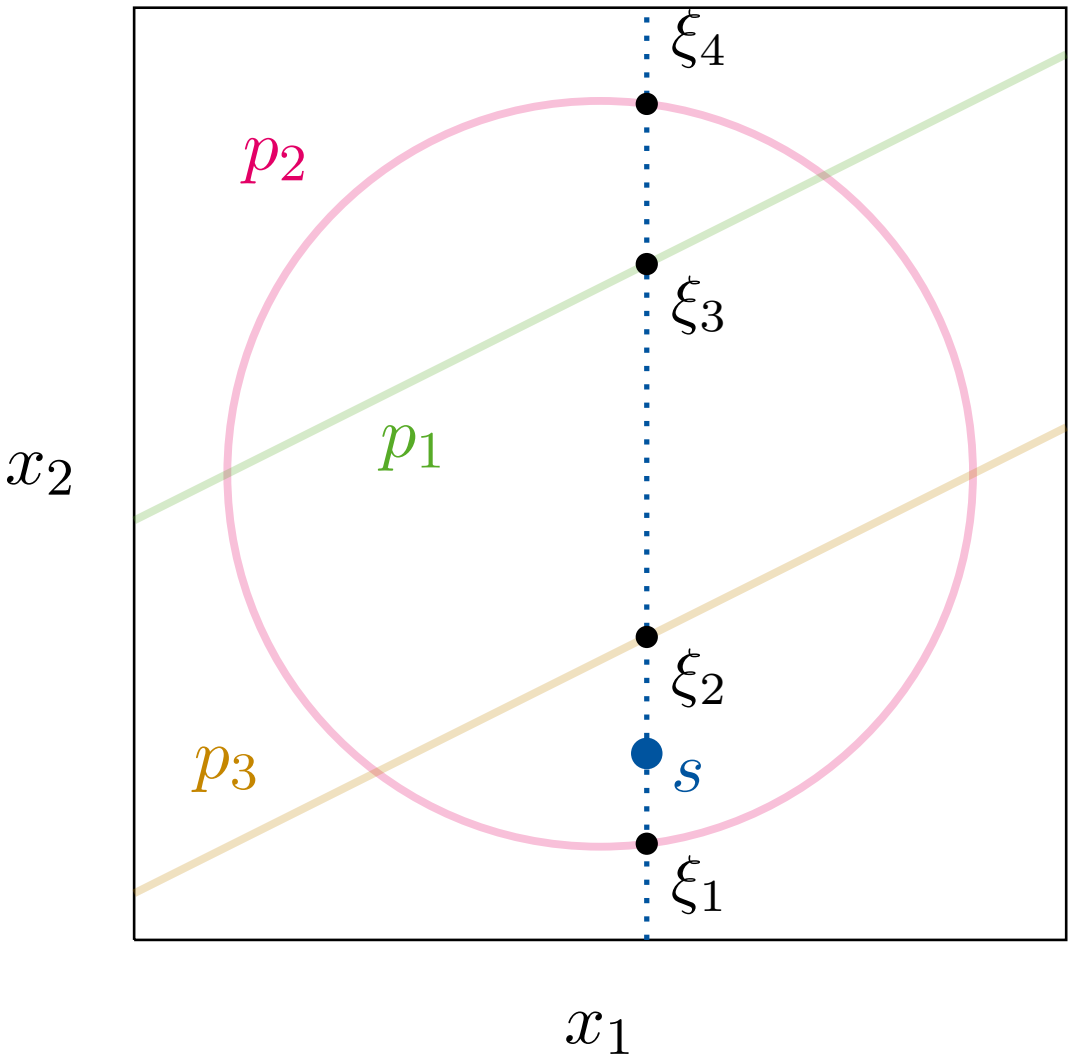


order: p_1, p_2, p_3

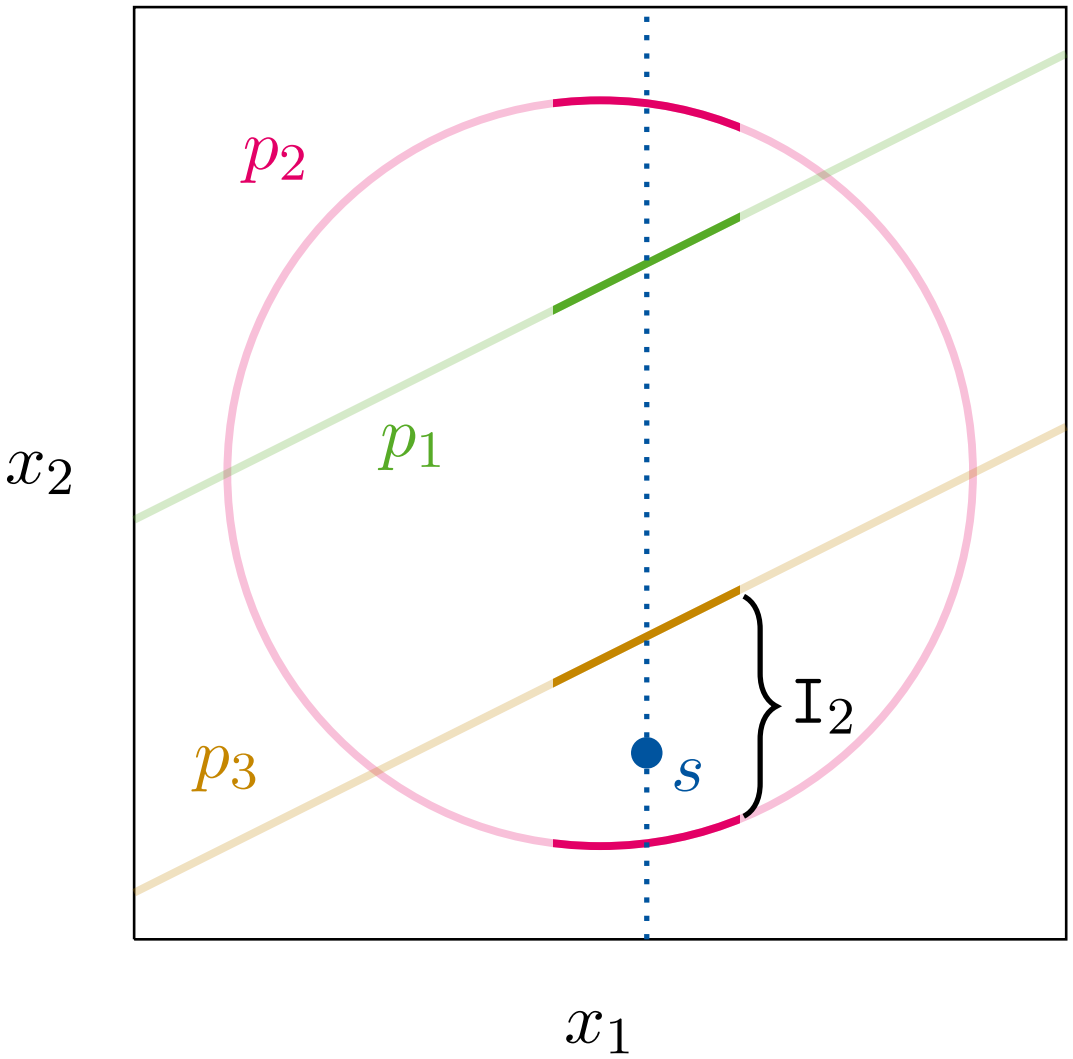


order: p_3, p_2, p_1

Levelwise single cell construction

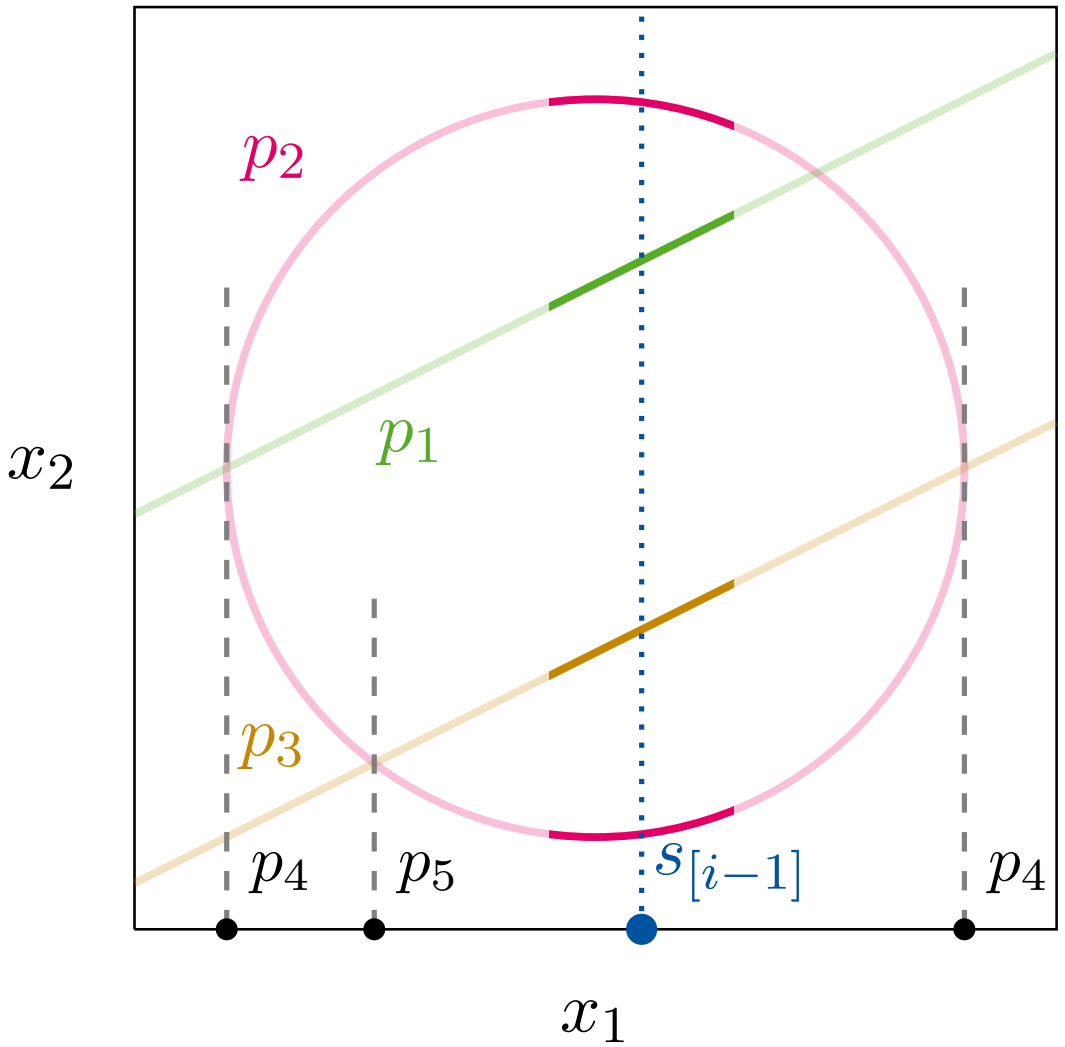


Levelwise single cell construction



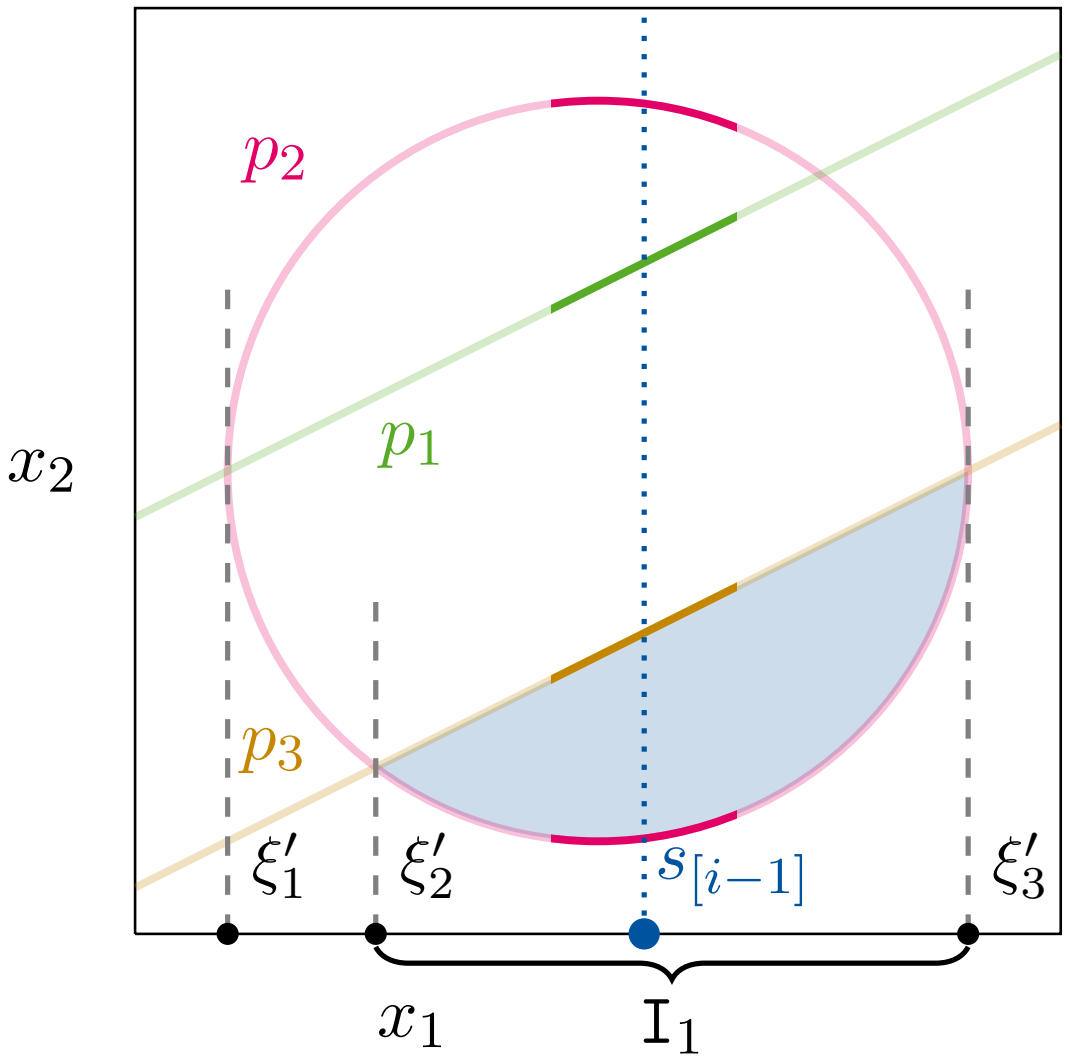
$$\mathtt{I}_2 = (\mathrm{root}_{x_2}[p_2, 1], \mathrm{root}_{x_2}[p_3, 1])$$

Levelwise single cell construction



$$\mathtt{I}_2 = (\mathrm{root}_{x_2}[p_2, 1], \mathrm{root}_{x_2}[p_3, 1])$$

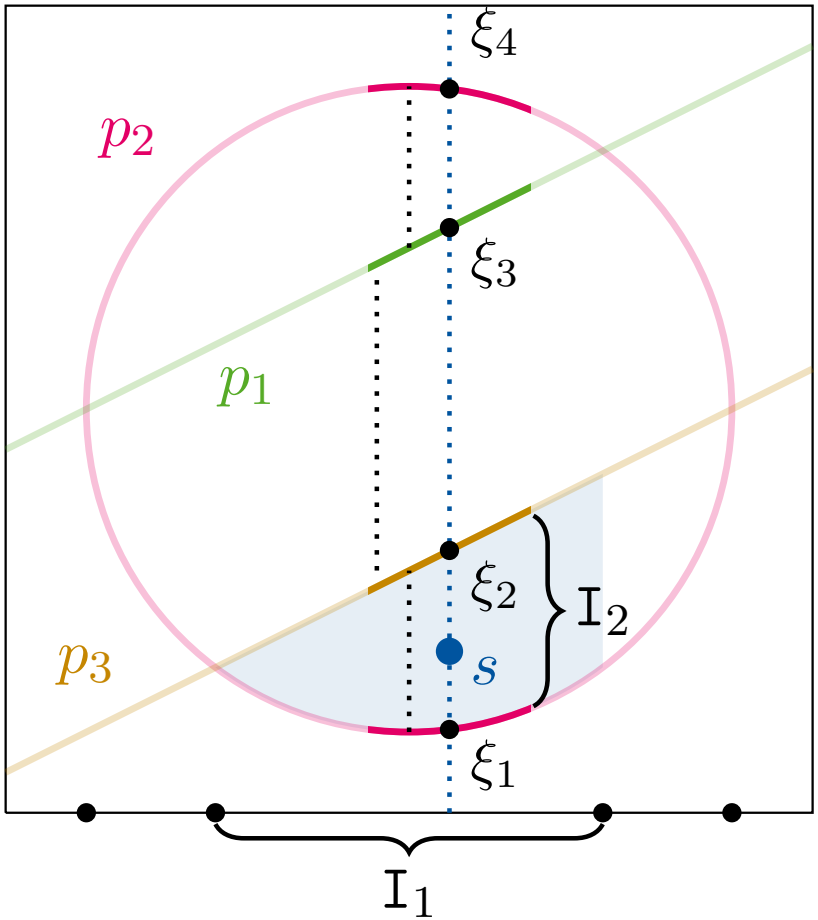
Levelwise single cell construction



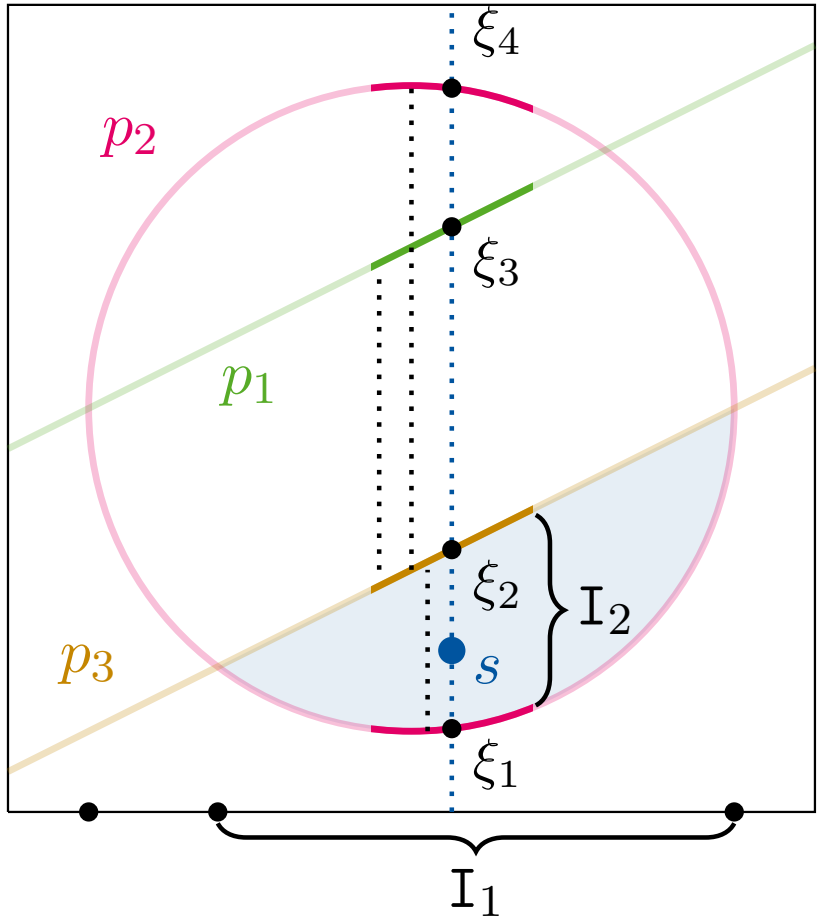
$$\mathrm{I}_2 = (\mathrm{root}_{x_2}[p_2, 1], \mathrm{root}_{x_2}[p_3, 1])$$

$$\mathrm{I}_1 = (\mathrm{root}_{x_2}[\mathrm{res}_{x_2}(p_3, p_2), 1], \mathrm{root}_{x_2}[\mathrm{res}_{x_2}(p_2, p_3), 2])$$

Levelwise single cell construction



$$\xi_1 \preceq \xi_2 \preceq \xi_3 \preceq \xi_4$$



$$\xi_1 \preceq \xi_2 \begin{matrix} \preceq \xi_3 \\ \preceq \xi_4 \end{matrix}$$

A proof system

$$\boxed{\mathit{sgn\_inv}(p_1)} \rightarrow \mathit{sample}(\boxed{s}), \mathit{repr}(\boxed{\mathtt{I}_2}, s_1), \mathit{ir\_ord}(\boxed{\preceq}, s_1), \mathit{an\_del}(p_1), \mathit{an\_sub}(1), \mathit{connected}(1)$$

$$\boxed{\mathit{sgn\_inv}(p_2)} \rightarrow \mathit{sample}(\boxed{s}), \mathit{repr}(\boxed{\mathtt{I}_2}, s_1), \mathit{ir\_ord}(\boxed{\preceq}, s_1), \mathit{an\_del}(p_2), \mathit{an\_sub}(1), \mathit{connected}(1)$$

$$\boxed{\mathit{sgn\_inv}(p_3)} \rightarrow \mathit{sample}(\boxed{s}), \mathit{repr}(\boxed{\mathtt{I}_2}, s_1), \mathit{ir\_ord}(\boxed{\preceq}, s_1), \mathit{an\_del}(p_3), \mathit{an\_sub}(1), \mathit{connected}(1)$$

$$\mathit{sample}(\boxed{s}) \rightarrow \mathit{repr}(\boxed{\mathtt{I}_2}, s_1), \mathit{sample}(s_1)$$

$$\mathit{repr}(\boxed{\mathtt{I}_2}, s_1) \rightarrow \boxed{R = \mathit{setOf}(R{\downarrow_{[1]}}), \boxed{\mathtt{I}_2}}, \mathit{an\_del}(p_2), \mathit{an\_del}(p_3), \mathit{sample}(s_1)$$

$$\mathit{ir\_ord}(\boxed{\preceq}, s_1) \rightarrow \mathit{an\_del}(p_1), \mathit{an\_del}(p_2), \mathit{an\_del}(p_3), \mathit{ord\_inv}(\mathrm{res}_{x_2}(p_3, p_1)), \mathit{ord\_inv}(\mathrm{res}_{x_2}(p_3, p_2)), \mathit{an\_sub}(1), \mathit{connected}(1), \mathit{sample}(s_1)$$

$$\mathit{an\_del}(p_1) \rightarrow \mathit{non\_null}(p_1), \mathit{ord\_inv}(\mathrm{disc}_{x_2}(p_1)), \mathit{an\_sub}(1), \mathit{connected}(1), \mathit{sgn\_inv}(\mathrm{ldcf}_{x_2}(p_1))$$

$$\mathit{an\_del}(p_2) \rightarrow \mathit{non\_null}(p_2), \mathit{ord\_inv}(\mathrm{disc}_{x_2}(p_2)), \mathit{an\_sub}(1), \mathit{connected}(1), \mathit{sgn\_inv}(\mathrm{ldcf}_{x_2}(p_2))$$

$$\mathit{an\_del}(p_3) \rightarrow \mathit{non\_null}(p_3), \mathit{ord\_inv}(\mathrm{disc}_{x_2}(p_3)), \mathit{an\_sub}(1), \mathit{connected}(1), \mathit{sgn\_inv}(\mathrm{ldcf}_{x_2}(p_3))$$

$$\mathit{non\_null}(p_1) \rightarrow \text{trivial}$$

$$\mathit{non\_null}(p_2) \rightarrow \text{trivial}$$

$$\mathit{non\_null}(p_3) \rightarrow \text{trivial}$$

$$\mathit{ord\_inv}(\mathrm{disc}_{x_2}(p_1)) \rightarrow \text{trivial}$$

$$\mathit{ord\_inv}(\mathrm{disc}_{x_2}(p_2)) \rightarrow \mathit{sgn\_inv}(p_4), \mathit{sample}(s_1)$$

$$\mathit{ord\_inv}(\mathrm{disc}_{x_2}(p_3)) \rightarrow \text{trivial}$$

$$\mathit{sgn\_inv}(p_4) \rightarrow \mathit{repr}(\boxed{\mathtt{I}_1})$$

$$\mathit{ord\_inv}(\mathrm{res}_{x_2}(p_3, p_1)) \rightarrow \text{trivial}$$

$$\mathit{ord\_inv}(\mathrm{res}_{x_2}(p_3, p_2)) \rightarrow \mathit{sgn\_inv}(p_5), \mathit{sample}(s_1)$$

$$\mathit{sgn\_inv}(p_5) \rightarrow \mathit{repr}(\boxed{\mathtt{I}_1})$$

$$\mathit{sgn\_inv}(\mathrm{ldcf}_{x_2}(p_1)) \rightarrow \text{trivial}$$

$$\mathit{sgn\_inv}(\mathrm{ldcf}_{x_2}(p_2)) \rightarrow \text{trivial}$$

$$\mathit{sgn\_inv}(\mathrm{ldcf}_{x_2}(p_3)) \rightarrow \text{trivial}$$

$$\mathit{an\_sub}(1) \rightarrow \mathit{repr}(\boxed{\mathtt{I}_1})$$

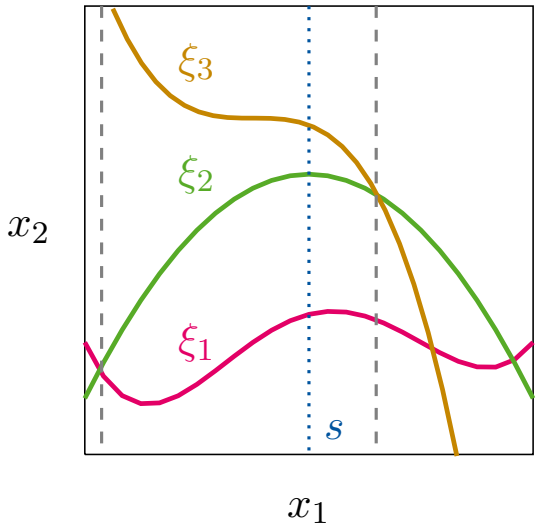
$$\mathit{connected}(1) \rightarrow \text{trivial}$$

$$\mathit{sample}(s_1) \rightarrow \mathit{repr}(\boxed{\mathtt{I}_1})$$

$$\mathit{repr}(\boxed{\mathtt{I}_1}) \rightarrow \boxed{R{\downarrow_{[1]}} = \mathit{setOf}(\boxed{\mathtt{I}_1})}$$

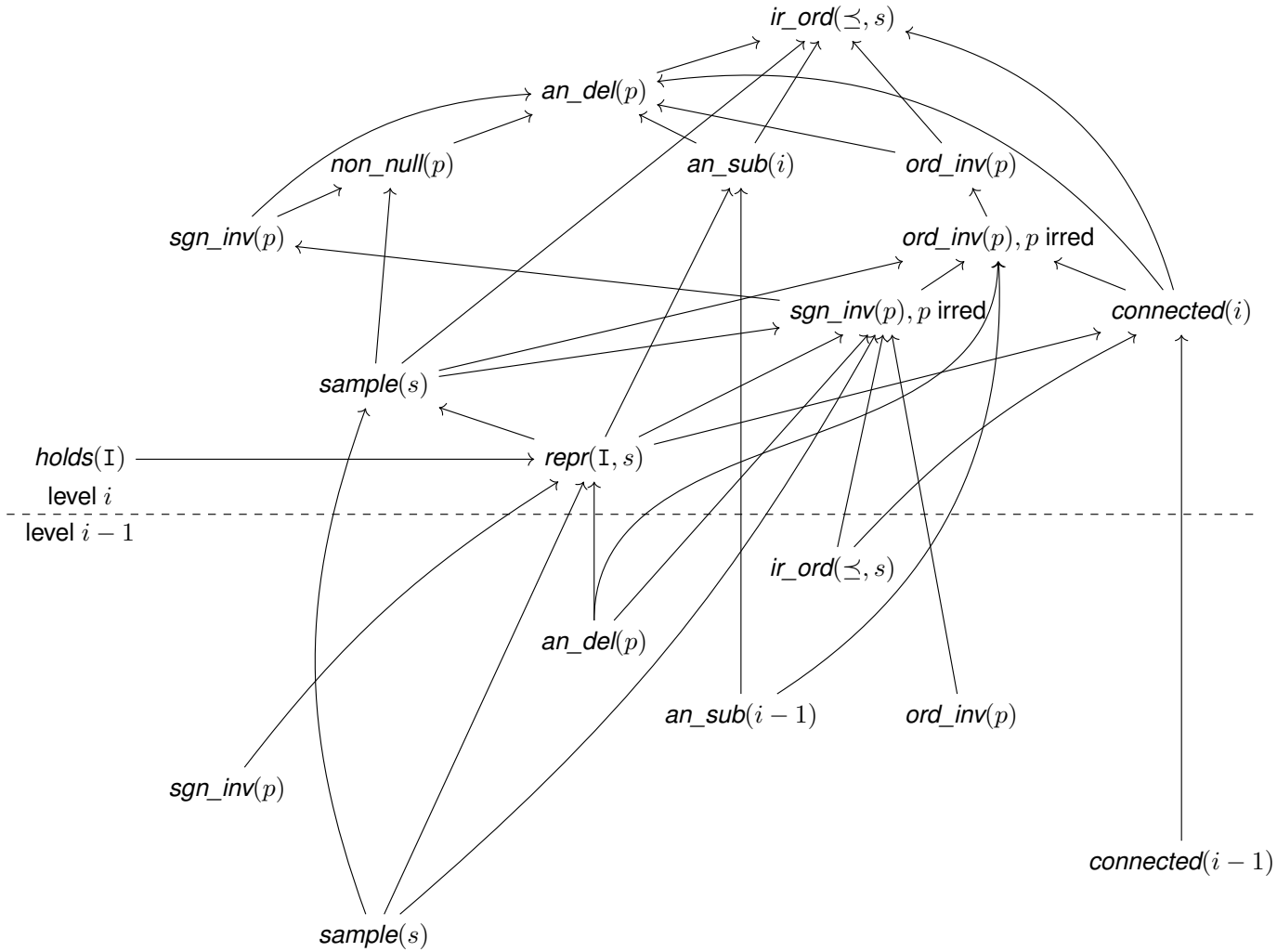
An exemplary proof rule

$i \in \mathbb{N}$, $R \subseteq \mathbb{R}^i$, $s \in \mathbb{R}^i$,
$\preceq$ an indexed root ordering of level $i+1$,
$\xi.p$ is irreducible for all $\xi \in \mathrm{dom}(\preceq)$,
$\preceq$ matches s



$$\begin{aligned} \textit{sample}(s)(R),\ \textit{an\_sub}(i)(R),\ \textit{connected}(i)(R), \\ \forall \xi \in \mathrm{dom}(\preceq).\ \textit{an\_del}(\xi.p)(R), \\ \forall (\xi, \xi') \in \preceq .\ \textit{ord\_inv}(\mathrm{res}_{x_{i+1}}(\xi.p, \xi'.p))(R) \quad \vdash \textit{ir\_ord}(\preceq, s)(R) \end{aligned}$$

A graph of properties



Heuristics

Heuristics

- Application of proof rules

Heuristics

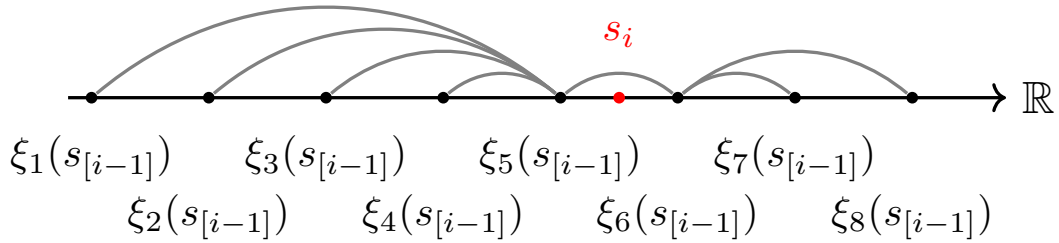
- Application of proof rules
- Choice of representation `I`

Heuristics

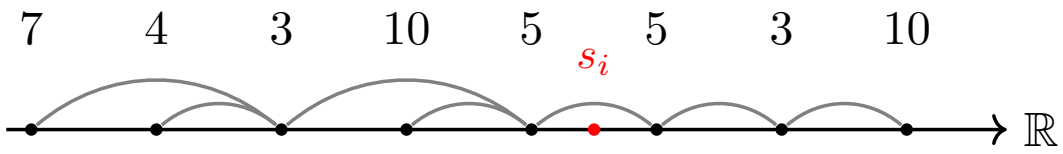
- Application of proof rules
- Choice of representation $\mathtt{I}$
- Choice of indexed root orderings $\preceq$

Heuristics: Indexed root orderings

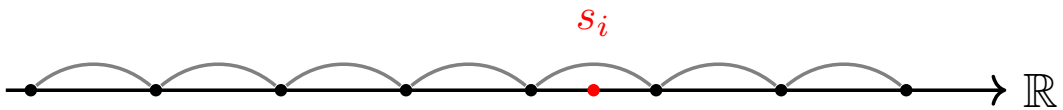
Biggest cell



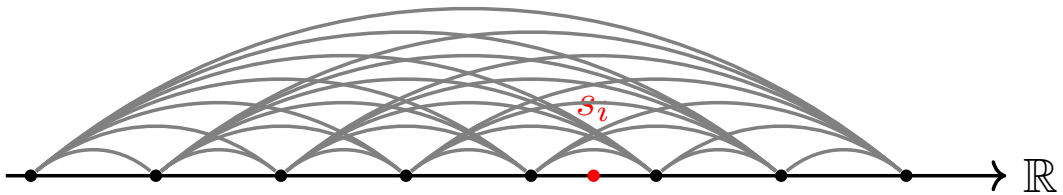
Lowest degree barriers



Chain

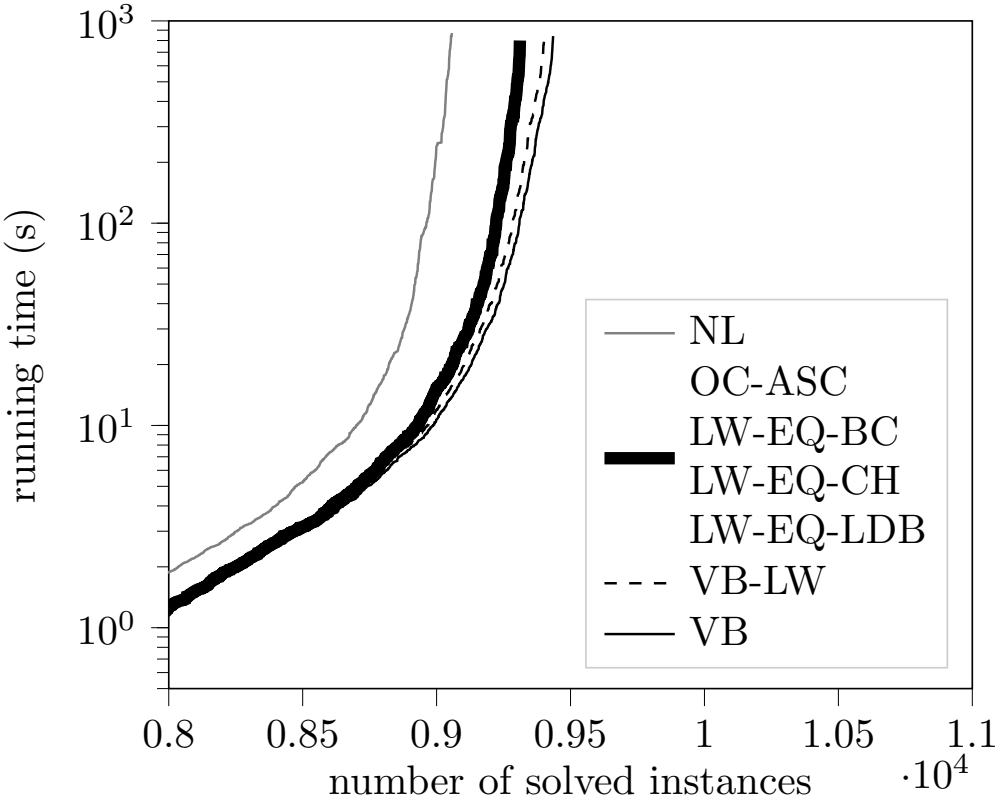


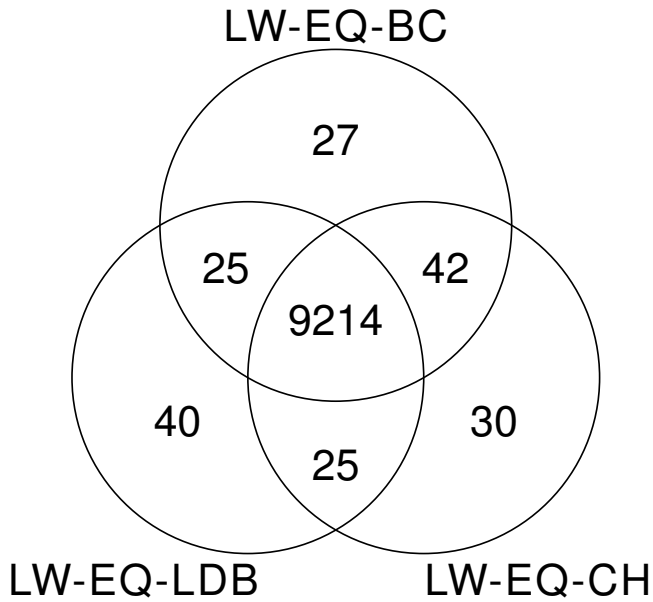
Full



Experimental results

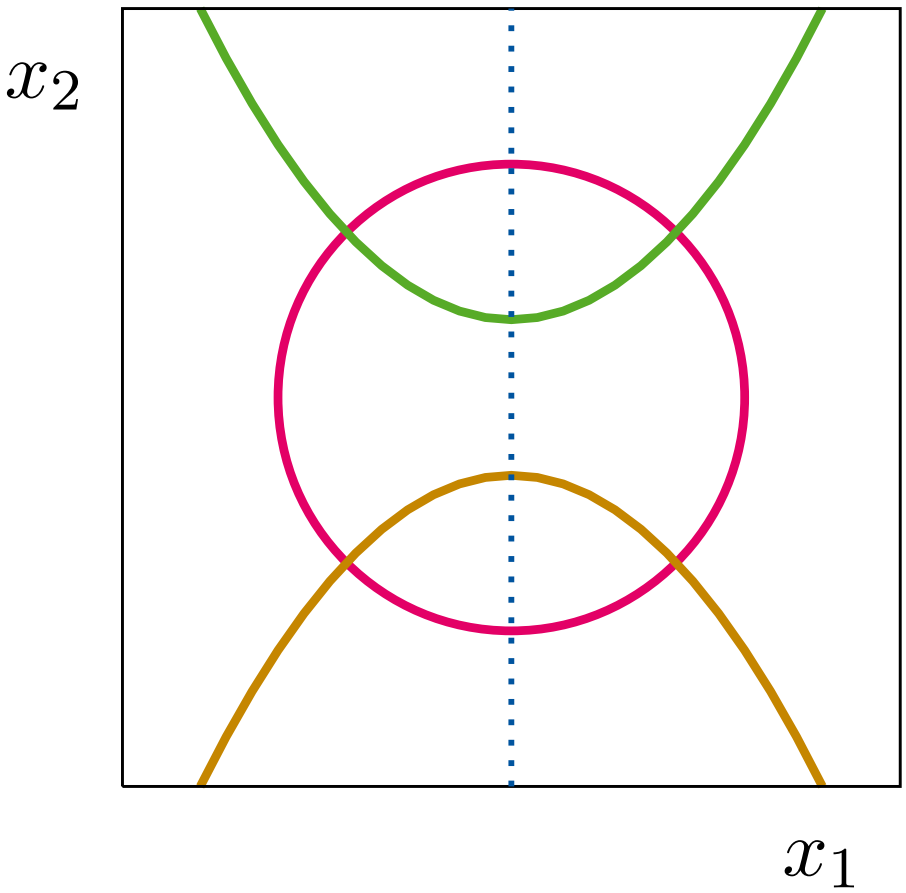
Implementation in SMT-RAT-MCSAT





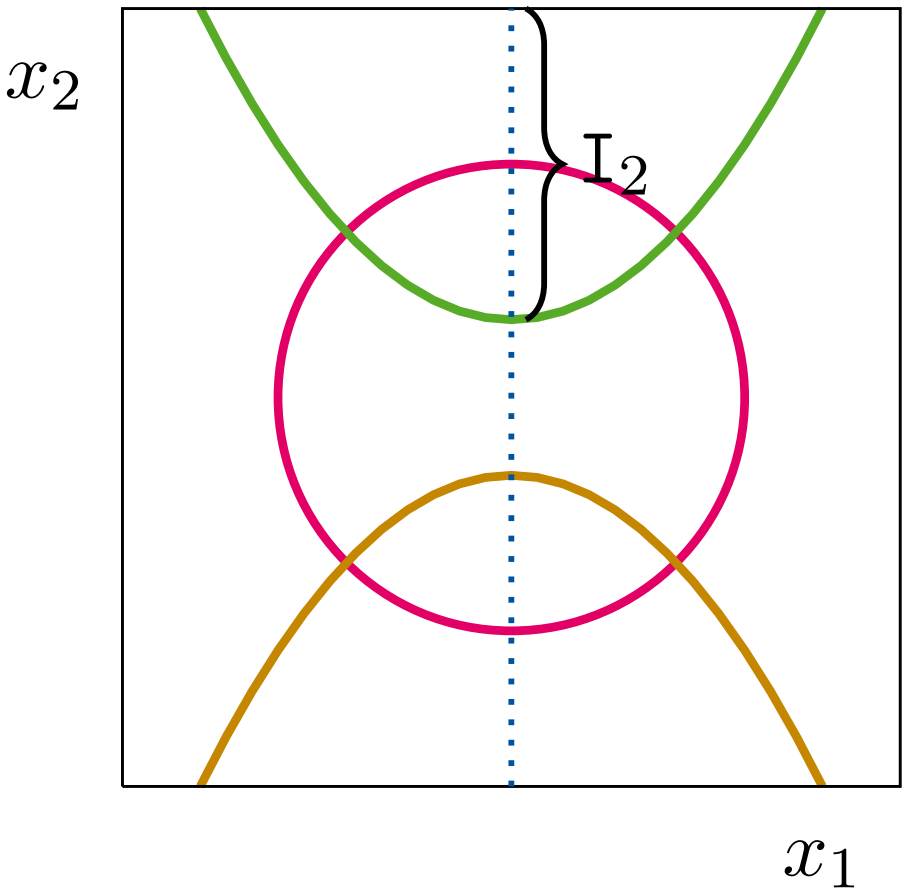
Cylindrical algebraic coverings

$$x^2 - y > -0.5 \wedge x^2 + y^2 > 1 \wedge -x^2 - y < 0.5$$



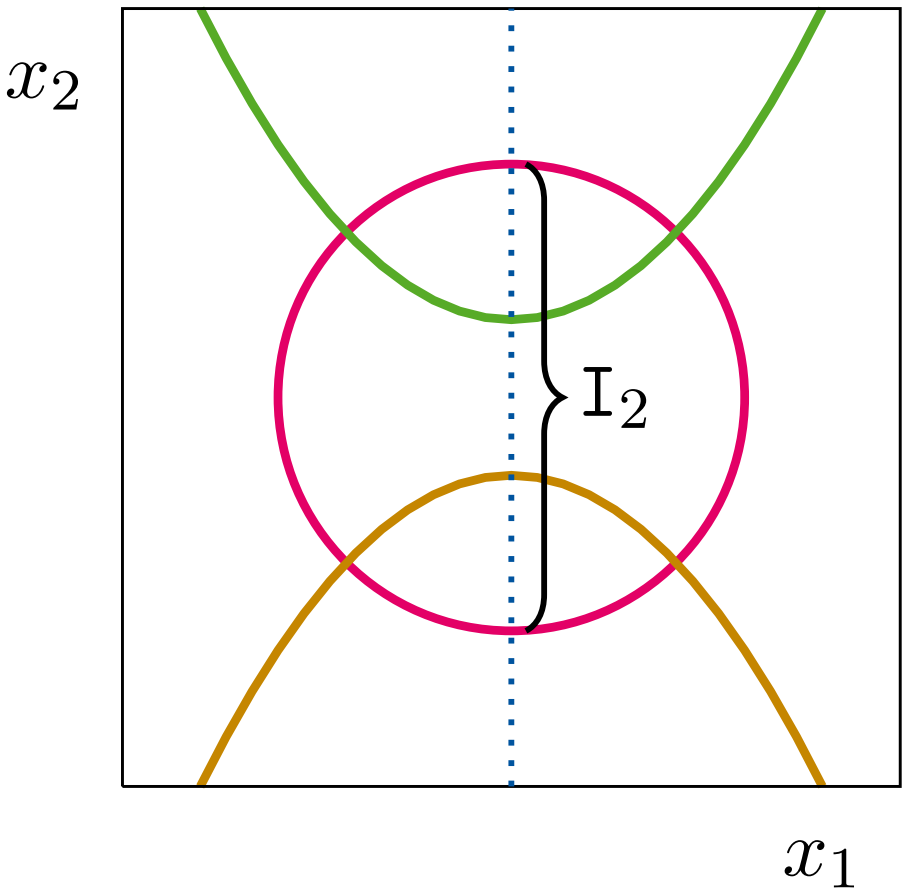
Cylindrical algebraic coverings

$$x^2 - y > -0.5 \wedge x^2 + y^2 > 1 \wedge -x^2 - y < 0.5$$



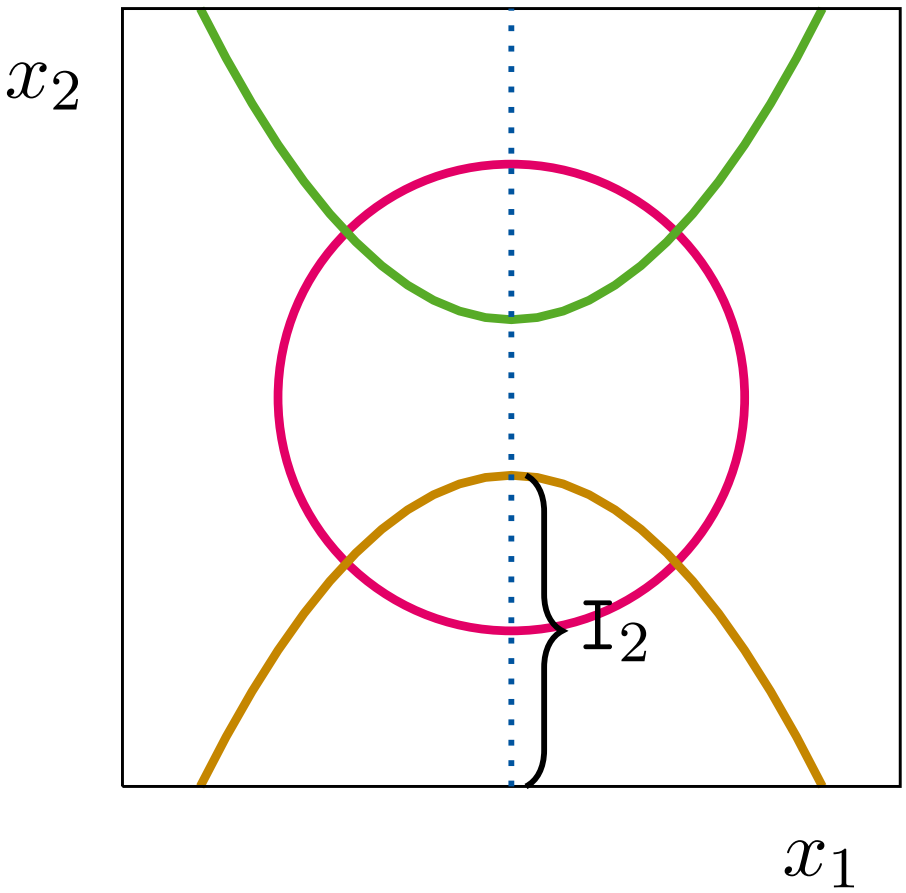
Cylindrical algebraic coverings

$$x^2 - y > -0.5 \wedge x^2 + y^2 > 1 \wedge -x^2 - y < 0.5$$



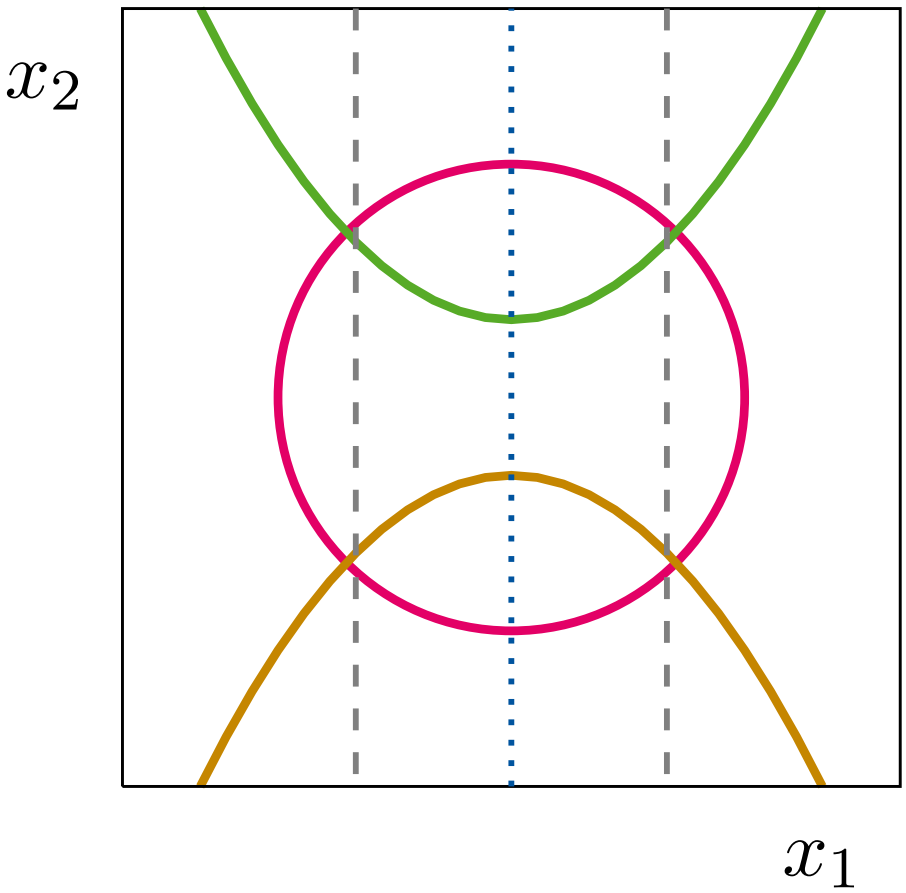
Cylindrical algebraic coverings

$$x^2 - y > -0.5 \wedge x^2 + y^2 > 1 \wedge -x^2 - y < 0.5$$



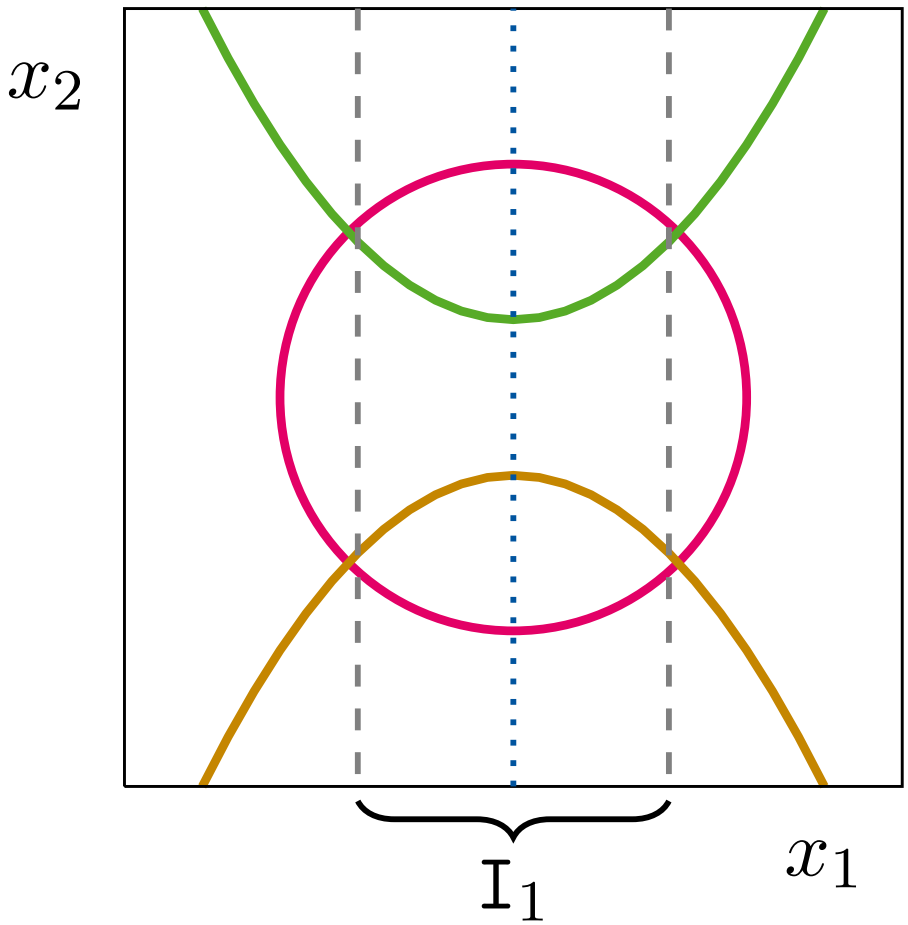
Cylindrical algebraic coverings

$$x^2 - y > -0.5 \wedge x^2 + y^2 > 1 \wedge -x^2 - y < 0.5$$

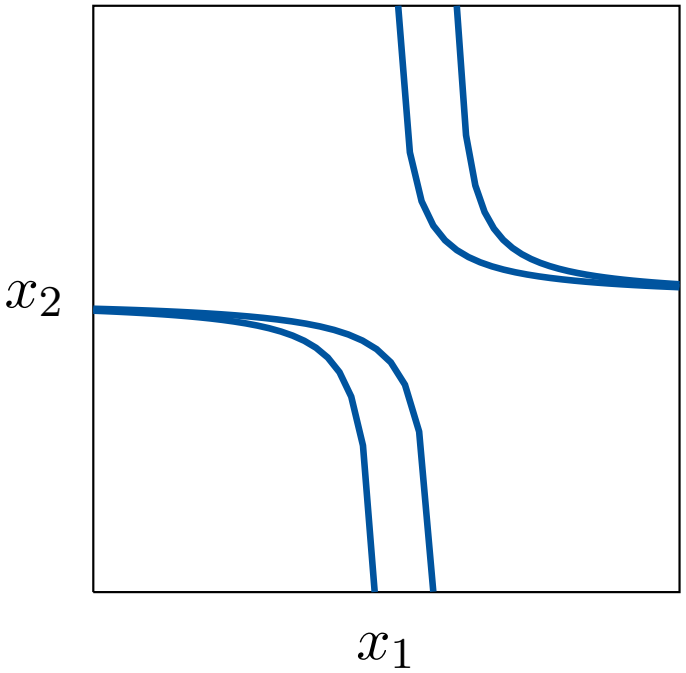


Cylindrical algebraic coverings

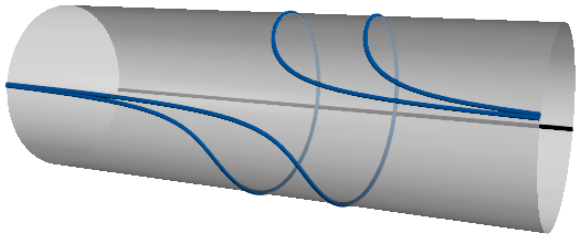
$$x^2 - y > -0.5 \wedge x^2 + y^2 > 1 \wedge -x^2 - y < 0.5$$



Projective delineability



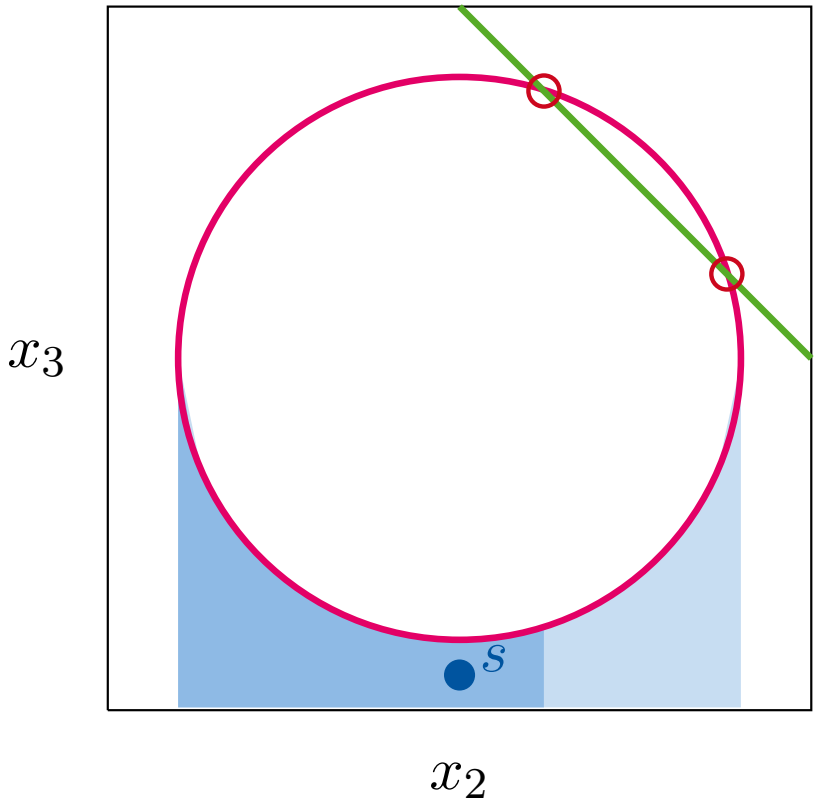
Real space

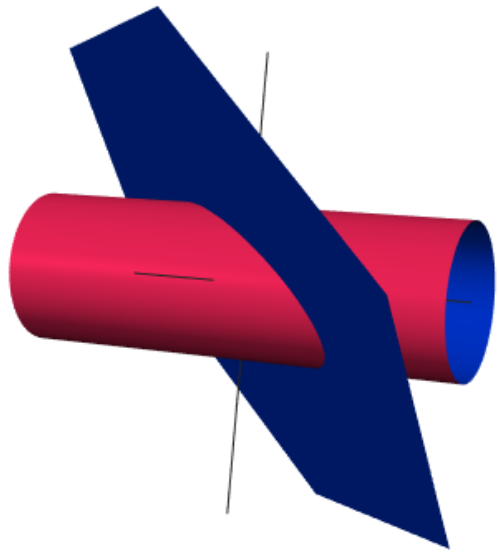


Projective space

Challenge: How to define ordering relations of projective roots?

Irrelevant intersections and local delineability

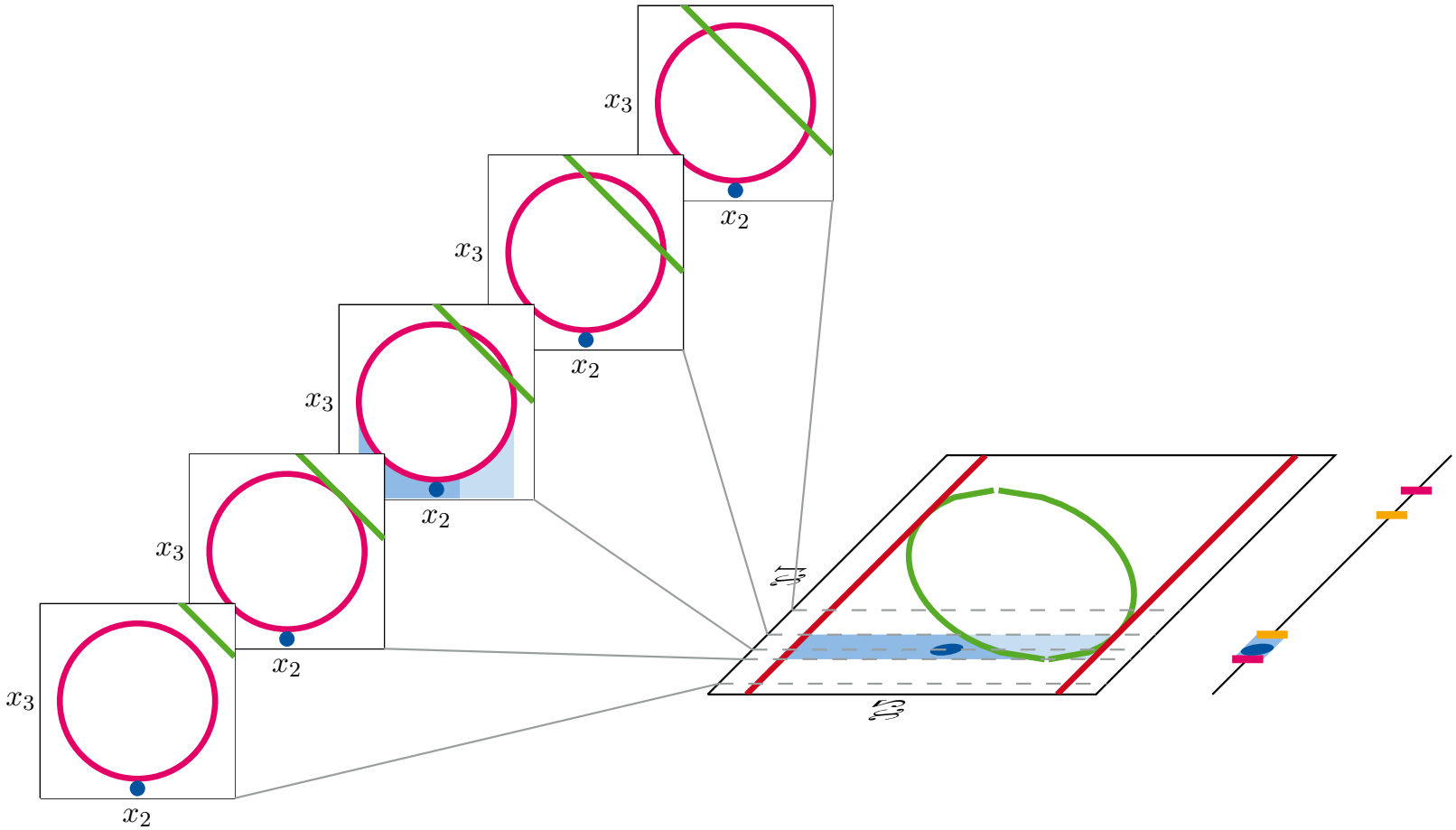




$x_2^2 + x_3^2 - 1$, $-1.25 + x_2 + x_3$

$x_2^2 + x_3^2 - 1$ and $x_1 + x_2 + x_3$

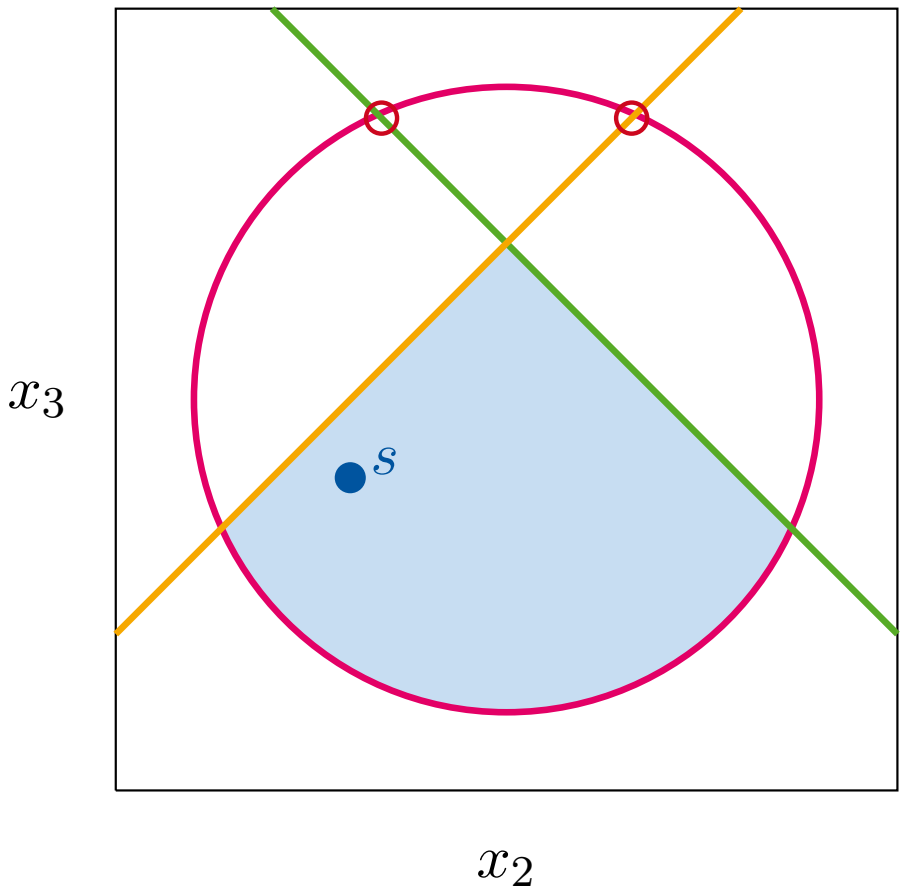
Irrelevant intersections and local delineability



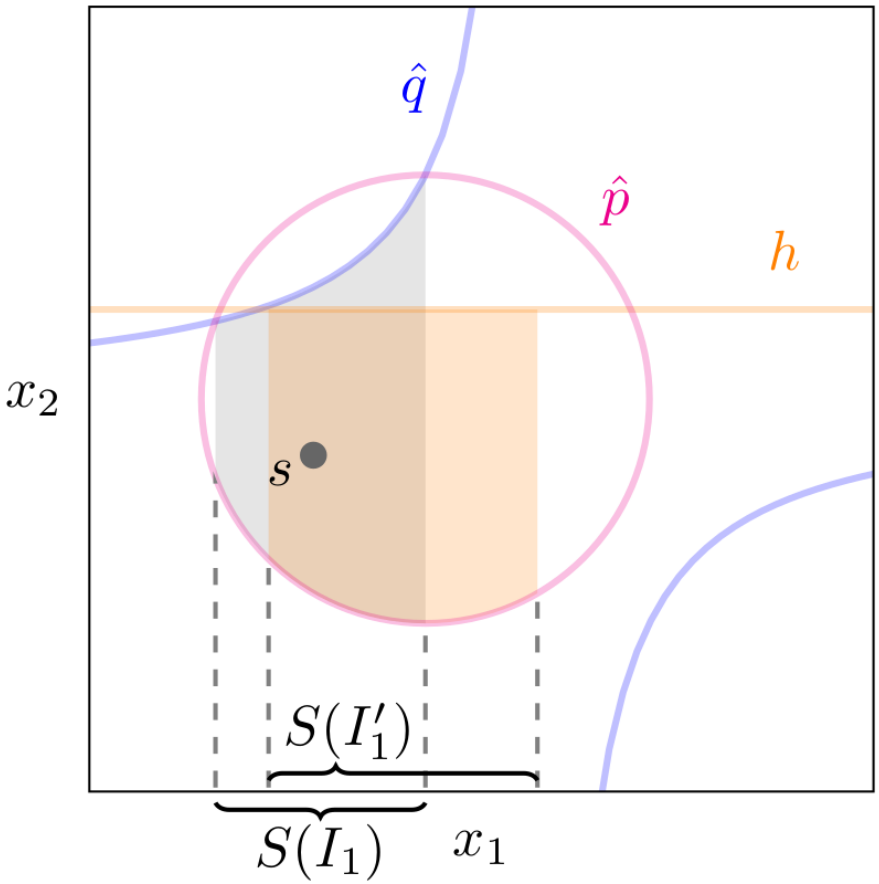
Solution: The resultant needs to be locally delineable

Extension: Weak interval bounds

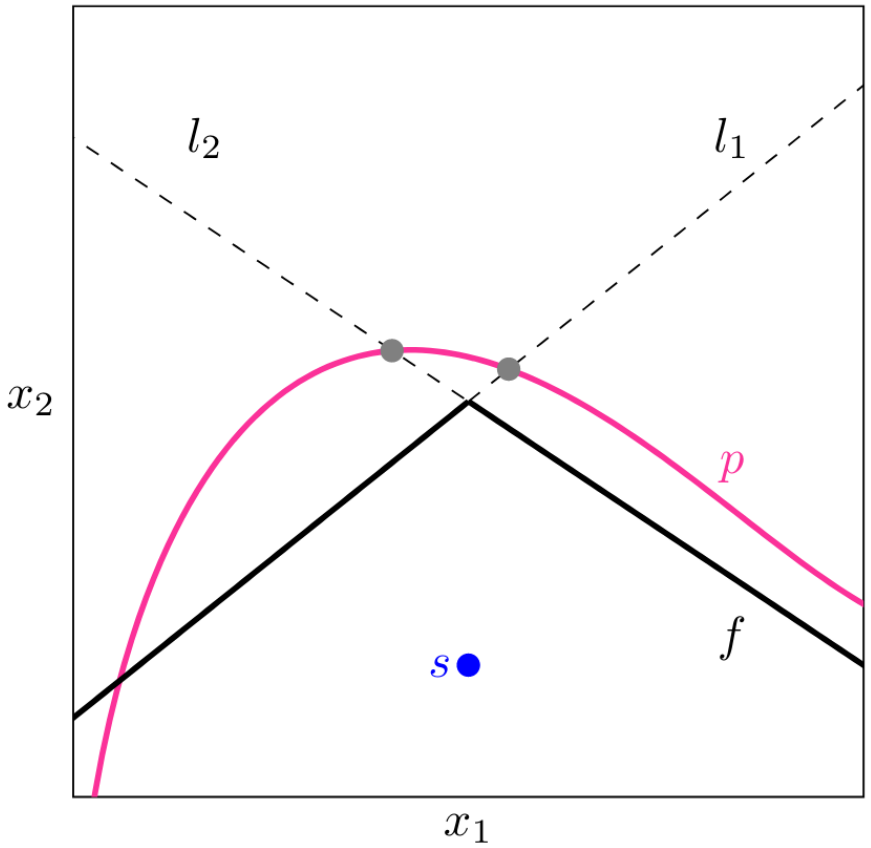
Compound bounds



Under-approximating cells



Linear under-approximation
(Valentin Promies)



Piecewise linear
under-approximation
(Paul Wagner)

Future work

Future work

- Heuristics: solve combinatorial problems exactly

Future work

- Heuristics: solve combinatorial problems exactly
- Completeness through Lazard's projection

Future work

- Heuristics: solve combinatorial problems exactly
- Completeness through Lazard's projection
- Proof generation

Conclusion

Conclusion

- Levelwise single cell generation

Conclusion

- Levelwise single cell generation
- Proof system separating correctness from algorithmic decisions

Conclusion

- Levelwise single cell generation
- Proof system separating correctness from algorithmic decisions
- Interfaces for heuristics

Conclusion

- Levelwise single cell generation
- Proof system separating correctness from algorithmic decisions
- Interfaces for heuristics
- Experimental results show potential for better heuristics

Conclusion

- Levelwise single cell generation
- Proof system separating correctness from algorithmic decisions
- Interfaces for heuristics
- Experimental results show potential for better heuristics
- Extensions allow for even more degrees of freedom