# Cylindrical Algebraic Decomposition

Grubhofer Alexander

# History

- invented by George E. Collins in 1975
- improved by H. Hong, C. Brown and others
- implemented by A. Strzebonski in Mathematica

# Introduction

- **Input:** system of polynomial inequalities over the reals
- **Output:** an equivalent system of inequalities, which
  - has a nice structural property to answering nontrivial questions

# Notations

- **[Cells]** Given a finite set $\{p_1,\ldots., p_m\} \in \mathbb{R}[x_1,\ldots x_n]$ this set induces a **decomposition** ("partition") of $\mathbb{R}^n$ into maximal sign-invariant **cells** ("regions")

  Look at an example (Mathematica)

  This example has 13 cells.

# Notations

- Let be $\pi_n : \mathbb{R}^n \to \mathbb{R}^{n-1}$ $(x_1,....,x_n) \mapsto (x_1,....,x_{n-1})$
  the canonical projection
- **[Cylindrical]** Let $p_1,....,p_m \in \mathbb{Q}[x_1,...x_n]$
  $\{p_1,....,p_m\}$ is called cylindrical iff

1. For any two cells C,D the images $\pi_n(C),\pi_n(D)$ are either identical or disjoint.
2. The algebraic decomposition $\{p_1,....,p_m\} \cap \mathbb{Q}[x_1,...x_{n-1}]$ is cylindrical

# Algortihm

- 3 Phases
  - Projection
  - Lifting
  - Solution

# Projection Phase

- Input polynomials $p_1, \ldots., p_m$
- find $q_1, \ldots., q_k$ s.t. the algebraic decomposition of $\{p_1, \ldots.., p_m, q_1, \ldots., q_k\}$ is cylindrical

# Projection Phase (2)

- The projection operator is defined as

$$\begin{array}{ccc} A & \mapsto & P_n(A) \\ \cap & & \cap \\ \mathbb{R}[x_1,....,x_n] & & \mathbb{R}[x_1,....,x_{n-1}] \end{array}$$

such that:

If B is a CAD of $P_n(A)$ ,then $A \cup B$ is a CAD of A.

# Projection Phase (3)

- $P_n(A)$ is defined as

$$P_n(A) = \bigcup_{p \in A} coeffs_{x_n}(p) \cup \bigcup_{p \in A} disc_{x_n}(p) \cup \bigcup_{p,q \in A} res_{x_n}(p,q)$$

$$\dim(p) = n > 2$$

$$disc_{x_n}(p) = (-1)^{n*(n-1)/2} res_{x_n}(p, \frac{\partial}{\partial x_n} p)$$

$$disc_{x_n}(p) = res_{x_n}(p, \frac{\partial}{\partial x_n} p)$$

# Projection algorithm

- **Input:** $A \subseteq \mathbb{Q}[x_1, ...., x_n]$
- **Output:** $C \subseteq \mathbb{Q}[x_1, ...., x_n]$ s.t. $A \subseteq C$ and C is CAD

1. C:=A
2. for k=n down to 2
3. $C := C \cup P_k(C \cap \mathbb{Q}[x_1, ...., x_k])$
4. return C

# Lifting Phase

- Construct sample points for each cell in this decomposition considering one dimension after the other in a bottom-up fashion.

# Lifting Phase (Case 1 variable)

$p_1(x), \ldots, p_m(x) \in (\overline{\mathbb{Q}} \cap \mathbb{R})[x]$

- look for all real roots $r_1, \ldots, r_k$ of $\forall i : p_i(x)$

- choose $\rho_0, \ldots, \rho_k \in \mathbb{Q}$ such that

$\rho_0 < r_1 \qquad r_i < \rho_i < r_{i+1} \qquad r_k < \rho_k$

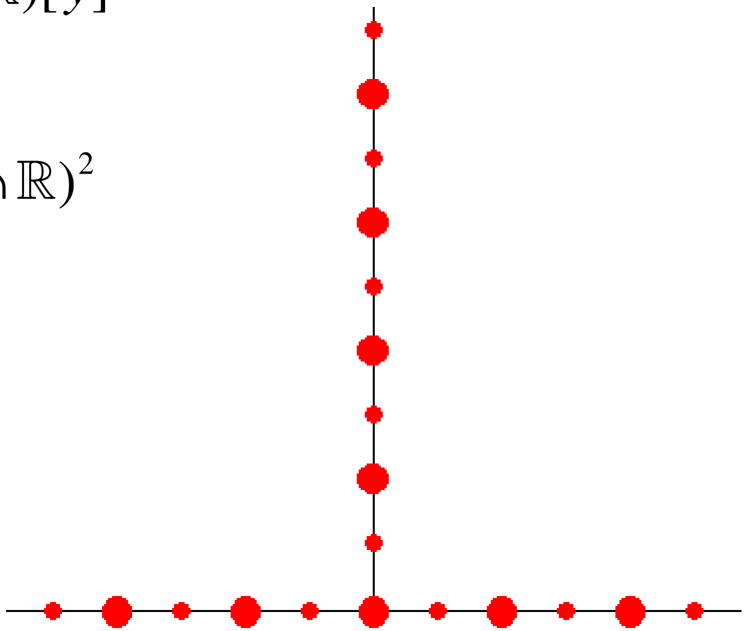
- the sample points are

$\rho_0, r_1, \rho_1, r_2, \ldots, r_k, \rho_k$

# Lifting Phase (Case 2 variables)

$p_1(x, y), \ldots, p_m(x, y) \in (\overline{\mathbb{Q}} \cap \mathbb{R})[x, y]$

- look for sample points $r_1, \ldots, r_{2k+1}$ of $p_i(x, y)$ which are free of y
- for each $r_i$ , look for sample points $r_{i,1}, \ldots, r_{i,l}$ for the polynomials $p_i(r_i, y) \in (\overline{\mathbb{Q}} \cap \mathbb{R})[y]$
- the sample points are $(r_i, r_{i,j}) \in (\overline{\mathbb{Q}} \cap \mathbb{R})^2$



# Lifting algorithm

- **Input**: a CAD $C \subseteq \mathbb{Q}[x_1, \ldots., x_n]$
- **Output:** a set of sample points $\sigma \in (\overline{\mathbb{Q}} \cap \mathbb{R})^n$ for C

1. $S_1 :=$ sample points for $C \cap \mathbb{Q}[x_1]$
2. for k=2 to n do
3. $C_k := C \cap \mathbb{Q}[x_1, \ldots., x_k]$
4. $S_k := \bigcup_{\sigma \in S_{k-1}} \sigma \times \text{samplePoints} \, for \, C_k \big|_{(x_1, \ldots., x_k) = \sigma}$
5. return S

# Solution (or Extension) Phase

- select the regions of interest
- check for some simplification
- construct a solution formula accordingly

# Solution Phase (2)

- Assigning truth values to cells amounts to determining the sign of polynomials at the sample point

- Quantifier elimination:
  - $\forall x \in \mathbb{R}$ becomes „for all sample points“
  - $\exists x \in \mathbb{R}$ becomes „for at least one sample point“

- Formula construction is easy

# Example

- See Mathematica (Circle, Tacnode)

# CAD

- CAD brings the system of inequalities in following form:

$$\ldots \vee (. < x_1 < . \wedge [\ \ ]) \vee (x_1 = . \wedge [\ \ ]) \vee \ldots$$

$$[\ \ ] :\Leftrightarrow \ldots \vee (. < x_2 < . \wedge [\ \ ]_1) \vee (x_2 = . \wedge [\ \ ]_1) \vee \ldots$$

# CAD (1)

- for the unit circle follows:

$$(x_1 = 1 \land y = 0) \lor (x_1 = -1 \land y = 0) \lor$$

$$\lor(-1 < x_1 < 1 \land [\ \ ])$$

$$[\ \ ] :\Leftrightarrow (y = \sqrt{1 - x^2} \lor -\sqrt{1 - x^2} < y < \sqrt{1 - x^2} \lor y = -\sqrt{1 - x^2})$$

# CAD (2)

- CAD for the unit sphere

$$(x_1 = 1 \wedge y = 0 \wedge z = 0) \vee (x_1 = -1 \wedge y = 0 \wedge z = 0) \vee$$

$$\vee(-1 < x_1 < 1 \wedge [\ ])$$

$$[\ ] :\Leftrightarrow ((y = \sqrt{1-x^2} \wedge z = 0)$$

$$\vee(-\sqrt{1-x^2} < y < \sqrt{1-x^2} \wedge [\ ]_1)$$

$$\vee(y = -\sqrt{1-x^2} \wedge z = 0))$$

$$[\ ]_1 :\Leftrightarrow ((z = -\sqrt{1-x^2-y^2} \vee -\sqrt{1-x^2-y^2} < z < \sqrt{1-x^2-y^2} \vee z = \sqrt{1-x^2-y^2})$$

# Example

- We have 2 formulas

$$f_1(x) = x^2 - 2x$$

$$f_2(x) = x^2 - 4x + 3$$

- The real roots of both are

$$x_{1,1} = 0; x_{1,2} = 2$$

$$x_{2,1} = 1; x_{2,2} = 3$$

# Example (2)

- The sample points are
  $-1, 0, 1/2, 1, 3/2, 2, 5/2, 3, 4$
- Region of truth for
  $\exists x \in \mathbb{R} : (x^2 - 2x \geq 0 \wedge x^2 - 4x + 3 \geq 0)$
  it is true for x=0, false for x=1
  $\forall x \in \mathbb{R} : (x^2 - 2x \geq 0 \wedge x^2 - 4x + 3 \geq 0)$
  it is allways false

# Example(3)

- The **CylindricalDecomposition** brings the system in following form:

  x<=0||x>=3.