ISSAC 2014 (extended tutorial: handout) at Kobe Univ.

FUJITSU

shaping tomorrow with you

Effective quantifier elimination for industrial applications

Hirokazu Anai

FUJITSU LABORATORIES LTD
Kyusyu University
National Institute of Informatics

Special thanks to Dr. Iwane (NII / Fujitsu lab) for his collaboration.

ISSAC 2014 - Tutorials

July 22	Extended Tutorials	
9:00-11:10	**Lihong Zhi** *Symbolic-numeric algorithms for computing validated results*	**Mitsushi Fujimoto** *How to develop a mobile computer algebra system*
11:10-12:45	lunch break	
12:45-14:55	**Francois Le Gall** *Algebraic Complexity Theory and Matrix Multiplication*	**Hirokazu Anai** Effective quantifier elimination for industrial applications
14:55-15:15	coffee break	
15:15-17:25	**Hidefumi Ohsugi** *Gröbner bases of toric ideals and their application*	**Hiroyuki Goto** *An introduction to max-plus algebra*
17:45-19:45	Welcome Reception	

Abstract

- In this tutorial, we will give an overview of typical algorithms of quantifier elimination over the reals and illustrate their actual applications in industry. Some recent research results on computational efficiency improvement of quantifier elimination algorithms, in particular for solving practical industrial problems, will be also mentioned. Moreover, we will briefly explain valuable techniques and tips to effectively utilize quantifier elimination in practice.

Contents

- Introduction
- Quantifier Elimination
  - Definition
  - Brief history (with applications in control system design)
  - Typical Algorithms (for practical use)
  - Complexity
  - Usage
    - Symbolic/parametric optimization
    - Symbolic-numeric optimization
- Applications
  - System Design: Control system design
  - Manufacturing Design : Optimal shape design
- Software
- References

Quantifier Elimination (QE)

Real Quantifier Elimination

Quantifier elimination

- algorithm to compute an equivalent quantifier-free formula for a given first-order formula over the reals

Input : first-order formula in the elementary theory of **real closed fields**

$$F := Q_1 x_1 \cdots Q_n x_n [\varphi(x_1, \ldots, x_n, r_1, \ldots, r_t)],$$

where $Q_i \in \{\exists, \forall\}$ and φ is a quantifier-free formula

- formula φ : polynomial equations, inequalities, inequations over $\boldsymbol{R}$,
  Boolean operations[$\wedge$, $\vee$, $\neg$, $\Rightarrow$, etc]

Output: an equivalent quantifier-free formula in free variables

$$\psi(r_1, \ldots, r_t)$$

- ***Feasible regions*** of free variables as semi-algebraic sets
- ***True*** or ***False*** if all variables are quantified (*Decision problem*)

Examples: Quantifier Elimination

FUJITSU

Input

First-order formula

Output

An equivalent quantifier-free formula

$$\forall x \ (x^2 + bx + c > 0) \iff b^2 - 4c < 0$$

$$\exists x \ (ax^2 + bx + c = 0) \iff \begin{array}{l} (a \neq 0 \ \wedge \ b^2 - 4ac \geq 0) \ \vee \\ (a = 0 \ \wedge \ b \neq 0) \ \vee \\ (a = 0 \ \wedge \ b = 0 \ \wedge \ c = 0) \end{array}$$

$$\begin{array}{r} \forall x \exists y \ (x^2 + xy + b > 0 \wedge \\ x + ay^2 + b \leq 0) \end{array} \iff a < 0 \ \wedge \ b > 0$$

QE algorithms

Typical QE algorithms

FUJITSU

General QE algorithm

- For arbitrary formulas
  - QE by Cylindrical Algebraic Decomposition (CAD)

$$\mathrm{O}(2^{2^n}) : \qquad n = \#\text{ of variables}$$

Special QE algorithm

- For restricted classes of formulas
  - QE by Virtual Substitution
    - for linear/quadratic formulas (w.r.t. quantified variables)

$$\mathrm{O}(2^k) : \qquad k = \#\text{ of quantified variables}$$

  - QE by the Sturm-Habicht sequence
    - for sign definite condition (SDC) : $\forall x\ (x \geq 0 \rightarrow f(x) > 0)$

$$\mathrm{O}(2^d) : \qquad d = \deg_z(f(z))$$

QE algorithms - some more

- Cylindrical Algebraic Decomposition — Collins 1975
- $\exists x(a_2x^2 + a_1x + a_0 = 0 \wedge \phi(x, a_2, a_1, a_0, \ldots))$ — Hong 1993
- $\forall x(f(x) > 0)$ — Gonzalez-Vega 1989, Yang et.al, 1996
- $\forall x(x \geq 0 \rightarrow f(x) > 0)$ — Anai et.al, 1999
  - Sign Definite Condition
- Low degree formula w.r.t quantified variables (degree limit : n=1,2,3)
  - Virtual Substitution
  - n = 1 : Weispfenning et.al, 1988
  - n = 2: Loos et.al, 1993
  - n = 3: Weispfenning 1993
- One block QE — Hong et.al., 2012
  - Allows measure-zero error

Brief History of QE algorithms

FUJITSU

1930	■ Tarski proved QE is possible over $\boldsymbol{R}$
1951	■ Tarski proposed a QE algorithm over $\boldsymbol{R}$ ■ Computational complexity cannot be bound by any tower of exponentials
1975	■ Collins made a breakthrough ■ QE by Cylindrical Algebraic Decomposition (CAD) ■ Computational complexity down to doubly exponential w.r.t. the number of variables
1988	■ QE computation is proved to be heavy ■ Doubly exponential in worst case (Davenport and Heinz, 1988)
1990	■ QEPCAD: First CAD-based QE implementation (Hong)

- 1980's Different approaches
  - QE algorithms for a restricted class of input
    - QE for up to linear/quartic formulas, Positive polynomial condition

Typical QE tools

FUJITSU

QEPCAD	**CAD** RISC-Linz + etc. (G.Collins, H.Hong, C.Brown)
REDLOG REDUCE	**VS, CAD,** Univ. Passau (T.Sturm, A.Dolzmann,V.Weispfenning)
Wolfram Mathematica 9	**CAD, VS** Wolfram Research, Inc. (A.Strzebonski)
SyNRAC Maple 17 The Essential Tool for Mathematics and Modeling	**CAD, VS, SDC** Fujitsu Laboratories Ltd. (H.Yanami, H.Iwane, H.Anai)

QE algorithm:
Cylindrical Algebraic Decomposition

Cylindrical Algebraic Decomposition (CAD)

FUJITSU

QE by CAD

- First proposed by G.E.Collins 1975
- QE by partial CAD (Collins & Hong 1991)

CAD

- **Input** : $F_r \subset \mathbb{Q}[x_1, \ldots, x_r]$
- **Output :** $\mathbb{R}^r = \cup C_i^{(r)}$
  - partition of the r-dimensional real space, where all the input polynomials are sign-invariant within each cell

Implementation

- QEPCAD, Mathematica, SyNRAC

Properties

- Complexity: $\mathrm{O}(2^{2^n})$: $n = \#$ of variables
- Output formula is in general simple
- No restriction on input formula

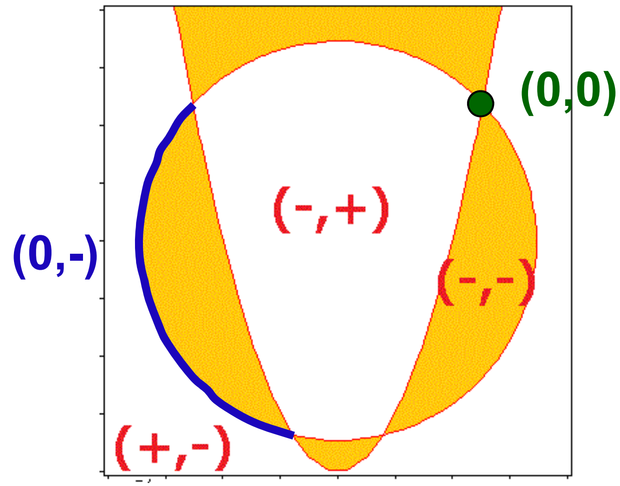
Cylindrical Algebraic Decomposition (CAD)

FUJITSU

CAD

- **Input** : $F_r \subset \mathbb{Q}[x_1, \ldots, x_r]$
- **Output :** $\mathbb{R}^r = \cup C_i^{(r)}$
  - partition of the r-dimensional real space, where all the input polynomials are sign-invariant within each cell

$F_2 = \{x_1^2 + x_2^2 - 3, x_2 - 2x_1^2 + 2\}$



CAD algorithm

- consists of 3 phases
  - projection phase
  - base phase
  - lifting phase

projection factor set

$\boldsymbol{F}_r \subset \mathbb{Q}[x_1, \ldots, x_r] \rightarrow \mathbb{R}^r = \cup C_i^{(r)}$

projection ↓ ↑ lifting

$\boldsymbol{F}_{r-1} \subset \mathbb{Q}[x_1, \ldots, x_{r-1}]$ $\mathbb{R}^{r-1} = \cup C_i^{(r-1)}$

projection ↓ ↑ lifting

$\vdots$ $\vdots$

projection ↓ ↑ lifting

$\boldsymbol{F}_2 \subset \mathbb{Q}[x_1, x_2]$ $\mathbb{R}^2 = \cup C_i^{(2)}$

projection ↓ ↑ lifting

$\boldsymbol{F}_1 \subset \mathbb{Q}[x_1] \xrightarrow{\text{base}} \mathbb{R}^1 = \cup C_i^{(1)}$

Cylindrical Algebraic Decomposition (CAD)

FUJITSU

3 phases of CAD algorithm

- projection phase
  - Many "projection operators" have been proposed
  - Projection operators that produce small sets are good
  - Example:
    - Input $F_k \subset \mathbf{Q}[x_1, \ldots, x_k]$

$$\Big\downarrow PROJ(F_k)$$

    - Output $F_{k-1} \subset \mathbf{Q}[x_1, \ldots, x_{k-1}]$

$$
\begin{array}{l}
PROJ(F_k) \\
\quad C \leftarrow \{\mathrm{coeffs}(f, x_k) | f \in F_k\} \\
\quad D \leftarrow \{\mathrm{discriminant}(f, x_k) | f \in F_k\} \\
\quad R \leftarrow \{\mathrm{resultant}(f, g, x_k) | f, g \in F_k, f \neq g\} \\
\quad \text{return IrreducibleFactors}(\{C, D, R\})
\end{array}
$$

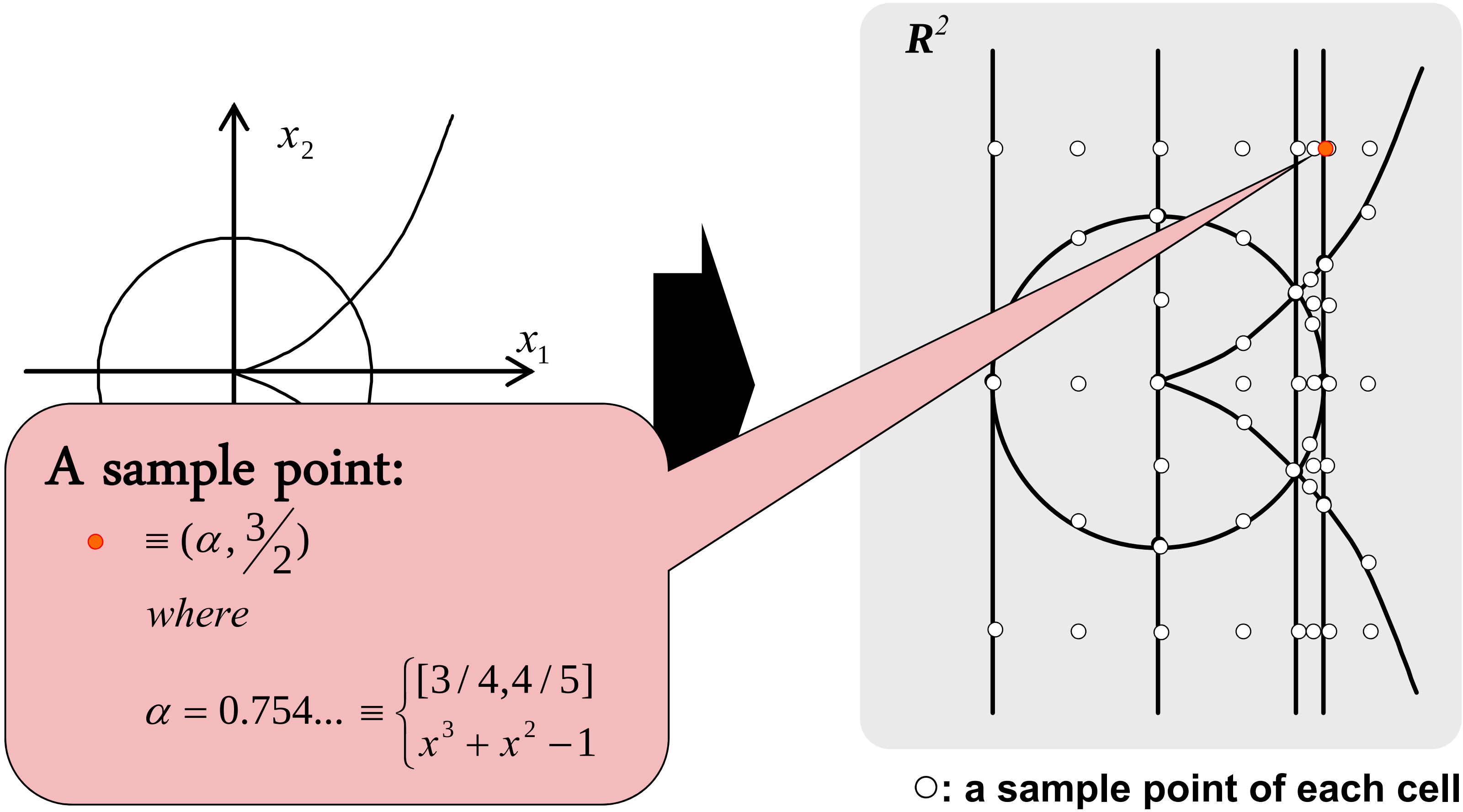
- base phase
  - Real root isolation
    - Often univariate poly over algebraic ext.
- Lifting phase
  - Algebraic extension
  - validated numerics (SNCAD)

$$
\begin{array}{ccc}
\boldsymbol{F}_r \subset \mathbb{Q}[x_1, \ldots, x_r] & \longrightarrow & \mathbb{R}^r = \cup C_i^{(r)} \\
\downarrow \text{projection} & & \uparrow \text{lifting} \\
\boldsymbol{F}_{r-1} \subset \mathbb{Q}[x_1, \ldots, x_{r-1}] & & \mathbb{R}^{r-1} = \cup C_i^{(r-1)} \\
\downarrow \text{projection} & & \uparrow \text{lifting} \\
\vdots & & \vdots \\
\downarrow \text{projection} & & \uparrow \text{lifting} \\
\boldsymbol{F}_2 \subset \mathbb{Q}[x_1, x_2] & & \mathbb{R}^2 = \cup C_i^{(2)} \\
\downarrow \text{projection} & & \uparrow \text{lifting} \\
\boldsymbol{F}_1 \subset \mathbb{Q}[x_1] & \xrightarrow{\text{base}} & \mathbb{R}^1 = \cup C_i^{(1)}
\end{array}
$$

Cylindrical Algebraic Decomposition

Example

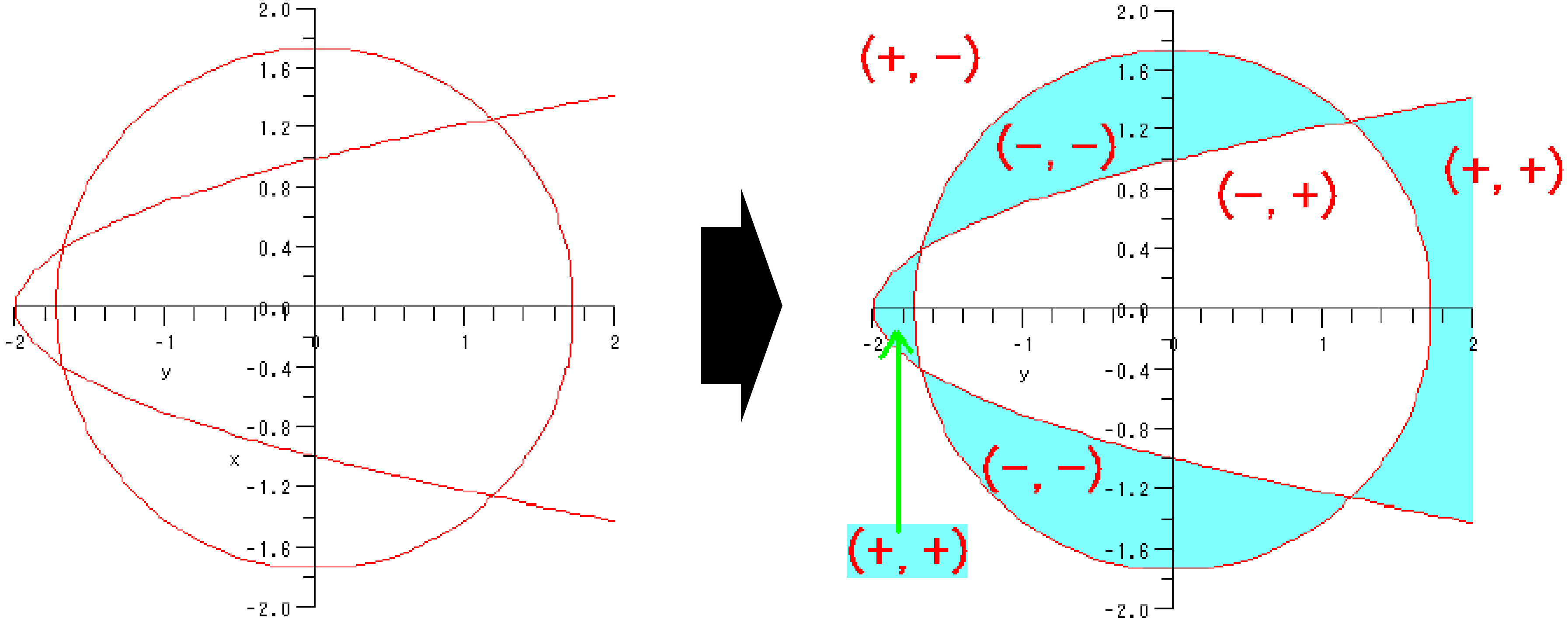
- **Input :** $F_2 = \{f_1(x_1, x_2) = x_1^2 + x_2^2 - 1, f_2(x_1, x_2) = x_1^3 - x_2^2\}$
- **Output :** an F_2 - sign invariant CAD of $\boldsymbol{R}^2$



○: **a sample point of each cell**

Cylindrical Algebraic Decomposition

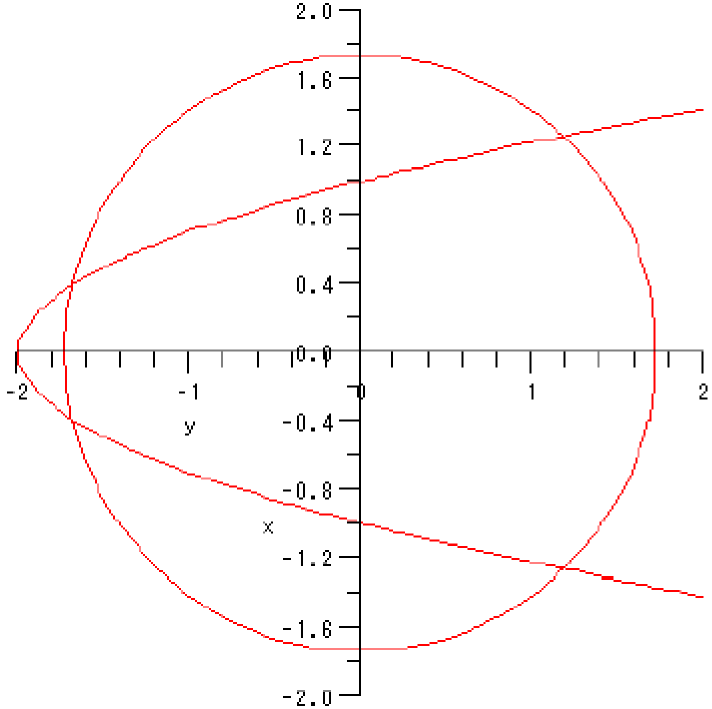
- Example
  - **Input** : $F_2 = \{f_1(x,y) = x^2 + y^2 - 3, f_2(x,y) = y - 2x^2 + 2\}$
  - **Output :** an F_2 - sign invariant CAD of $\boldsymbol{R}^2$



Cylindrical Algebraic Decomposition

FUJITSU

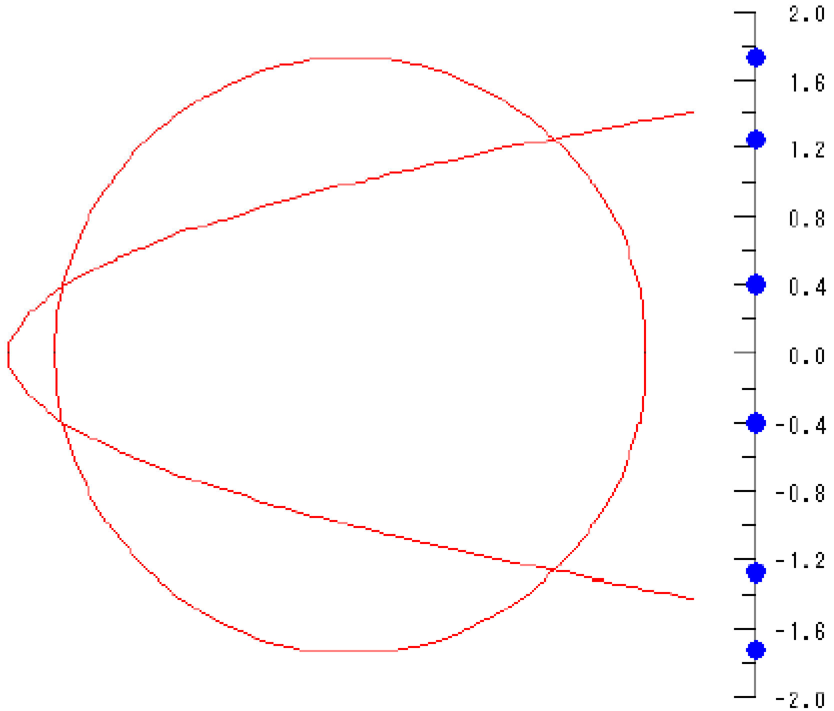
- Example
  - **Input** : $F_2 = \{f_1(x,y) = x^2 + y^2 - 3, f_2(x,y) = y - 2x^2 + 2\}$
  - **Output :** an F_2- sign invariant CAD of $\boldsymbol{R}^2$



CAD: Projection, Base phase

FUJITSU

- Example
  - **Input** : $F_2 = \{f_1(x,y) = x^2 + y^2 - 3, f_2(x,y) = y - 2x^2 + 2\}$
  - **Output :** an F_2 - sign invariant CAD of $\boldsymbol{R}^2$



● real root
(critical points)

- **Base**

real root
finding

- **Projection**

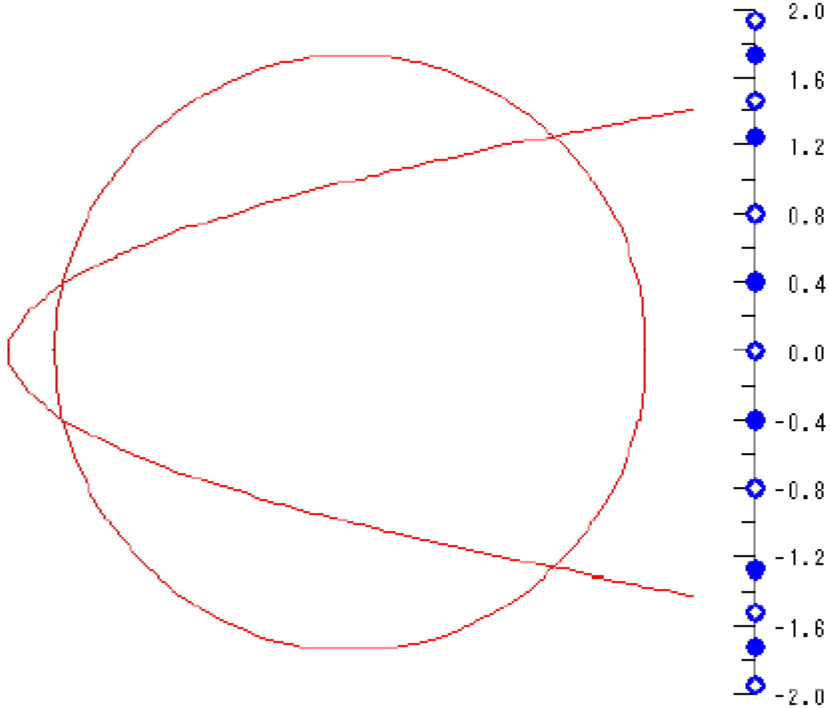
$$F_2 = \{x^2 + y^2 - 3, y - 2x^2 + 2\} \xrightarrow[PROJ(F_2)]{\text{projection}} F_1 = \{4x^4 - 7x^2 + 1, x^2 - 3\}$$

projection factors

CAD: Base phase

FUJITSU

- **Example**
  - **Input** : $F_2 = \{f_1(x,y) = x^2 + y^2 - 3, f_2(x,y) = y - 2x^2 + 2\}$
  - **Output :** an F_2 - sign invariant CAD of $\boldsymbol{R}^2$



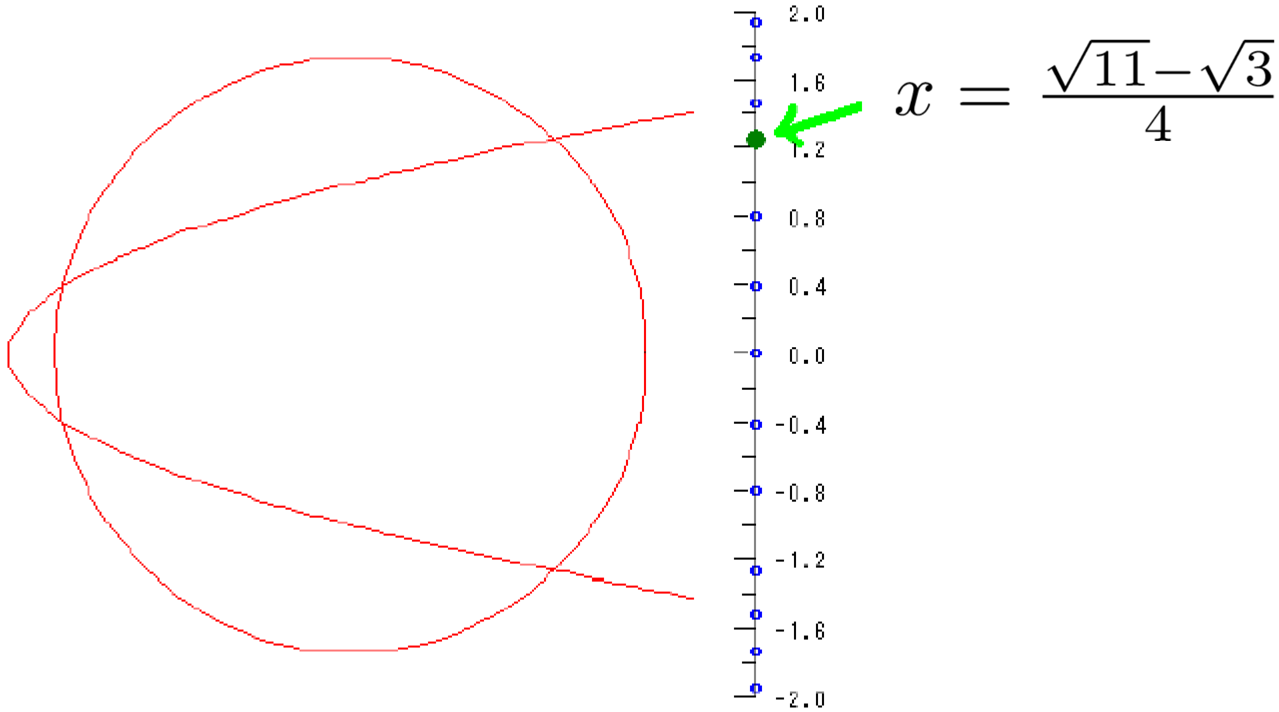
● real root (critical points)

○ Intermediate points

- **<u>Base</u>**
  - Select a point between each set of neighboring real roots => **sample points**

CAD: Projection, Base phase

- **Example**
  - **Input :** $F_2 = \{f_1(x,y) = x^2 + y^2 - 3, f_2(x,y) = y - 2x^2 + 2\}$
  - **Output :** an F_2 - sign invariant CAD of $\boldsymbol{R}^2$



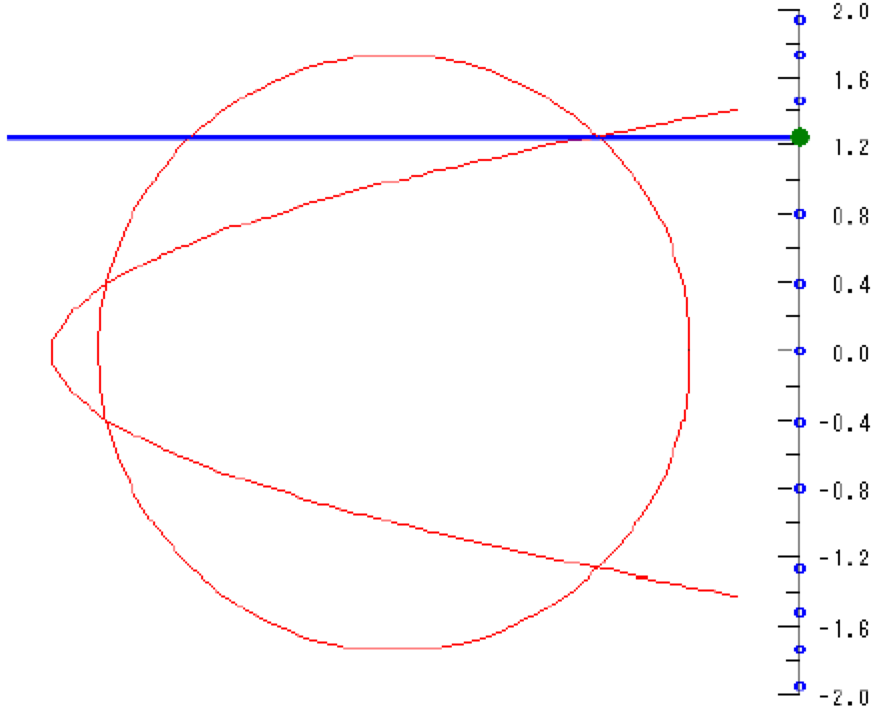
- **Base**
  - Select a point between each set of neighboring real roots => **sample points**

CAD: Lifting phase

FUJITSU

Example

- **Input** : $F_2 = \{f_1(x,y) = x^2 + y^2 - 3, f_2(x,y) = y - 2x^2 + 2\}$
- **Output :** an F_2 - sign invariant CAD of $\boldsymbol{R}^2$



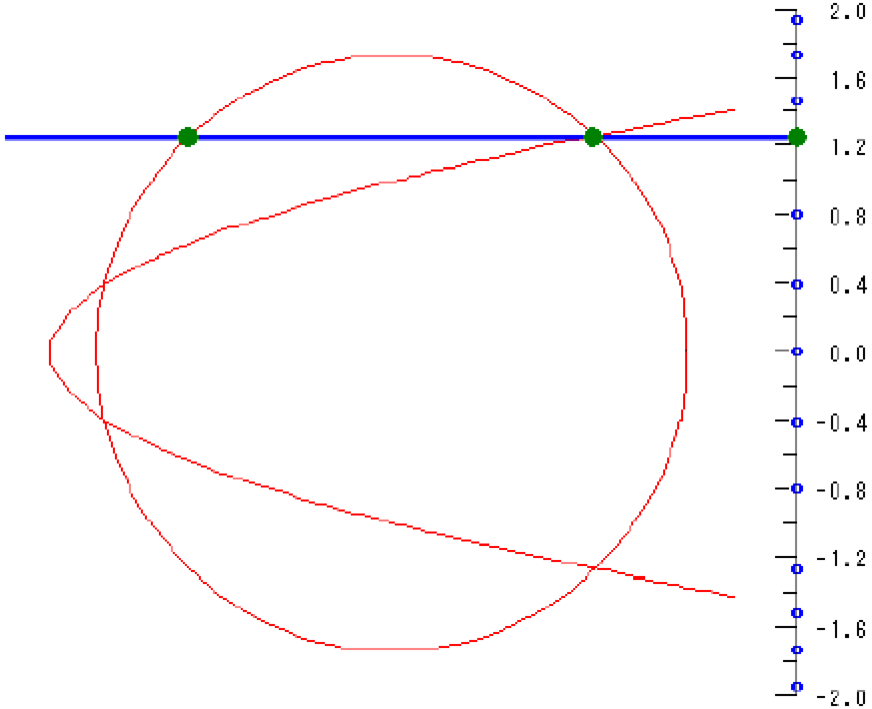
$$x = \frac{\sqrt{11}-\sqrt{3}}{4}$$

- **Lifting**
  - Lift the sample point (in R) to higher dimensions
    - Substitute $x = \frac{\sqrt{11}-\sqrt{3}}{4}$ for x in $F_2 = \{x^2 + y^2 - 3, y - 2x^2 + 2\}$
    - and we get a set of polynomials in y : $F_2'(y)$

CAD: Lifting phase

FUJITSU

- Example
  - **Input** : $F_2 = \{f_1(x,y) = x^2 + y^2 - 3, f_2(x,y) = y - 2x^2 + 2\}$
  - **Output :** an F_2- sign invariant CAD of $\boldsymbol{R}^2$

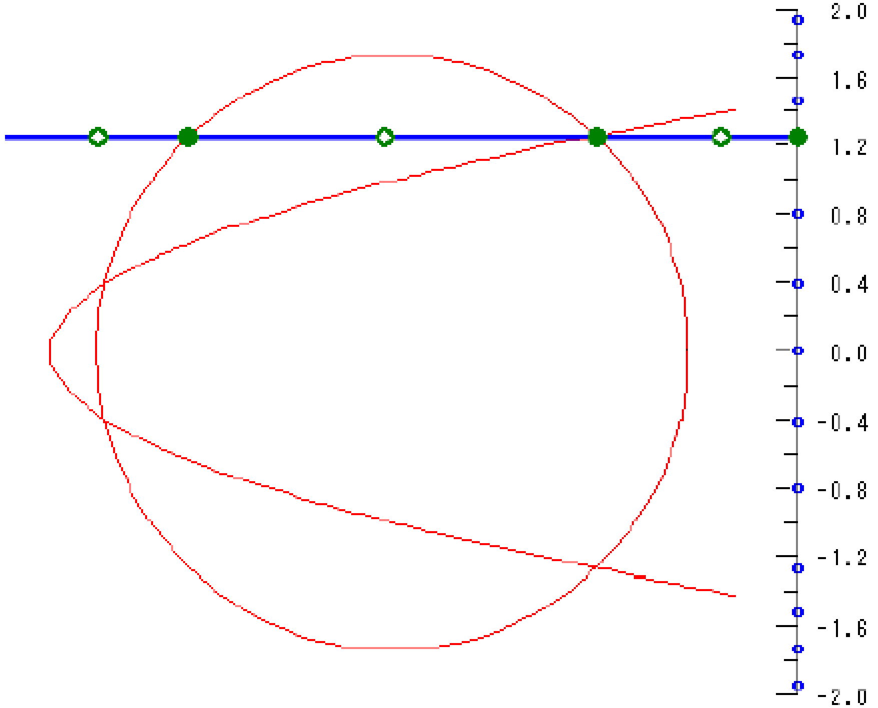


$$x = \frac{\sqrt{11}-\sqrt{3}}{4}$$

- **Lifting**
  - Fin real roots : of polynomials in $F_2'(y)$

CAD: Lifting phase

- Example
  - **Input** : $F_2 = \{f_1(x,y) = x^2 + y^2 - 3, f_2(x,y) = y - 2x^2 + 2\}$
  - **Output :** an F_2- sign invariant CAD of $\boldsymbol{R}^2$



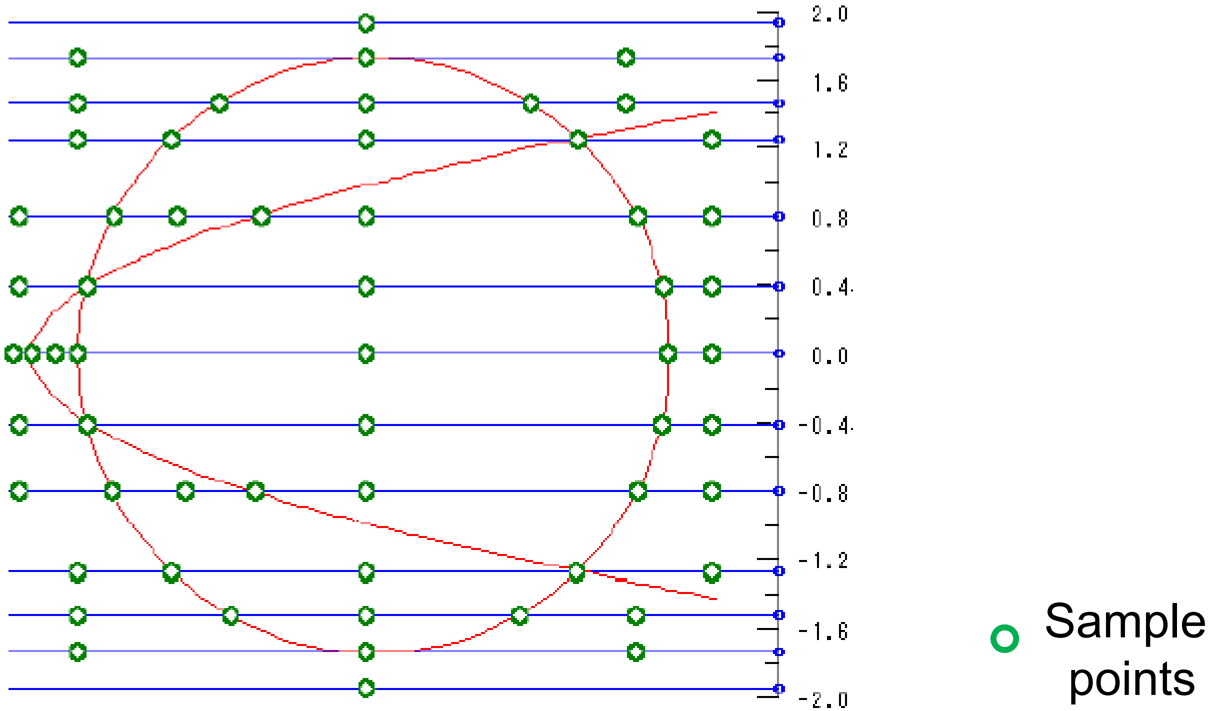
$$x = \frac{\sqrt{11} - \sqrt{3}}{4}$$

- **Lifting**
  - Fin real roots : of polynomials in $F_2'(y)$
  - Choose a point between each set of neighboring real roots

CAD: Lifting phase

FUJITSU

- **Example**
  - **Input :** $F_2 = \{f_1(x,y) = x^2 + y^2 - 3, f_2(x,y) = y - 2x^2 + 2\}$
  - **Output :** an F_2 - sign invariant CAD of $\boldsymbol{R}^2$



- **Lifting**
  - Do the same lifting process over all sample points
    - Fin real roots : of polynomials in $F_2'(y)$
    - Choose a point between each set of neighboring real roots

QE by CAD

Procedure of QE by CAD

- Input: First-order formula : $\exists x_2(x_1^2 + x_1^2 < 1 \land x_1 - x_2 < 0)$
  - CAD construction for $F_2 = \{x_1^2 + x_2^2 - 1, x_1 - x_2\}$
  - Collecting **true cells** in CAD in terms of the given first-order formula
  - Solution formula construction : Disjunction of the **formulas defining the true cells**

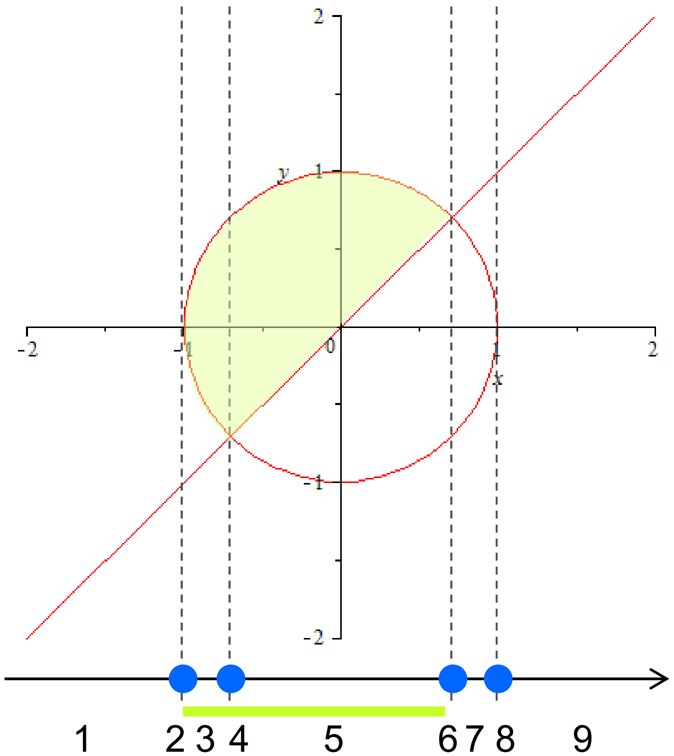
- Formula construction of a cell
  - Such formula is constructed from projection factors (CAD contains complete information about their signs)

- Note
  - “Simple” formulas are desirable!
    - Hong showed how to reduce simple formula construction to a combinatorial optimization problem
  - CAD’s ability to provide simple solution formulas is unique compared with special QE algorithms.

QE by CAD: formula construction

Formula construction of a cell

- First-order formula : $\exists x_2(x_1^2 + x_1^2 < 1 \wedge x_1 - x_2 < 0)$
- CAD construction: $F_2 = \{x_1^2 + x_2^2 - 1, x_1 - x_2\}$ $\quad F_1 = \{x_1 + 1, x_1 - 1, 2x_1^2 - 1\}$



cell	$x+1$	$x-1$	$2x^2-1$	T/F
1	$-$	$-$	$+$	F
2	0	$-$	$+$	F
3	$+$	$-$	$+$	T
4	$+$	$-$	0	T
5	$+$	$-$	$-$	T
6	$+$	$-$	0	F
7	$+$	$-$	$+$	F
8	$+$	0	$+$	F
9	$+$	$+$	$+$	F

$$\delta_{\mathcal{C}_3^{(1)}} = [x_1 + 1 > 0 \ \wedge \ x_1 - 1 < 0 \ \wedge \ 2x_1^2 - 1 > 0 \ \wedge \ x_1 < 0]$$
$$\delta_{\mathcal{C}_4^{(1)}} = [x_1 + 1 > 0 \ \wedge \ x_1 - 1 < 0 \ \wedge \ 2x_1^2 - 1 = 0 \ \wedge \ x_1 < 0]$$
$$\delta_{\mathcal{C}_5^{(1)}} = [x_1 + 1 > 0 \ \wedge \ x_1 - 1 < 0 \ \wedge \ 2x_1^2 - 1 < 0]$$

- Solution formula: $\delta_{\mathcal{C}_3^{(1)}} \vee \delta_{\mathcal{C}_4^{(1)}} \vee \delta_{\mathcal{C}_5^{(1)}}$

QE algorithm:
Virtual Substitution method

Virtual substitution (VS)

- QE by VS for low degree formulas (w.r.t quantified variables)
  - Linear : Weispfenning et al, 1988
  - Quadratic : Loos et al, 1993
  - Cubic : Weispfenning 1993

- Implementation
  - REDLOG
  - SyNRAC

- Properties
  - Complexity: $\mathrm{O}(2^k)$: $k = \#$ of quantified variables
  - Output formula is in general large and redundant .
  - Degree violation (except linear case)
  - Formula simplification is important.

Virtual substitution (VS)

- Linear case
  - Linear first-order formula

$$Q_1 x_1 \cdots Q_n x_n \;\; \varphi(x_1, \ldots, x_n, a, b, \ldots)$$

where every atomic formula in φ is of the form

$$a_0 + a_1 x_1 + \cdots + a_n x_n \;\; \rho \;\; 0, \qquad (\rho = \{\geq, >, =, \neq\})$$

- Problem $\boxed{\exists x \varphi(x)}$

1. Change the innermost quantifier into $\exists x_i$: $\forall x \varphi(x) \Leftrightarrow \neg(\exists x \neg \varphi(x))$
2. Remove the innermost quantifier $\exists x_i$
3. Iterate until the quantifiers run out.

Virtual substitution (VS)

- QE problem: $\exists x \varphi(x)$ φ : linear
- VS algorithm for a linear formula

$$\exists x \varphi(x) \Leftrightarrow \bigvee_{t \in S} \varphi(x /\!/ t)$$

S : a set of terms, where each term $t \in S$ does not contain x

Then S is called an *elimination set* for $\exists x \varphi$ if the equivalence

$$\exists x \varphi(x) \Leftrightarrow \bigvee_{t \in S} \varphi(x /\!/ t)$$

holds, where $\varphi(x /\!/ t)$ is a formula equivalent to the expression $\varphi(x / t)$, which is obtained from φ by substituting t for x.

When $\exists x \varphi$ is linear, it has an elimination set.

Virtual substitution (VS)

FUJITSU

- QE problem: $\exists x \varphi(x)$ $\qquad \varphi$: linear
- VS algorithm for a linear formula

$$\boxed{\exists x \varphi(x) \Leftrightarrow \bigvee_{t \in S} \varphi(x // t)}$$

- Elimination set S
  - Set of atomic formulas in φ :

$$\psi = \{a_i x - b_i \ \ \rho_i \ \ 0 \mid i \in I, \rho_i \in \{=, \neq, \leq, <\}\}$$

  - An elimination set for $\exists x \varphi(x)$

$$S = \left\{ \frac{b_i}{a_i}, \frac{b_i}{a_i} \pm 1 \mid i \in I \right\} \cup \left\{ \frac{1}{2}\left(\frac{b_i}{a_i} + \frac{b_j}{a_j} \right) \mid i, j \in I, i \neq j \right\}$$

    - Other elimination sets are known.
      - Using smaller elimination sets helps increase algorithm's efficiency.

QE algorithm:
Sturm-Habicht sequence method

Sturm-Habicht sequence (SH)

- QE by Sturm-Habicht sequence for sign conditions of an univariate polynomial $f(x)$ (with parametric coefficients)
  - $\boxed{\forall x(f(x) > 0),\ \exists x(f(x) = 0),\ \exists x(f(x) < 0)}$
    - Gonzalez-Vega 1989, Yang et al. 1996
  - $\boxed{\forall x(x \geq 0 \to f(x) > 0)}$: sign definite condition (SDC)
    - Anai & Hara 1999, Iwane et al. 2013

- Implementation (SDC)
  - SyNRAC

- Properties
  - Complexity: $O(2^d), \quad d = deg_x(f(x))$
  - Output formula is in general large and redundant.
  - Formula simplification is important.

Sturm-Habicht sequence (SH)

FUJITSU

QE problem

- Sign definite condition: $\boxed{\forall x(x \geq 0 \rightarrow f(x) > 0)}$
  - SDC is equivalent to a condition that $f(x)$ has no real roots in $x \geq 0$ when the leading coefficient of $f(x)$ is positive:

A special QE algorithm using SH sequence for SDC

- Sturm-Habicht sequence of $f(x)$: SH(f)
  - Counts the number of real roots of $f(x)$ in an interval (like the Sturm sequence) through counting the number of sign changes of the sequence SH(f) at the endpoints of the interval
- SDC

$$f(x) \text{ has no real roots in } x \geq 0 \Leftrightarrow V_0(\mathrm{SH}(f)) - V_\infty(\mathrm{SH}(f)) = 0$$

- Combinatorial QE method
  - Enumeration of sign changes of the sequence $SH(f)$ having the above property

Sturm-Habicht sequence

- **Definition** Let f be a polynomial in $\mathbb{R}[x]$ with the degree n. The Sturm-Habicht sequence associated to f is defined as the sequence $\mathsf{SH}(f) := \{\mathsf{SH}_n(f), \ldots, \mathsf{SH}_0(f)\}$:
$$\mathsf{SH}_n = f, \quad \mathsf{SH}_{n-1} = \frac{df}{dx},$$
$$\mathsf{SH}_j = \delta_{n-j}\ \mathrm{Sres}_j(f, \tfrac{df}{dx})\ (j = 0, \ldots, n-2),$$
where $\delta_j = (-1)^{j(j+1)/2}$ and $\mathrm{Sres}_j(f, g)$ is a j-th subresultant which is defined as the determinant of the j-th Sylvester matrix of f and g.

Remark $\deg(\mathsf{SH}_k(f)) \leq k$

- **Definition** We define the sign of a real number is 1, 0, or -1 if the number is positive, zero, or negative, respectively.

Sturm-Habicht Sequence

FUJITSU

- **Definition** Let $A = \{a_m, \ldots, a_0\}$ be a finite sequence of real numbers. We define the number of sign variations $V(A)$ in the following rules:

  0: $\{+1, +1\}$, $\{-1, -1\}$

  1: $\{-1, +1\}$, $\{+1, -1\}$, $\{-1, 0, +1\}$, $\{+1, 0, -1\}$, $\{-1, 0, 0, +1\}$, $\ldots$,

  2: $\{+1, 0, +1\}, \{-1, 0, -1\}, \{+1, 0, 0, +1\}$, $\{-1, 0, 0, -1\}$, $\ldots$.

  Let $S(x) = \{S_n(x), S_{n-1}(x), \ldots, S_0(x)\}$ be a finite sequence of polynomials in $\mathbb{R}[x]$ and let α be a real number. We construct a sequence $\{h_s, \ldots, h_0\}$ of polynomials in $\mathbb{R}[x]$ obtained from $S(x)$ by deleting the polynomial identically zero. The number of sign variations $V_\alpha(S)$ is defined by $V(\{h_s(\alpha), \ldots, h_0(\alpha)\})$.

Example $V(+1, 0, 0, -1, +1, +1, -1) = V(+1, 0, 0, +1, -1) = 3$

Note This is different from that of Sturm sequence.

Real Root Counting by Sturm-Habicht Sequence

FUJITSU

- **Theorem** (González-Vega, et al. 1993) Let f be a polynomial in $\mathbb{R}[x]$ and α, β in $\mathbb{R} \cup \{-\infty, +\infty\}$ with $\alpha < \beta$ and $f(\alpha)f(\beta) \neq 0$. Then $V_\alpha(\mathsf{SH}(f)) - V_\beta(\mathsf{SH}(f))$ is equal to the number of real roots of $f(x)$ in the interval $[\alpha, \beta]$.

Example $f(x) = x^4 + 3x^2 + 5x + 1$

Sturm-Habicht sequence

$$
\begin{aligned}
\mathsf{SH}_4(f) &= x^4 + 3x^2 + 5x + 1 \\
\mathsf{SH}_3(f) &= 4x^3 + 6x + 5 \\
\mathsf{SH}_2(f) &= -24x^2 - 60x - 16 \\
\mathsf{SH}_1(f) &= -1020x - 420 \\
\mathsf{SH}_0(f) &= -8375
\end{aligned}
$$

Sturm sequence

$$
\begin{aligned}
F_0 &= x^4 + 3x^2 + 5x + 1, \\
F_1 &= 4x^3 + 6x + 5, \\
F_2 &= -3/2x^2 - 15/4x - 1, \\
F_3 &= -85/3x - 35/3, \\
F_4 &= -335/1156
\end{aligned}
$$

$$V_\infty(\mathsf{SH}(f)) = V_0(\mathsf{SH}(f)) = 1$$

Sturm-Habicht sequence (SH)

- Sign definite condition: $\boxed{\forall x(x \geq 0 \rightarrow f(x) > 0)}$
  - when the leading coefficient of $f(x)$ is positive:

$$f(x) \text{ has no real roots in } x \geq 0 \iff V_0(\mathsf{SH}(f)) - V_\infty(\mathsf{SH}(f)) = 0$$

- Notations

$$s_k \ : \ \text{sign of } \mathsf{SH}_k(f) \text{ at } x = \infty$$

$$c_k \ : \ \text{sign of } \mathsf{SH}_k(f) \text{ at } x = 0$$

- **Remark** Let $\mathsf{SH}_k(f) = a_k x^k + a_{k-1} x^{k-1} + \cdots + a_0$.

$s_k = 0$ is equivalent to $a_i = a_{k-1} = \cdots = a_0 = 0$
(i.e., $\mathsf{SH}_k(f)$ is identically zero).
$s_k > 0$ is equivalent to $(a_k > 0) \vee (a_k = 0 \wedge a_{k-1} > 0) \vee \cdots \vee (a_k = a_{k-1} = \cdots = a_1 = 0 \wedge a_0 > 0)$.
c_k is equivalent to the sign of a_0.
s_0 is equivalent to c_0. (degree of $\mathsf{SH}_0(f) \leq 0$)

Algorithm & Implementation (Anai & Hara 1999)

FUJITSU

Combinatorial QE algorithm for SDC

1. consider all the 3^{2n-2} (at most) possible sign conditions over s_k and c_k,
2. choose all sign conditions φ_n which satisfy $V_0(\mathsf{SH}(f)) - V_\infty(\mathsf{SH}(f)) = 0$,
3. construct semi-algebraic sets generated by coefficients of polynomials in $\mathsf{SH}(f)$ for each selected sign conditions and combine them as a union.

Implementation

- Since steps 1 and 2 are independent of an input polynomial, we can execute these steps beforehand and store the results in a database.

This greatly improves the total efficiency of the algorithm.

Algorithm & Implementation (Anai & Hara 1999)

FUJITSU

Example

- SDC:
  - $\forall x (x \geq 0 \to f(x) > 0)$ for $f(x) = x^2 + a_1 x + a_0$
- Sturm-Habicht sequence of $f(x)$

$$\mathrm{SH}_2 = f = x^2 + a_1 x + a_0$$
$$\mathrm{SH}_1 = df/dx = 2x + a_1$$
$$\mathrm{SH}_0 = a_1^2 - 4a_0$$

- Remark: $s_2 = s_1 = c_2 > 0, s_0 = c_0$
- Formula construction

$$\underbrace{a_1^2 - 4a_0}_{s_0} > 0 \wedge \underbrace{a_0}_{c_2} > 0 \wedge \underbrace{a_1}_{c_1} = 0$$
$$a_1^2 - 4a_0 = 0 \wedge a_0 > 0 \wedge a_1 = 0$$

Number of real roots in $x \geq 0$

s_2	s_1	s_0	c_2	c_1	c_0	#
+	+	+	+	+	+	0
+	+	+	+	0	+	2
+	+	+	+	-	+	2
+	+	0	+	+	0	0
+	+	0	+	0	0	0
+	+	0	+	-	0	1
+	+	-	+	+	-	0
+	+	-	+	0	-	0
+	+	-	+	-	-	0

For speeding up

The simplification of φ_n makes the algorithm and post-processing more efficient.

$$\mathrm{SH}_2 = f = x^2 + a_1 x + a_0$$
$$\mathrm{SH}_1 = df/dx = 2x + a_1$$
$$\mathrm{SH}_0 = a_1^2 - 4a_0$$

- Finds redundant sign conditions
  - The following conditions do not hold for all $a_1, a_0 \in \mathbb{R}$.

$$\overset{s_0}{\overset{\parallel}{a_1^2 - 4a_0}} > 0 \;\wedge\; \overset{c_2}{\overset{\parallel}{a_0}} > 0 \;\wedge\; \overset{c_1}{\overset{\parallel}{a_1}} = 0$$
$$a_1^2 - 4a_0 = 0 \;\wedge\; a_0 > 0 \;\wedge\; a_1 = 0$$

Necessary conditions for SDC

+	+	+	+	0	+	2	F
+	+	+	+	-	+	2	F
+	+	0	+	+	0	0	F
+	+	0	+	0	0	0	T
+	+	0	+	-	0	1	F
+	+	-	+	-	-	0	T

- Simplifies by the rules $\begin{cases} < \cup = \leftrightarrow \le, \\ \le \cup > \leftrightarrow \mathrm{Tr} \end{cases}$

Simplification by using Boolean function manipulation

  - Combinatorial optimization.
  - $(a_1^2 - 4a_0 < 0 \wedge a_0 > 0) \;\vee\; (a_0 > 0 \wedge a_1 > 0)$

$\cancel{\vee\; (a_1^2 - 4a_0 = 0 \wedge a_0 > 0 \wedge a_1 = 0)}$

Necessary Conditions for SDC (Iwane et al. 2013)

FUJITSU

s_2	s_1	s_0	c_2	c_1	c_0	#
+	+	+	+	0	+	2
+	+	0	+	0	0	0

- **Theorem** Let f be a polynomial in $\mathbb{R}[x]$ where the leading coefficient is nonzero and its degree is n, and let u be the smallest nonnegative integer k such that $s_k \neq 0$. When f satisfies $s_n > 0 \wedge c_n > 0$, the following conditions hold.

sign of leading coef. of f

sign of constant term of f

$$s_{n-1} > 0,\ c_u \neq 0,\ \underline{c_{n-1} = 0 \to c_{n-2} < 0},\ s_{n-2} = 0 \to s_{n-3} = \cdots = s_0 = 0,$$
$$s_k = 0 \to c_k = 0,\ (\forall k \in \{0, \ldots, n-2\}),$$
$$c_{k+2} \neq 0 \wedge c_{k+1} = 0 \to c_k \neq c_{k+2},\ (\forall k \in \mathcal{N} = \{u, \ldots, n-2\}),$$
$$c_k = c_{k+1} = 0 \wedge c_{k-1}c_{k+2}s_k s_{k+1} \neq 0 \to s_k s_{k+2} < 0,\ (\forall k \in \mathcal{N}),$$
$$c_k = \cdots = c_{k+m} = 0 \to s_{k+1} = \cdots = s_{k+m-1} = 0\ (\forall k \in \mathcal{N}, m > 1),$$
$$s_{k+2} = 0 \wedge s_{k+1} \neq 0 \to s_k \neq 0,\ (\forall k \in \mathcal{N}),$$
$$s_{k-1} \neq 0 \wedge s_k = \cdots = s_{k+m} = 0 \wedge s_{k+m+1} \neq 0 \to s_{k+m+2}^m s_{k-1} = \delta_{m+2} s_{k+m+1}^{m+1}$$
$$\wedge s_{k+m+2}^m c_{k-1} = \delta_{m+2} s_{k+m+1}^m c_{k+m+1},\ (\forall k \in \mathcal{N}, m \geq 0).$$

Necessary Conditions for SDC (Iwane et al. 2013)

- Most of conditions are obtained by utilizing Sturm-Habicht structure theorem (González-Vega, et al. 1993).

Theorem Let f be a polynomial in $\mathbb{R}[x]$ with degree n. Then for every $k \in \{1, \ldots, n-1\}$ such that $\mathsf{SH}_{k+1}(f)$ is regular and $\deg(\mathsf{SH}_k(f)) = r \leq k$ we have

(A) if $r < k-1$ then, $\mathsf{SH}_{k-1}(f) = \cdots = \mathsf{SH}_{r+1}(f) = 0$,

(B) if $r < k$ then, $\mathrm{lc}(\mathsf{SH}_{k+1}(f))^{k-r}\mathsf{SH}_r(f) = \delta_{k-r}\mathrm{lc}(\mathsf{SH}_k(f))^{k-r}\mathsf{SH}_k(f)$,

(C) $\mathrm{lc}(\mathsf{SH}_{k+1}(f))^{k-r+2}\mathsf{SH}_{r-1}(f) = \delta_{k-r+2}\mathrm{prem}(\mathsf{SH}_{k+1}(f), \mathsf{SH}_k(f))$,

where $\mathrm{lc}(g)$ is the leading coefficient of the polynomial g and $\mathrm{prem}(g, h)$ is a pseudo remainder of the polynomial g by the polynomial h defined as

$$\mathrm{prem}(g, h) = \mathrm{remainder}(\mathrm{lc}(h)^{\deg(g)-\deg(h)+1}g, h).$$

Boolean Algebra / Boolean Function

- Boolean Algebra: $(B, +, \cdot, ', 0, 1)$

  Defined on a set of two elements: $B = \{0, 1\}$, with rules for the three operations.

OR(+)

x	y	$x + y$
0	0	0
0	1	1
1	0	1
1	1	1

AND($\cdot$)

x	y	$x \cdot y$
0	0	0
0	1	0
1	0	0
1	1	1

NOT($'$)

x	x'
0	1
1	0

- Boolean expression is the combination of a finite number of Boolean variables and Boolean constants by means of the Boolean operations.
- A (completely specified) Boolean function with m variables is a function $f : B^m \to B$.
  - An incompletely specified Boolean function is a Boolean function which is defined over a subset of B^m.
  - An input combination for which the function is not specified is called a don't care.
    - we can choose a completely specified Boolean function by assigning 0 or 1 to each don't care.

Boolean Function Manipulation

- There are a number of Boolean expressions to represent a Boolean function
  - e.g., $(x+y)' = x' + y'$
  - Simplification of Boolean expressions.
    - Finding a Boolean expression which has relatively a small number of product terms

- Boolean expressions have wide range of application
  - Simplification of Boolean expressions directly corresponds to minimization of area of the designed circuit.

- ESPRESSO (Brayton *et al.*, 1984)
  - A heuristic method to simplify Boolean expressions.
  - http://diamond.gem.valpo.edu/~dhart/ece110/espresso/tutorial.html

Simplification based on Boolean Function Manipulation

- The sign of real number is three-valued, we need two Boolean variables to represent them.

sign	Boolean expression
0	$x'y'$
$+1$	xy'
-1	$x'y$

- Introduction of **don't cares** to simplify Boolean expression further.
  - xy
  - Sign conditions which do not satisfy the necessary conditions.
  - Sign conditions which satisfy $V_0(\mathsf{SH}(f)) - V_\infty(\mathsf{SH}(f)) < 0.$
    - The number of real roots is non-negative

ESPRESSO for quadratic poly. problem

FUJITSU

$f(s_0, c_1)$

s_0	c_2	c_1	T/F
+	+	+	T
+	+	0	DC
+	+	-	F
0	+	+	T
0	+	0	DC
0	+	-	F
-	+	+	T
-	+	0	T
-	+	-	T

INPUT FILE

```
.lib s0x s0y c1x c1y
.i 4
.o 1
.ob f0
1010 1
1000 2
0010 1
0000 2
0110 1
0100 1
0101 1
11-- 2
--11 2
.e
```

OUTPUT FILE

```
.lib s0x s0y c1x c1y
.i 4
.o 1
.ob f0
.p 2
-1-- 1
---0 1
.e
```

$\overbrace{01}^{s_0}\overbrace{00}^{c_1}$ $\overset{f(s_0, c_1)}{1}$

sign	Boolean expression
0	$00 = x'y'$
$+1$	$10 = xy'$
-1	$01 = x'y$

1 = true

2 = don't care

s_0	c_2	c_1
<	>	*
*	>	≧

Symbolic optimization by QE

Symbolic optimization accomplished by QE

FUJITSU

Advantages

- Exact (global) optimal value even for nonconvex case
- Parametric solving e.g., parametric optimum, feasible regions

Enabler for variants of parametric optimization

- Parametric constraint solving: *feasible region*
- (Multi-) parametric optimization: *optimal value function*
- Multi-objective optimization: *Pareto optimal front (trade-off line)*

Constraint solving by QE

Constraint solving by QE

- Find $x_1,\cdots,x_n$ s.t. $\{f_i(x_1,\cdots,x_n)\ \rho_i\ \ 0,\quad i=1,\cdots,s\}$ $\rho_i \in \{=,\geq,\neq\}$

$$\exists x_1\cdots\exists x_n\left(f_1(x_1,\cdots,x_n)\rho_1 0\wedge\cdots\wedge f_s(x_1,\cdots,x_n)\rho_s 0\right)$$

⇨ true (+ sample solution) / false

$$\exists x\exists y\ [\ 1<x<10\wedge y>0\wedge 6xy>0\ \wedge xy-2>0\wedge (xy-2)(2+4x-2xy)-6xy>0\ ]\ \xrightarrow{QE}$$

True
Sample point: (x,y)=(5,1)

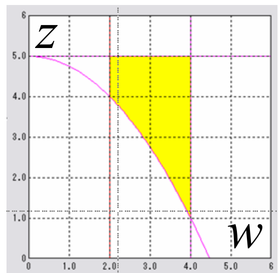
- Find the feasible regions of x_1, x_2 s.t. $\{f_i(x_1,\cdots,x_n)\ \rho_i\ \ 0,\quad i=1,\cdots,s\}$

$$\exists x_3\cdots\exists x_n\left(f_1(x_1,\cdots,x_n)\rho_1 0\wedge\cdots\wedge f_s(x_1,\cdots,x_n)\rho_s 0\right)$$

⇨ $\varphi(x_1,x_2)$: quantifier-free formula in x_1, x_2

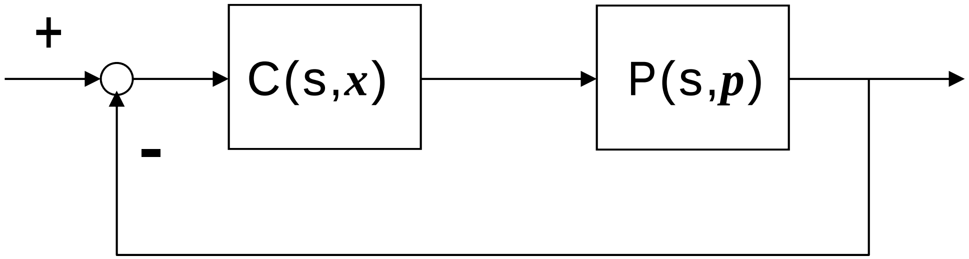
$$\exists x\ \exists y\ (4x - w^2 = 0 \wedge x - xy - z + 5 = 0 \wedge 1 \leqq x \leqq 4 \wedge 1 \leqq y \leqq 2)\ \xrightarrow{QE}$$

$[\ w-2 \geqq 0 \vee w+2 \leqq 0\ ]$
$\wedge\ w+4 \geqq 0\ \wedge\ w-4 \leqq 0\ \wedge$
$z-5 \leqq 0 \wedge 4z + w\char`^2 - 20 \geqq 0$



Stability of a Control system

FUJITSU



$$C(s,\mathbf{x}) = \frac{n_c(s,\mathbf{x})}{d_c(s,\mathbf{x})}$$

$$P(s,\mathbf{p}) = \frac{n_p(s,\mathbf{p})}{d_p(s,\mathbf{p})}$$

$\mathbf{p} = [p_1, \cdots, p_l]$: plant parameters

$\mathbf{x} = [x_1, \cdots, x_t]$: controller parameters

■ closed-loop characteristic polynomial

$$\boxed{f(s,\mathrm{x},\mathrm{p}) = n_p n_c + d_p d_c}$$

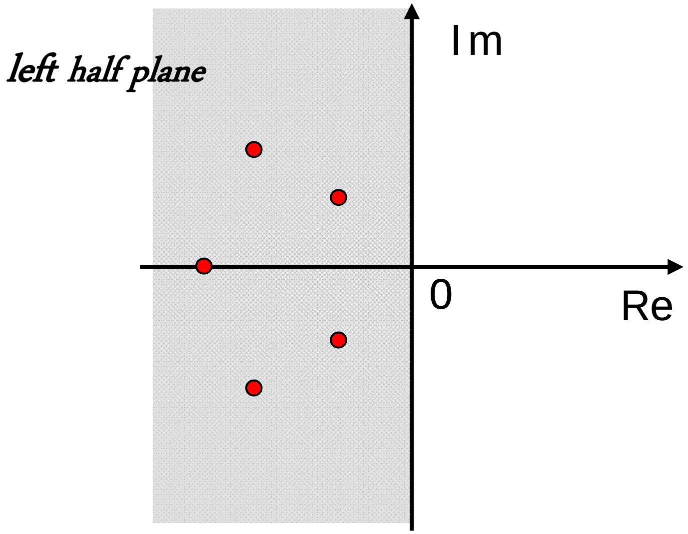
Stable? or
Unstable?

Stability of a Control system

■Hurwitz stability

$$f(s) = s^n + b_1 s^{n-1} + \cdots + b_{n-1}s + b_n$$

is stable $\Leftrightarrow$



P.Dorato et.al (1995)
M.Jirstrand (1996)

Routh-Hurwitz criterion

QE

$$f(s) = s^n + b_1 s^{n-1} + \cdots + b_{n-1}s + b_n$$

is stable $\Leftrightarrow$ $\Delta_1 > 0, \Delta_2 > 0, \ldots, \Delta_n > 0,$

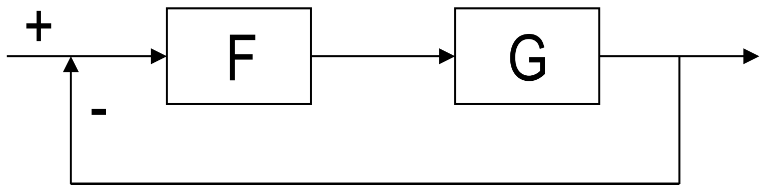
$$\Delta_k = \begin{vmatrix} b_1 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 \\ b_3 & b_2 & b_1 & 1 & 0 & 0 & \cdots & 0 \\ b_5 & b_4 & b_3 & b_2 & b_1 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ b_{2k-1} & b_{2k-2} & b_{2k-3} & b_{2k-4} & b_{2k-5} & b_{2k-6} & \cdots & b_k \end{vmatrix}.$$

***bi* : parametric**

Stability Analysis of Linear Systems

FUJITSU

- Unstable linear system :



$$G(s) = \frac{4}{s^2 - 2s + 2} \qquad poles = \{1 \pm i\}$$

$$F(s) = N\frac{s+b}{s+Nb} \qquad b > 0,\ \ 1 < N < 10 \text{ [phisical limitations]}$$

Resulting closed-loop system :

$$G_c(s) = \frac{GF}{1+GF} = \frac{4N(s+b)}{s^3 + (Nb-2)s^2 + (2+4N-2Nb)s + 6Nb}$$

Stability condition (Hurwitz criterion) :

$$6Nb > 0,\ \ Nb - 2 > 0,\ \ (Nb-2)(2+4N-2Nb) - 6Nb > 0$$

Stability Analysis of Linear Systems

FUJITSU

Q1 . For which values of b > 0, exists a value of N ∈ (1,10) s.t. the closed-loop system is stable ?

- $\exists$N $[1 < N < 10 \wedge b > 0 \wedge 6Nb > 0 \wedge Nb - 2 > 0 \wedge (Nb-2)(2+4N-2Nb) - 6Nb > 0]$

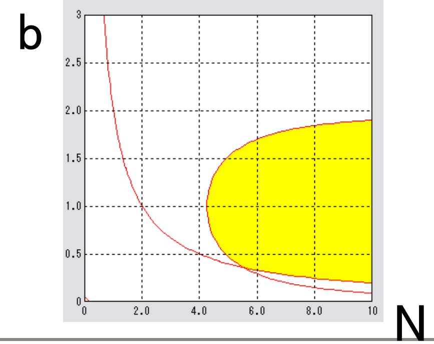
QE $[50b^2 - 100b + 21 < 0 \ \wedge \ b > 0]$ b∈(0.24,1.76)

- $\exists$b $[1 < N < 10 \wedge b > 0 \wedge \cdots]$

QE $[N^2 - 4N - 2 \ \wedge \ N > 1 \ \wedge \ N - 10 < 0]$ N∈(4.45,10.0)

- $\exists$b $\exists$N $[1 < N < 10 \wedge b > 0 \wedge \cdots]$

QE $true$



Stability Analysis of Linear Systems

FUJITSU

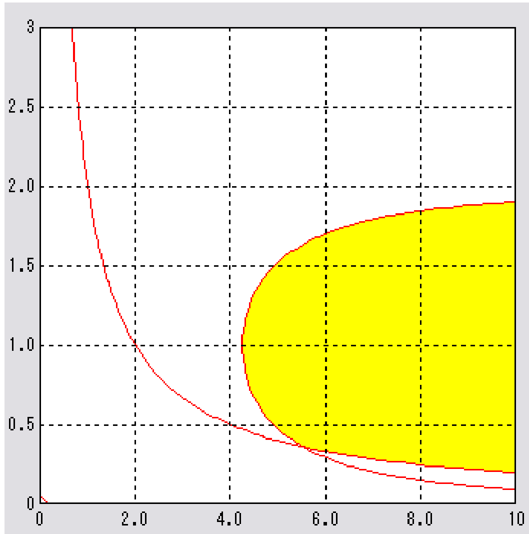
Q2 . For which values of b > 0, hold that for all N ∈ (5,10) the closed-loop system is stable ?

- $\forall \mathbf{N}\ [[5 < N < 10] \rightarrow [b > 0 \wedge 6Nb > 0 \wedge Nb - 2 > 0 \wedge (Nb-2)(2+4N-2Nb) - 6Nb > 0]]$

QE $$[25b^2 - 50b + 22 \le 0\,]$$

b ∈ (0.66, 1.34)

$$1+\frac{\sqrt{3}}{5},\ 1-\frac{\sqrt{3}}{5}$$



Solving optimization problems by QE

Optimization by using QE

FUJITSU

Problem

$$\text{Minimize} \quad f(\boldsymbol{x})$$
$$\text{subject to} \quad C(\boldsymbol{x})$$

QE

$$\exists \boldsymbol{x} \ (y = f(\boldsymbol{x}) \land C(\boldsymbol{x}))$$

QE ⇒ feasible region for y

Example

$$\begin{aligned} &\text{Minimize} && -x_1 - x_2 \\ &\text{subject to} && x_1 \geq 0, x_2 \geq 0, \\ & && x_1^2 + x_2^2 \leq 1 \end{aligned}$$

QE

$$\exists x_1 \exists x_2 \left(\begin{array}{l} y = -x_1 - x_2 \land \\ x_1 \geq 0 \land x_2 \geq 0 \land \\ x_1^2 + x_2^2 \leq 1 \end{array} \right)$$

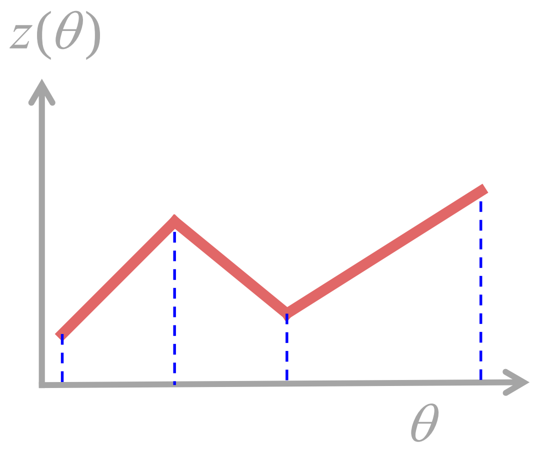
QE ⇒ $-\sqrt{2} \leq y \leq 0$

Parametric optimization

- **Given**:
  - an objective function to optimize
  - a vector of constraints
  - a vector of parameters

$$
\begin{aligned}
z(\theta) &= \min_{\mathbf{x}} f(\mathbf{x}, \theta) \\
\text{s.t. } & g(\mathbf{x}, \theta) \geq 0 \\
& \mathbf{x} \in \Re^n \\
& \theta \in \Re^s
\end{aligned}
$$

- **Obtain**:
  - the performance criterion (and the optimization variables) as a function of the parameters
  - the regions in the space of parameters where these functions remain valid



Obtain optimal solution as a function of parameters

Parametric optimization by using QE

Problem

$$\begin{array}{ll} \text{Minimize} & f(\boldsymbol{x}, \boldsymbol{\theta}) \\ \text{subject to} & C(\boldsymbol{x}, \boldsymbol{\theta}) \end{array}$$

QE

$$\exists \boldsymbol{x} \ \ (y = f(\boldsymbol{x}, \boldsymbol{\theta}) \wedge C(\boldsymbol{x}, \boldsymbol{\theta}))$$

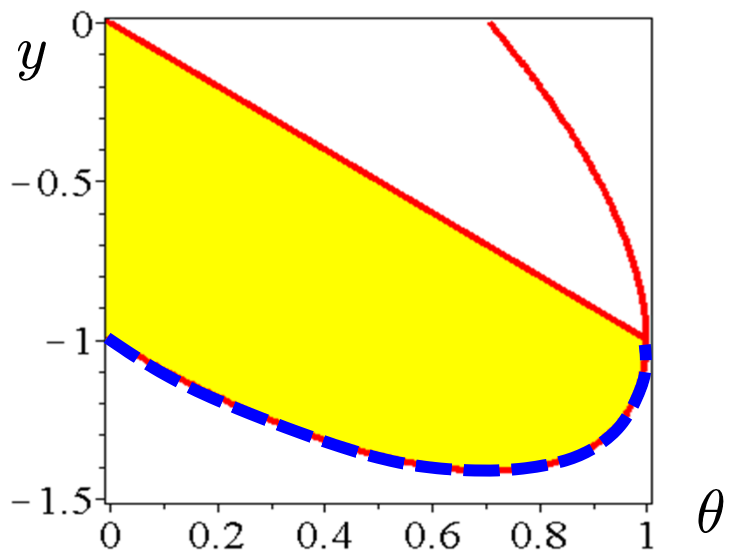
QE → feasible region for $(\boldsymbol{\theta}, y)$

Example

$$\begin{array}{ll} \text{Minimize} & -x_1 - \theta \\ \text{subject to} & x_1 \geq 0, \theta \geq 0, \\ & x_1^2 + \theta^2 \leq 1 \end{array}$$

$$\exists x_1 \left(\begin{array}{l} y = -x_1 - \theta \wedge \\ x_1 \geq 0 \wedge \theta \geq 0 \wedge \\ x_1^2 + \theta^2 \leq 1 \end{array} \right)$$

QE →



$$y^2 + 2\theta y + 2\theta^2 \leq 1 \wedge$$
$$0 \leq \theta \leq 1 \wedge y \leq \theta$$

Optimization by QE

Parametric optimization

Example

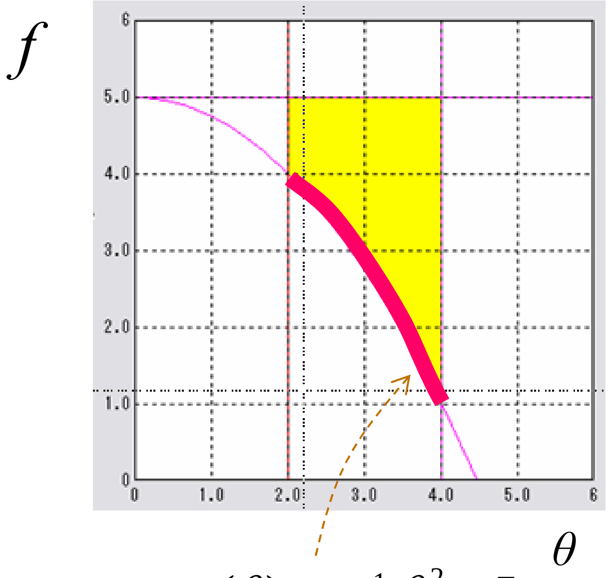
$$z_{\min}(\theta) = \min_{\mathbf{x}} f(\mathbf{x}, \theta)$$

$$\text{s.t.}\quad f(\mathbf{x}, \theta) = x_1 - x_1 x_2 + 5$$

$$4x_1 - \theta^2 = 0$$

$$1 \le x_1 \le 4$$

$$1 \le x_2 \le 2$$



$$z_{\min}(\theta) = -\tfrac{1}{4}\theta^2 + 5$$

optimal solution as a function of parameter

QE problem

$\exists x \; \exists y$

$(x_1 - x_1 x_2 + 5 = z \wedge 4 x_1 - \theta^2 = 0$

$\wedge 1 \leqq x_1 \leqq 4 \wedge 1 \leqq x_2 \leqq 2)$

$\xrightarrow{QE}$

$[\theta - 2 \geqq 0 \vee \theta + 2 \leqq 0\,]$

$\wedge\ \theta + 4 \geqq 0 \ \wedge \theta - 4 \leqq 0 \ \wedge$

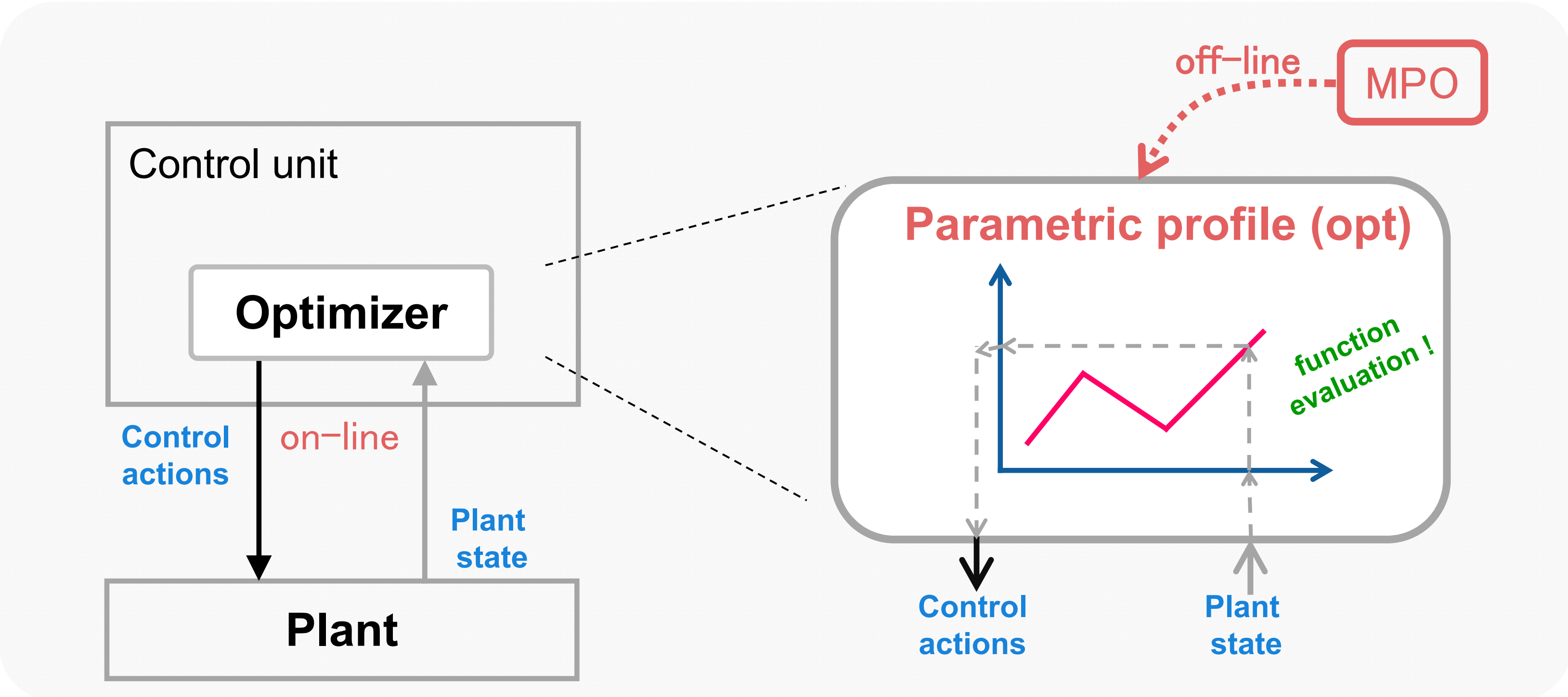
$z - 5 \leqq 0 \wedge 4z + \theta^2 - 20 \geqq 0$

Feasible region of z- θ

Multi-parametric optimization in control

FUJITSU

- Applications
  - Bi-level / Hierarchical programming
  - Optimization under uncertainty
  - Model predictive control
  - On-line control and optimization of
    - chemical, biomedical, automotive systems



Multi-objective optimization by QE

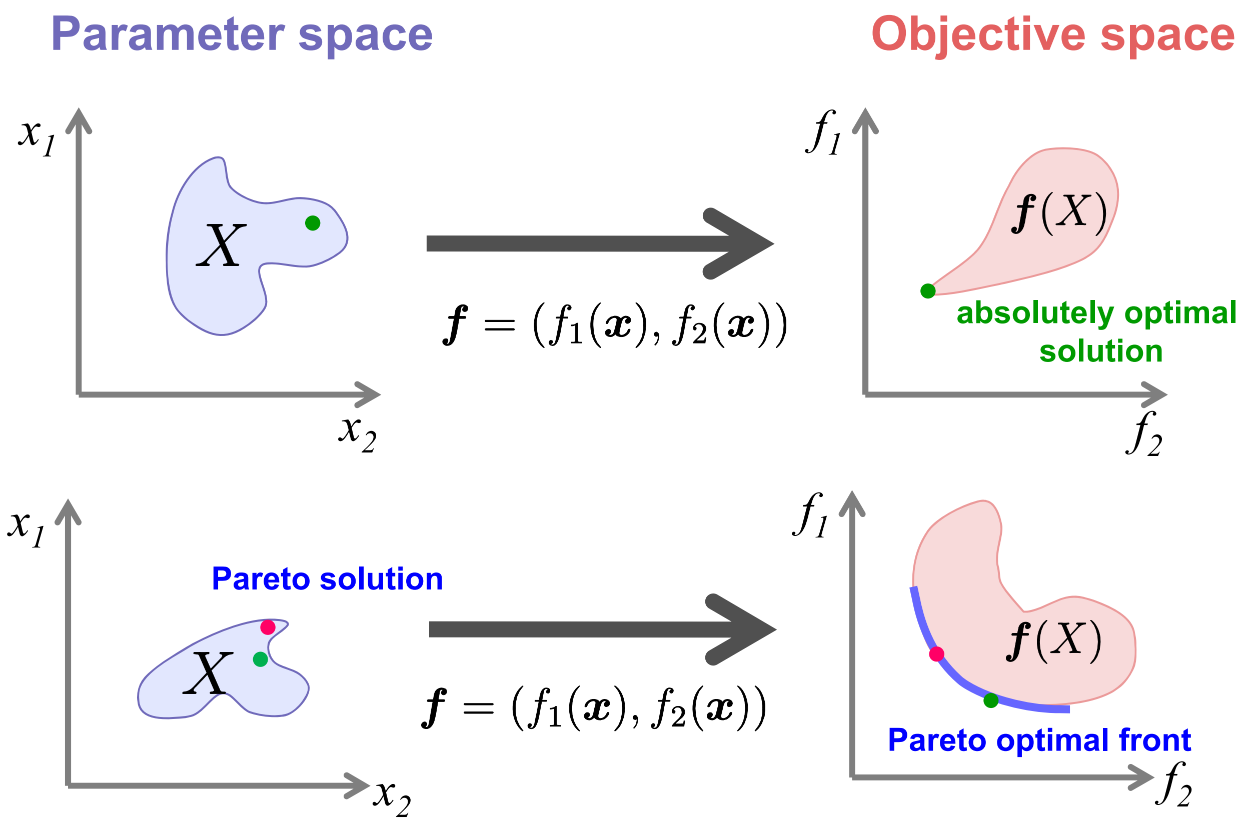
Multi-objective optimization (MOO)

FUJITSU

■ Problem

$$\begin{aligned} &\text{minimize} \quad \boldsymbol{f} = (f_1(\boldsymbol{x}), \ldots, f_r(\boldsymbol{x})) \\ &\text{subject to} \quad \boldsymbol{x} \in X \subset \mathbb{R}^n \end{aligned}$$

■ Solution



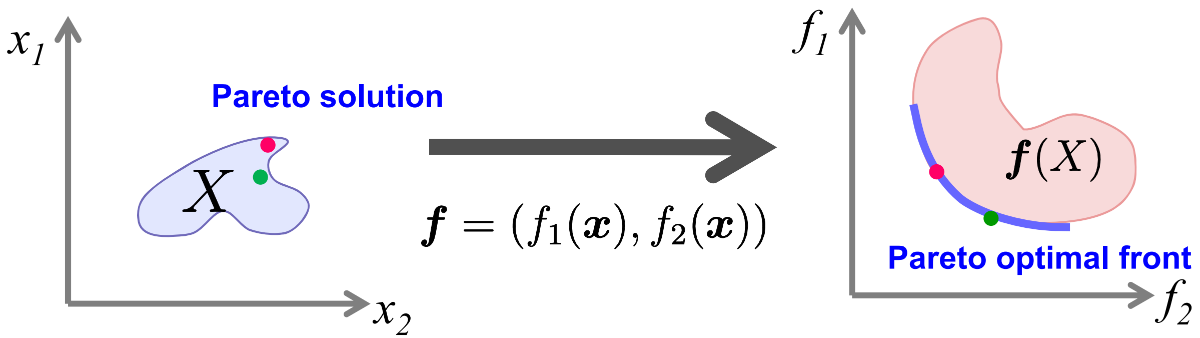
Multi-objective optimization by QE

- Problem

$$\begin{array}{ll} \text{minimize} & \boldsymbol{f} = (f_1(\boldsymbol{x}), \ldots, f_r(\boldsymbol{x})) \\ \text{subject to} & \boldsymbol{x} \in X \subset \mathbb{R}^n \end{array}$$

- QE problem

input	output
$\exists \boldsymbol{x}\, (\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}) \wedge C(\boldsymbol{x}))$	Feasible region $F(\boldsymbol{y})$
$F(\boldsymbol{y}) \wedge \underline{\neg \exists \boldsymbol{y}' (F(\boldsymbol{y}') \wedge \boldsymbol{y}' \le \boldsymbol{y})}$	Pareto set $P(\boldsymbol{y})$
$\exists \boldsymbol{y}\, (\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}) \wedge C(\boldsymbol{x}) \wedge P(\boldsymbol{y}))$	Optimizer $\boldsymbol{x}$



Multi-objective optimization by QE

FUJITSU

Problem

$$\begin{array}{ll} \text{Minimize} & \boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), \ldots, f_m(\boldsymbol{x})) \\ \text{subject to} & C(\boldsymbol{x}) \end{array}$$

QE

$$\exists \boldsymbol{x} \ \ (\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}) \wedge C(\boldsymbol{x}))$$

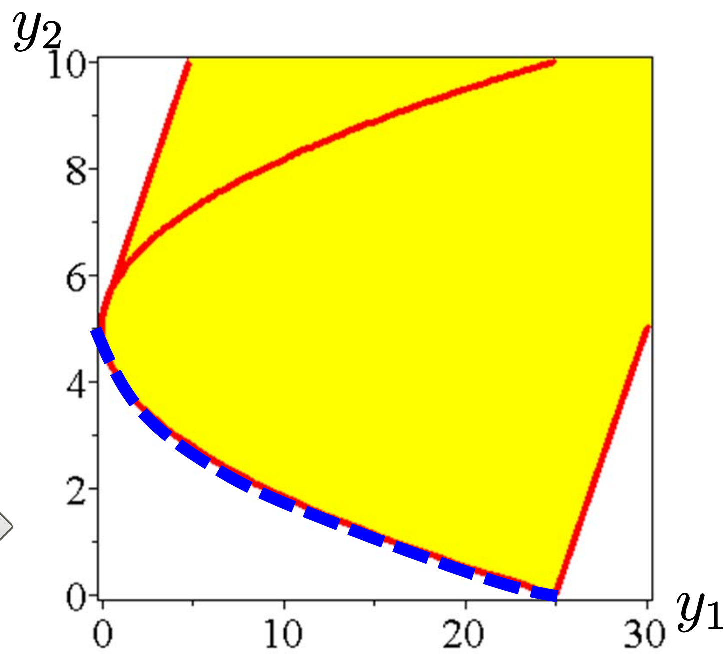
QE $\Rightarrow$ feasible region for $\boldsymbol{y}$

Example

$$\begin{array}{ll} \text{Minimize} & (x_1^2 + x_2^2, x_2^2 - x_1 + 5) \\ \text{subject to} & -5 \leq x_1, x_2 \leq 5 \end{array}$$

$$\begin{array}{l} \exists x_1 \\ \exists x_2 \end{array} \left(\begin{array}{l} y_1 = x_1^2 + x_2^2 \ \wedge \\ y_2 = x_2^2 - x_1 + 5 \ \wedge \\ -5 \leq x_1, x_2 \leq 5 \end{array} \right)$$

QE $\Rightarrow$



Example: Multi-objective optimization

FUJITSU

Problem

minimize $y_1 = f_1(x)$
$y_2 = f_2(x)$
$\cdots$

subject to $C(x)$

Solution

"Pareto set"

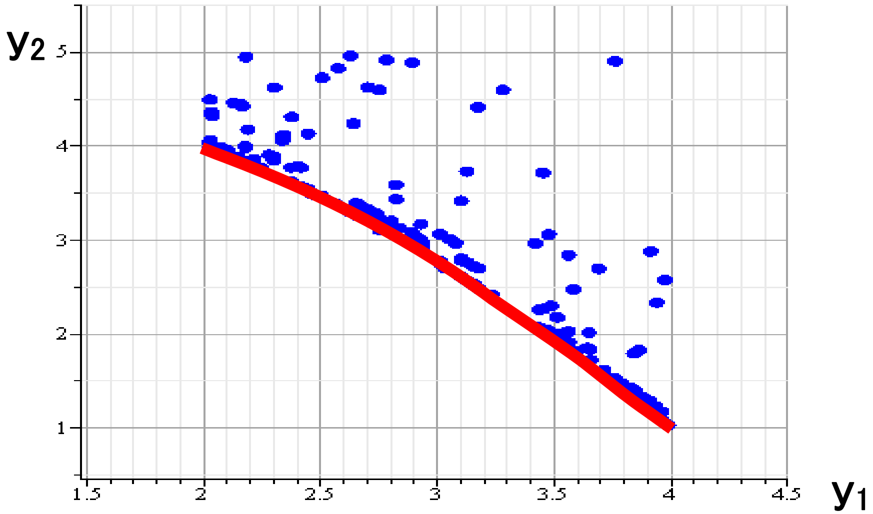
P = { all "minimal" y w.r.t $\leq$ }
$y \leq y'$ iff $\forall i \quad y_i \leq y_i'$

Toy Example

minimize $y_1 = 2\sqrt{x_1}$
$y_2 = x_1 - x_1 x_2 + 5$

subject to $1 \leqq x_1 \leqq 4,$
$1 \leqq x_2 \leqq 2$

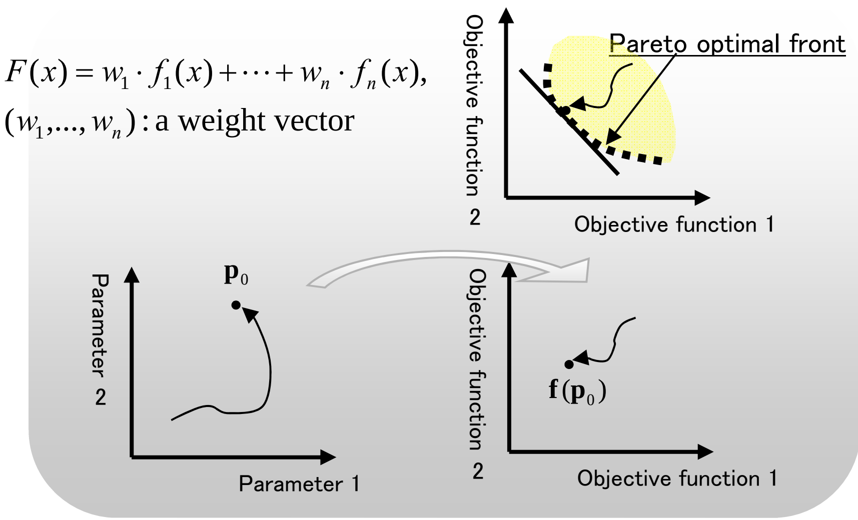
Solution



Existing numerical methods for MOO

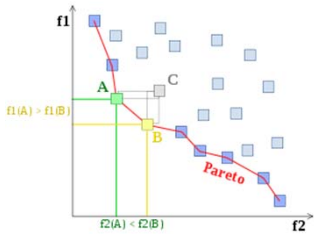
- **Using single-objective optimization**
  - Weighted sum strategy
  - Norm minimization
  - E-constraint method

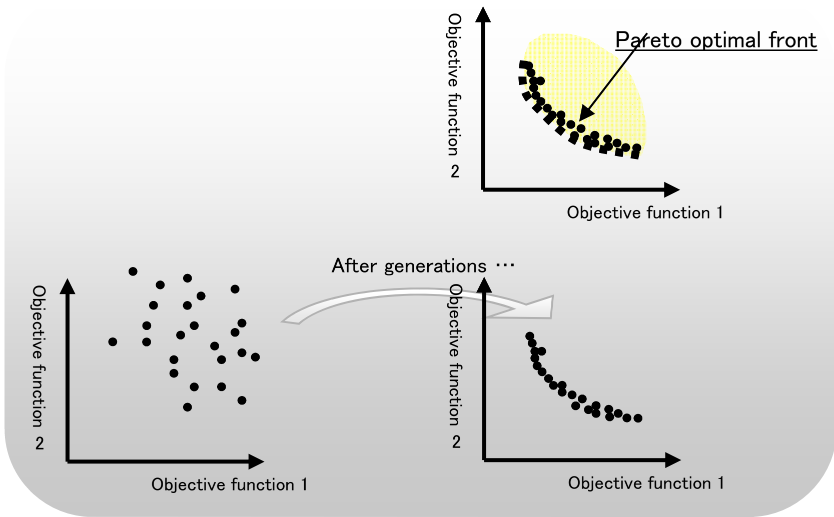
$$F(x) = w_1 \cdot f_1(x) + \cdots + w_n \cdot f_n(x),$$
$$(w_1, \ldots, w_n): \text{a weight vector}$$



- **Pareto analysis**
  - Normal boundary intersection

- **Using metaheuristic algorithms**
  - Evolutionary algorithms
  - Particle swarm optimization





Comparison: Symbolic vs. Numeric

Optimization Problem

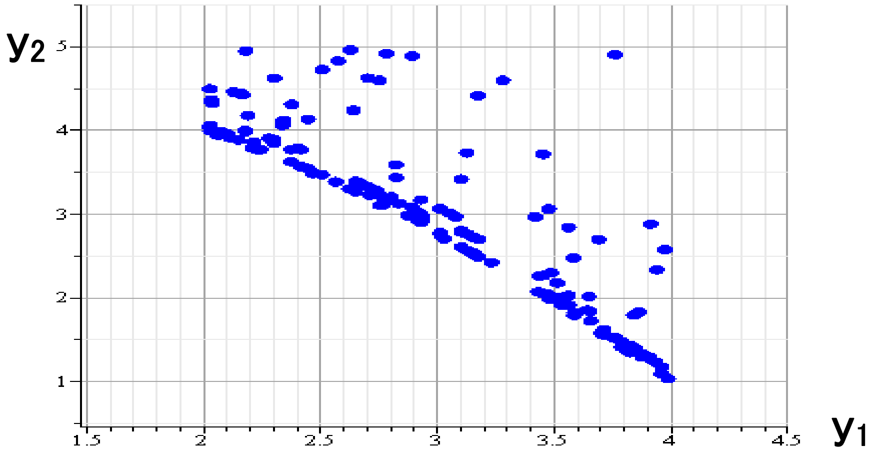
minimize $y_1 = 2\sqrt{x_1}$

$y_2 = x_1 - x_1 x_2 + 5$

subject to $1 \leqq x_1 \leqq 4,$

$1 \leqq x_2 \leqq 2$

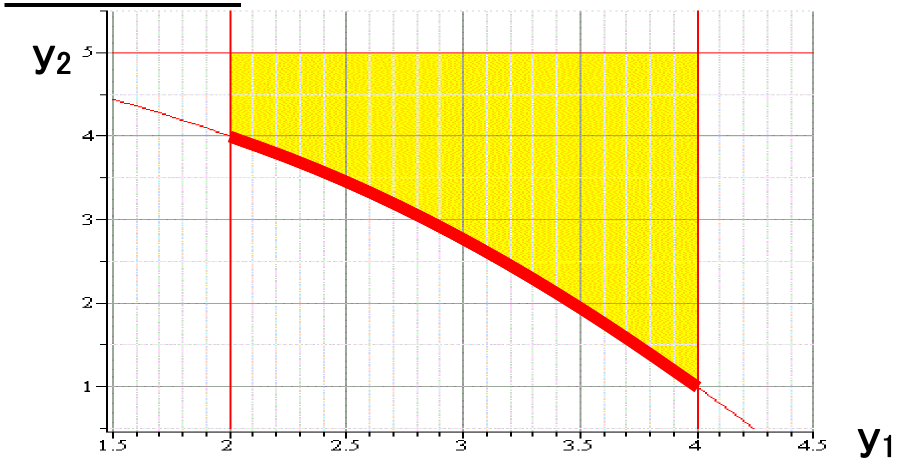
Solution



QE Problem

$$\exists x_1 \exists x_2 \; (y_1 = 2\sqrt{x_1} \;\wedge\; y_2 = x_1 - x_1 x_2 + 5 \;\wedge\; 1 \leqq x_1 \leqq 4 \;\wedge\; 1 \leqq x_2 \leqq 2)$$

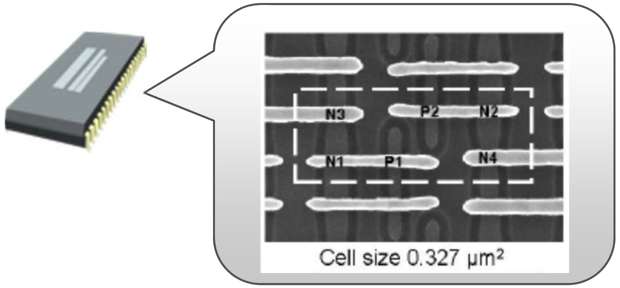
Solution

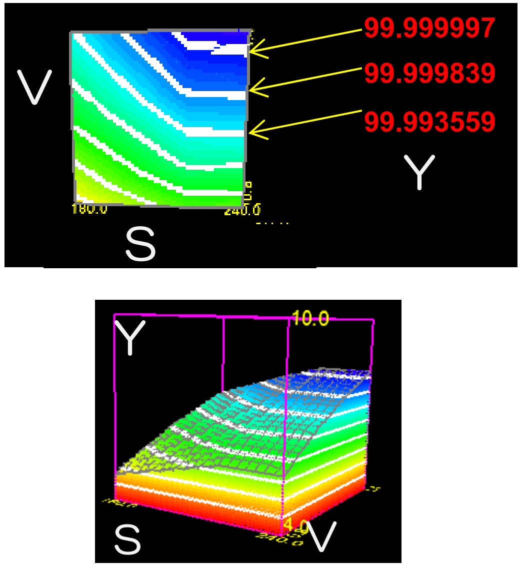


Applications : MOO by QE

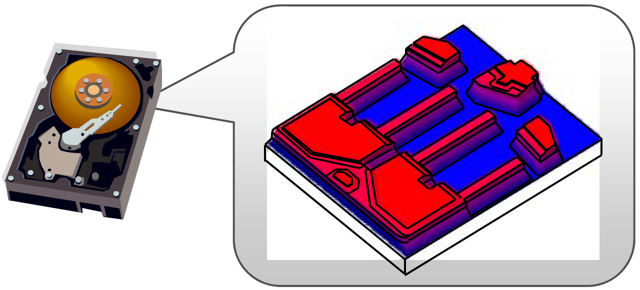
FUJITSU

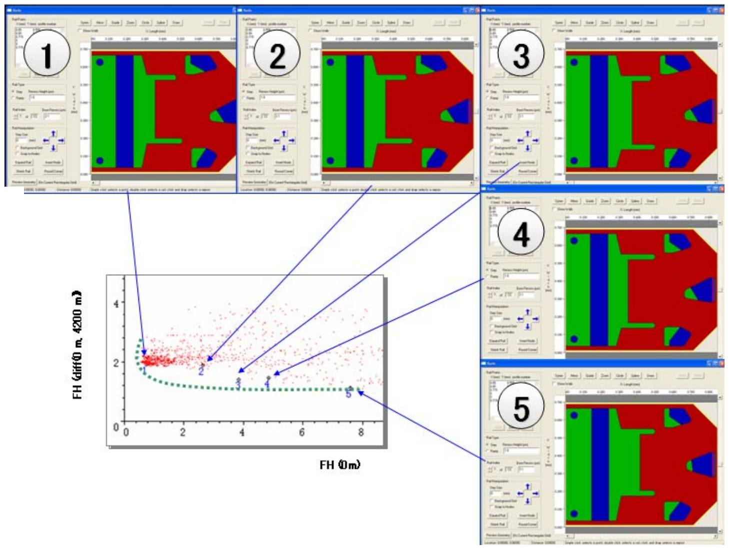
- SRAM shape optimization
  - Objective functions:
    - min (- Yield rate(Y), Voltage(V), Size(S))





- HDD (head) shape design
  - Objective functions:
    - Stability of Flight-height, attitude(Roll, Pitch, Yaw)



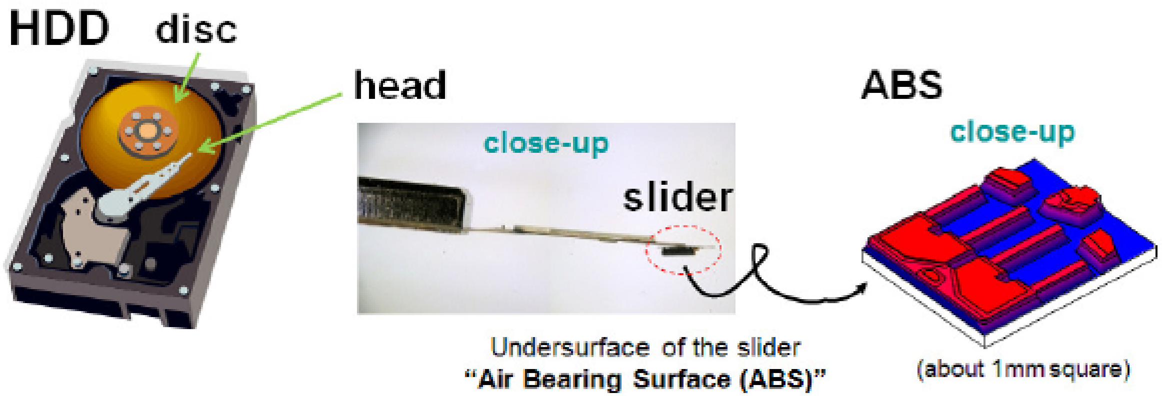


Multi-objective optimization by QE

Optimal shape design of Air Bearing Surface of HDD

Shape design of Air Bearing Surface of HDD

ABS (Air Bearing Surface)



- The disc is rapidly spinning and the ABS surfacing over the disc due to air current.

Problem: Find the optimal shape of ABS s.t.

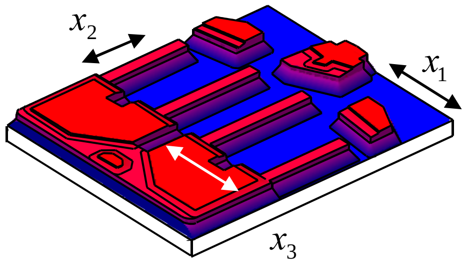
- flight height of the ABS from the rapidly spinning disc is close to a target value
- attitude (Roll, Pitch, Yaw) of the ABS is stable
- …

Shape design of Air Bearing Surface of HDD

FUJITSU

- Design problem: Find the optimal shape of ABS s.t.
  - 1) flight height of the ABS from the rapidly spinning disc is close to a target value
  - 2) attitude (Roll, Pitch, Yaw) of the ABS is stable
  - …
- Simulation

Input $x = (x_1, x_1, \ldots, x_n)$ **Output** **Objective functions**



Surfacing simulation (fluid dynamics)

h : flight height,
A: (roll, pitch, yaw,)
…

$f_1(x) = |h - \tilde{h}|^2$
$f_2(x) = \cdots$
…

- Optimization problem (Yanami et al., 2009)
  - Response surface methodology
    - Modeling of the objective functions $f_1(x), f_2(x), \ldots$ from a certain number of simulation results.
  - Multi-objective optimization

$$\begin{cases} \text{minimize} & \boldsymbol{f} = (f_1(\boldsymbol{x}), \ldots, f_r(\boldsymbol{x})) \\ \text{subject to} & \boldsymbol{x} \in X \subset \mathbb{R}^n \end{cases}$$

Shape design of Air Bearing Surface of HDD

FUJITSU

Our real problem

- Shape parameters : $x_1, \ldots, x_8$
- Objective functions: $f_1(x), f_2(x)$

Response surface construction

- Data set of $x_1, \ldots, x_8, f_1(x), f_2(x)$ for 553 different shapes
- Polynomial model of f_1, f_2 :
  - Linear regression (R^2 > 0.95)

f1= x1*1.56834776 +x2*0.804244896 +x3*21.32342295 +x4*(-7.71943013) +x5*(-4.262328228) +x6*12.95499327 +x7*0.29533099 +x8*(-1.142721635) +(-7.809437853);

f2:= x1*0.37323681 + x2*1.313718858 + x3*7.296804764 + x4*(-3.214736241) + x5*10.32056396 + x6*6.068769576 + x7*(-2.987556175) + x8*6.86732377 + (-4.344757609)

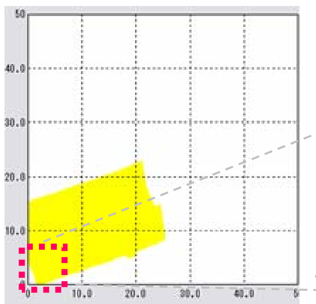
- Multi-objective optimization

$$\begin{cases} \min f_1(x), f_2(x) \\ \text{s.t. } x_1, x_2 \in X \subset R\ 2 \end{cases}$$

MOO by QE

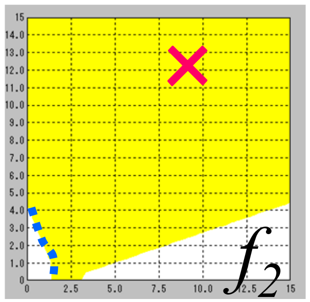
- Feasible region of f_1, f_2
  - Pareto front
  - Solution by a numerical method

f_1



f_2

f_1



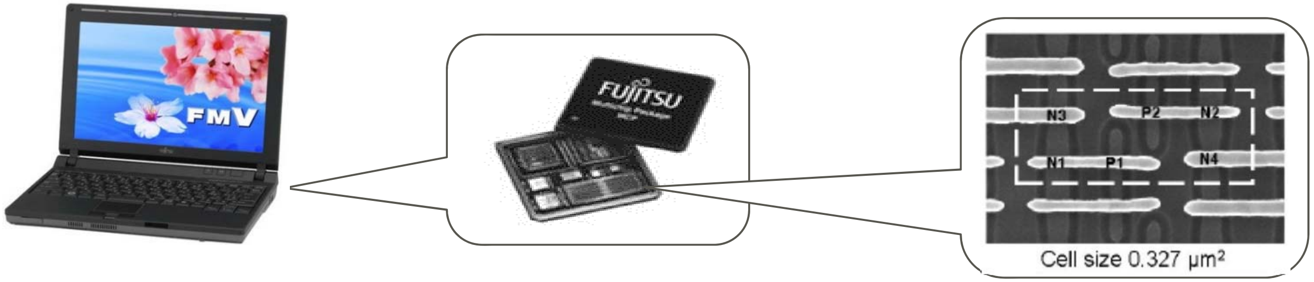
f_2

Parametric optimization by QE

Optimal shape design of SRAM

shape design of an SRAM cell

- Static random-access memory (SRAM) cell



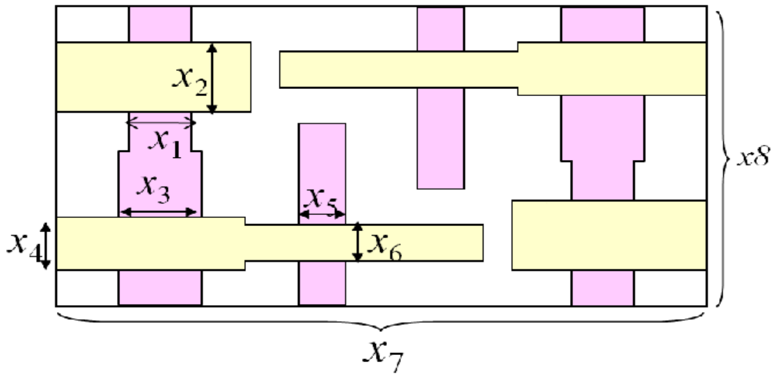
- Problem: Find the optimal shape of an SRAM cell with
  - Smaller cell size (area)

$$\theta = x_7 x_8$$

  - Lower fail rate

$$f(\boldsymbol{x}) = a_0 + a_1 x_1 + \cdots + a_6 x_6$$

$$(a_i\text{: constants})$$



shape design of an SRAM cell

FUJITSU

Multi-optimization problems

$$
\begin{aligned}
\min\ & f(\boldsymbol{x}) = a_0 + a_1 x_1 + \cdots + a_6 x_6 \quad (a_i\text{: constants})\\
& \theta = x_7 x_8\\
\text{s.t.}\quad & x_7 = 1420 + 250x_3 + 250x_5\\
& x_8 = 800 + 40x_2 + 40x_4\\
& x_4 + x_6 \geq 2x_2\\
& 0 \leq x_i \leq 1 \ (i = 1, \ldots, 6)
\end{aligned}
$$

MOO by QE

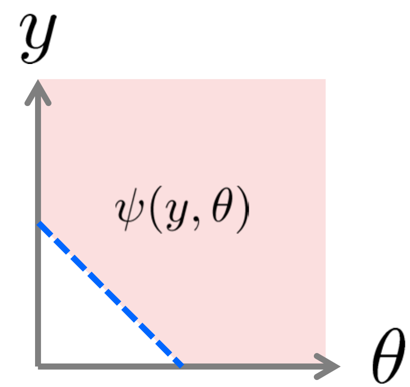
QE problem

$$
\begin{aligned}
\exists x_1 \cdots \exists x_8 \quad & (y = a_0 + a_1 x_1 + \cdots + a_6 x_6 \ \wedge\\
& \theta = x_7 x_8 \ \wedge\\
& x_7 = 1420 + 250x_3 + 250x_5 \ \wedge\\
& x_8 = 800 + 40x_2 + 40x_4 \ \wedge\\
& x_4 + x_6 \geq 2x_2 \ \wedge\\
& 0 \leq x_i \leq 1 \ (i = 1, \ldots, 6))
\end{aligned}
$$

an equivalent quantifier-free formula

$\psi(y, \theta)$

more than 1 hour….



◆ VS algorithm: REDUCE 07-oct-10 / REDLOG 1.60 GHz CPU / 8.0 GB memory

shape design of an SRAM cell

QE problem

$$\exists x_1 \cdots \exists x_8 \quad (y = a_0 + a_1 x_1 + \cdots + a_6 x_6 \wedge \theta = x_7 x_8 \wedge x_7 = 1420 + 250x_3 + 250x_5 \wedge x_8 = 800 + 40x_2 + 40x_4 \wedge x_4 + x_6 \geq 2x_2 \wedge 0 \leq x_i \leq 1 \; (i = 1, \ldots, 6))$$

more than 1 hour….

QE VS (linear, quadratic) ✖ $\psi(y, \theta)$

$$\exists x_1 \cdots \exists x_6 \quad (y = a_0 + a_1 x_1 + \cdots + a_6 x_6 \wedge \theta = x_7 x_8 \wedge x_7 = 1420 + 250x_3 + 250x_5 \wedge x_8 = 800 + 40x_2 + 40x_4 \wedge x_4 + x_6 \geq 2x_2 \wedge 0 \leq x_i \leq 1 \; (i = 1, \ldots, 6))$$

1 sec

QE VS (linear) ○ $\psi'(y, x_7, x_8, \theta)$

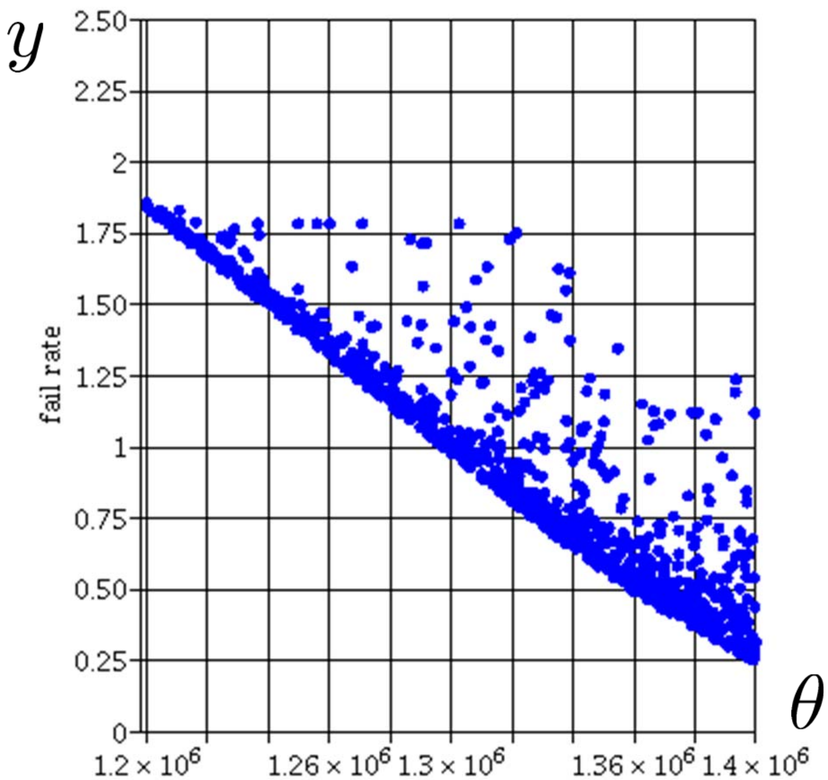
Symbolic-Numeric optimization (Iwane et al., 2011)

- We utilize $\psi'(y, x_7, x_8, \theta)$ as a search area in a numerical optimization approach
  - **better approximation** of an optimal values than ordinal numeric approaches
  - more **effective** than symbolic approach

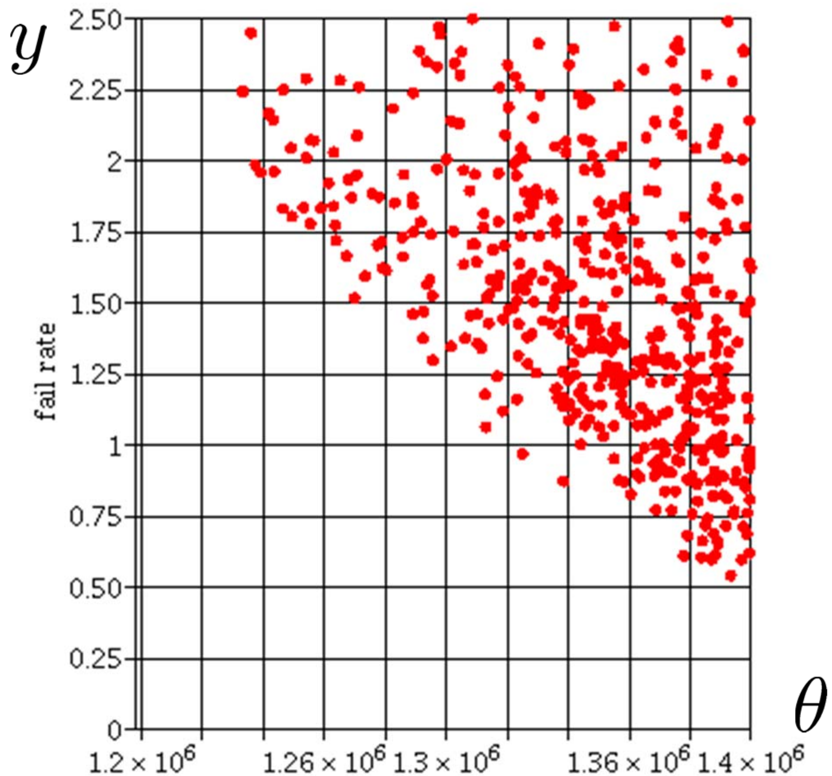
MOO results comparison

FUJITSU

Symbolic-Numeric



Numeric



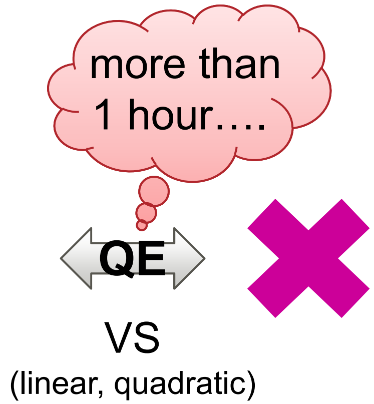
modeFrontier 4.2.1 / Particle Swarm Optimization (PSO) / 2000 samples
1.60 GHz CPU / 8.0 GB memory

shape design of an SRAM cell

FUJITSU

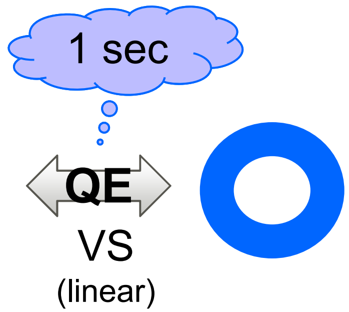
■ QE problem

$$
\begin{aligned}
\exists x_1 \cdots \exists x_8 \quad (& y = a_0 + a_1 x_1 + \cdots + a_6 x_6 \;\wedge \\
& \theta = x_7 x_8 \;\wedge \\
& x_7 = 1420 + 250 x_3 + 250 x_5 \;\wedge \\
& x_8 = 800 + 40 x_2 + 40 x_4 \;\wedge \\
& x_4 + x_6 \geq 2 x_2 \;\wedge \\
& 0 \leq x_i \leq 1 \; (i = 1, \ldots, 6))
\end{aligned}
$$



$$\psi(y, \theta)$$

$$
\begin{aligned}
\exists x_1 \cdots \exists x_6 \quad (& y = a_0 + a_1 x_1 + \cdots + a_6 x_6 \;\wedge \\
& \theta = x_7 x_8 \;\wedge \\
& x_7 = 1420 + 250 x_3 + 250 x_5 \;\wedge \\
& x_8 = 800 + 40 x_2 + 40 x_4 \;\wedge \\
& x_4 + x_6 \geq 2 x_2 \;\wedge \\
& 0 \leq x_i \leq 1 \; (i = 1, \ldots, 6))
\end{aligned}
$$



$$\psi'(y, x_7, x_8, \theta)$$

■ Symbolic-Numeric optimization (Iwane et al., 2011)

- We utilize $\psi'(y, x_7, x_8, \theta)$ as a search area in a numerical optimization approach
  - **better approximation** of an optimal values than ordinal numeric approaches
  - more **effective** than symbolic approach

INTERPLAY:
Quantifier elimination algorithms and applications in Control

Brief History of QE algorithms

FUJITSU

1930	■ Tarski proved QE is possible over ***R***
1951	■ Tarski proposed a QE algorithm over ***R*** ■ Computational complexity cannot be bound by any tower of exponentials
1975	■ Collins made a breakthrough ■ QE by Cylindrical Algebraic Decomposition (CAD) ■ Computational complexity down to doub... number of variables
1988	■ QE computation is proved to be ... ■ Doubly exponential in worst case (D...)
1990	■ QEPCAD: First CAD-based QE imp... (Hong)

Note !
Such special classes have close relations with Industrial applications !

- 1980's Different approaches
  - QE algorithms for a restricted class of input
    - QE for up to linear/quartic formulas, Positive polynomial condition

QE and control applications

- Purely symbolic and algebraic approaches have several “practical size” applications in control.

- Special algorithms by exploiting the structure of the problems have been successfully applied.

- Still we need to solve larger size problems in a reasonable amount of time.

- We employ validated numerical methods (interval arithmetic)
  - Symbolic-Numeric CAD computation (still exact)
    - Speeding up QE algorithm based on CAD
  - Approximated quantified constraint solving
    - Obtaining approximated feasible regions (with guarantee)

General QE: QE by CAD

- For speeding-up QE by CAD
  - QE by a partial CAD (Hong, Collins)
  - “Projection Operator”
    - Collins’ projection operator (the original)
    - Hong’s projection operator (improved Collins’)
    - McCallum’s projection operator
    - Brown-McCallum projection operator (improved McCallum’s)
    - “special purpose” projection operators:
      - Collins-McCallum: equational constraints, Seidl-Sturm: generic CAD,
      - Strzebo´nski : solving strict systems, Anai, Parrilo: solving SDP
  - Lifting
    - Full-dimensional cell
    - Lifting with symbolic-numeric computations (SN-CAD)

General QE: QE by CAD

QE by CAD

- Early period on QE applications (stability analysis in control) …

> ➢P.Dorato & I.Sakamaki IFAC Rocon'03, 2003
>
> While commercial software is now available for the application of symbolic QE for the design of robust feedback systems, only problems of limited complexity can be solved. Of course, super-computer systems can extend the level of complexity, But the level is likely to saturate on problems where **the order of combined plant and compensator is greater than 5 or 6.**
>
> ➢P.Dorato et.al. UNM Technical Report : EECE95-007, 1995
>
> Our Experience indicates that QEPCAD can always solve, in a few seconds on a large workstation, **most textbook examples**. It can also solve some significantly harder problems and **a few nontrivial problems**.

➡ **Particular subclasses : Special QE algorithms !**

- Still many QE problems requires general QE algorithms by CAD.
  - Use CAD properties for a given problems
    - Optimization problems:
      - ☐ Semi-definite programming (Anai & Parrilo 2003)
      - ☐ Polynomial optimization problems (Iwane et.al. 2013)

Special QE algorithms

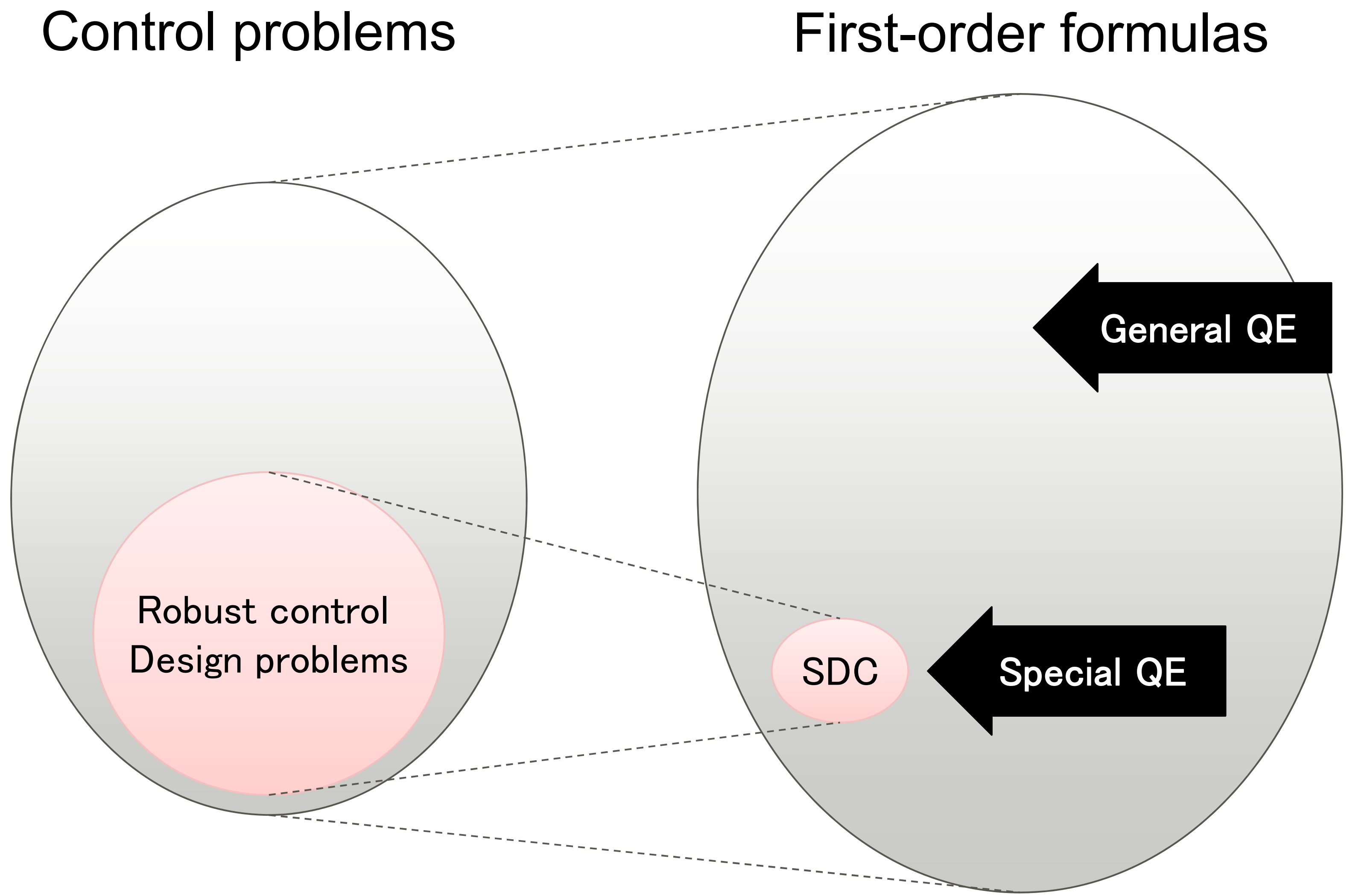
- Specialized QE (for restricted inputs)
  - Reducing the industrial problems into "nice / simple" formulas by exploring their structures.
  - Solving the formulas by specialized QE algorithms

- Examples
  - Sturm-Habicht sequence
    - Sign behavior of univariate polynomial
      - Sign definite condition (SDC): $\forall x \ (x \geq 0 \rightarrow f(x) > 0)$

Control system design problem

  - Virtual substitution
    - for Low-degree inputs (linear, quadratic)

Relevance of Special QE algorithm

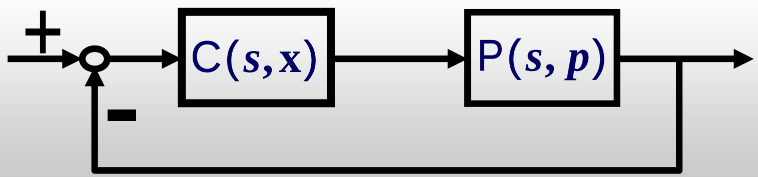


$$\forall x \ (x \geq 0 \rightarrow f(x) > 0)$$

Parametric robust control design

Problem

- Multi-objective low-order fixed-structure controller synthesis
  - Frequently required problems in industry
  - Specifications in frequency domain properties

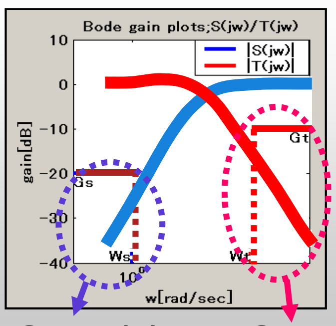


PI : $C(s) = k + \frac{m}{s}$

PID : $C(s) = k + \frac{m}{s} + \frac{d \cdot s}{1 + 0.1s}$

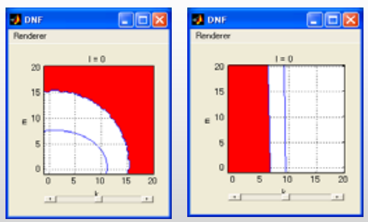
Our approach

- A parameter space approach by symbolic computation (QE)



Spec (a) Spec (b)

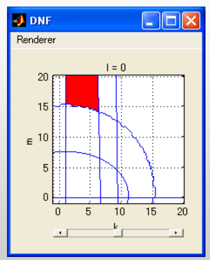
Parametric optimization



(a) (b)

Feasible regions of PI controller

Superpose



Robust control design by a general QE

FUJITSU

M.Jirstrand (1996)

Specifications

- $\| G(s) \|_{[0,\varpi_1]} < \gamma_s$
- $\| G(s) \|_{[\varpi_2,+\infty]} < \gamma_t$
- Gain margin > μ
- Phase margin > φ

- $0 < \varpi < \varpi_1 \rightarrow a^2 + b^2 < \gamma_s^2$
- $\varpi > \varpi_2 \rightarrow a^2 + b^2 < \gamma_t^2$
- $b = 0 \rightarrow a > -\dfrac{1}{\mu}$
- $[a^2 + b^2 = 1 \wedge b < 0] \rightarrow \dfrac{b}{a} > \tan(\phi)$

$a(s) = \mathrm{Re}[G(s)] \quad b(s) = \mathrm{Im}[G(s)]$

General QE
(QEPCAD)

Only small problems are solved
due to the double exponential complexity.
Useless for practical control problems!

Robust control design by a special QE

FUJITSU

Specifications

- $\|G(s)\|_{[0,\varpi_1]} < \gamma_s$
- $\|G(s)\|_{[\varpi_2,+\infty]} < \gamma_t$
- Gain margin > μ
- Phase margin > φ

M.Jirstrand (1996)

- $0 < \varpi < \varpi_1 \rightarrow a^2 + b^2 < \gamma_s^{\ 2}$
- $\varpi > \varpi_2 \rightarrow a^2 + b^2 < \gamma_t^{\ 2}$
- $b = 0 \rightarrow a > -\dfrac{1}{\mu}$
- $[a^2 + b^2 = 1 \wedge b < 0] \rightarrow \dfrac{b}{a} > \tan(\phi)$

$a(s) = \mathrm{Re}[G(s)] \quad b(s) = \mathrm{Im}[G(s)]$

General QE
(QEPCAD)

Anai & Hara (1999)

<u>Sign Definite Condition (SDC)</u>

$$\forall x\ (x \geq 0 \rightarrow f(x) > 0)$$

Special QE
(Sturm-Habicht seq.)

SDC reduction

FUJITSU

- H_∞-norm constraint

$$\boxed{G(s) = n(s)/d(s)}$$

$$\|G(s)\|_\infty := \sup_{\omega} |G(j\omega)| < 1$$

$$\Leftrightarrow \quad \forall \omega \;\; d(j\omega)d(-j\omega) > n(j\omega)n(-j\omega)$$

$$\Leftrightarrow \quad f(\omega^2) = d(j\omega)d(-j\omega) - n(j\omega)n(-j\omega) > 0$$

$$x = \omega^2$$

$$\Leftrightarrow \quad \boxed{\forall x > 0 \quad f(x) > 0}$$

- Frequency restricted H_∞-norm constraint

$$\|G(s)\|_{[\omega_1,\omega_2]} := \sup_{\omega_1 < \omega < \omega_2} |G(j\omega)| < 1$$

$$\Leftrightarrow \quad f(x) \neq 0 \text{ in } [\,-\omega_2^2, -\omega_1^2\,]$$

bilinear transformation

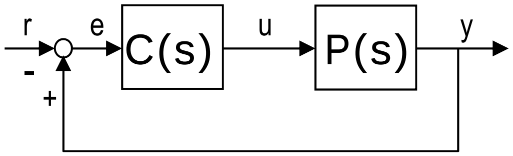
$$z = -(x + \omega_2^2)/(x + \omega_1^2)$$

$$\Leftrightarrow \quad \boxed{\forall z > 0 \quad h(z) > 0}$$

Example： mixed sensitivity problem

Mixed sensitivity problem

Specifications: Frequency restricted H_∞ norm constraints



Sensitivity function

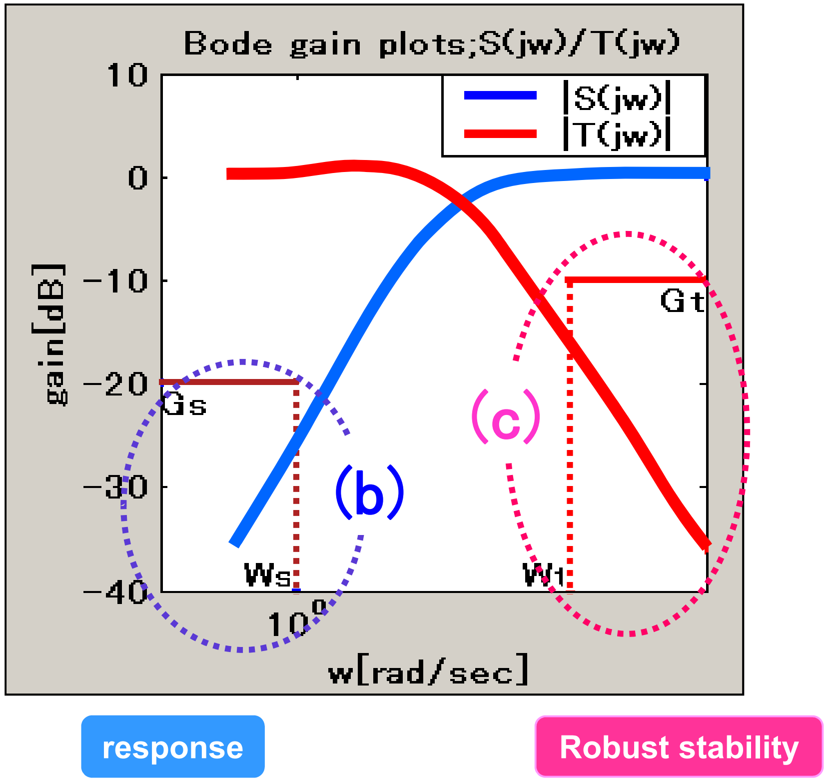
$$S = \frac{1}{1 + C(s)P(s)}$$

Complementary sensitivity

$$T = \frac{C(s)P(s)}{1 + C(s)P(s)}$$

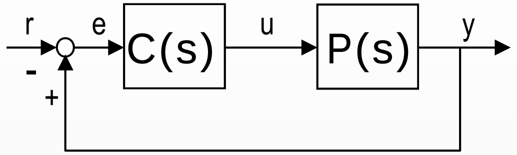
(b) $\| S(s) \|_{[0,1]} \equiv \max_{0 \le \omega \le 1} \| S(\boldsymbol{i}\omega) \| < 0.1$

(c) $\| T(s) \|_{[20,\infty]} \equiv \max_{20 \le \omega \le \infty} \| T(\boldsymbol{i}\omega) \| < 0.05$



Parametric robust control design

Stability with Mixed sensitivity



$$\mathrm{C}(s) = x_1 + \frac{x_2}{s}, \quad \mathrm{P}(s) = \frac{1}{s+1}$$

(a) Hurwitz Stability

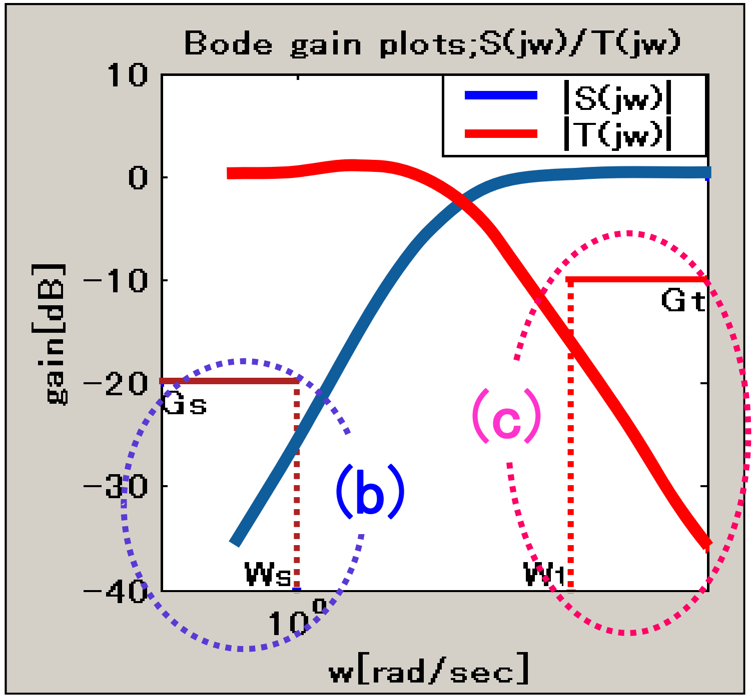
SDC

(b) Sensitivity

$$\| S(s) \|_{[0,1]} \equiv \max_{0 \le \omega \le 1} \| S(i\omega) \| < 0.1$$

(c) Complementary sensitivity

$$\| T(s) \|_{[20,\infty]} \equiv \max_{20 \le \omega \le \infty} \| T(i\omega) \| < 0.05$$



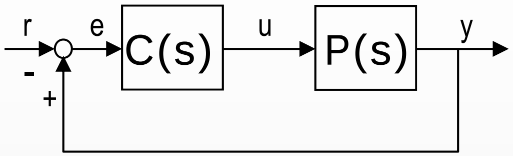
(b) $\Rightarrow$ $\forall z > 0 \ \ (x_2^2 - 2x_2 + x_1^2 - 99)z^3 + (3x_2^2 - 4x_2 + 2x_1^2 + 2x_1 - 99)z^2 + (3x_2^2 - 2x_2 + x_1^2 + 2x_1 - 99)z + x_2^2 > 0,$

(c) $\Rightarrow$ $\forall z > 0 \ \ z^3 + (-2x_1 + 1199)z^2 + (-2x_2 - 399x_1^2 - 1598x_1 + 479201)z - 399x_2^2 - 800x_2 - 159600x_1^2 - 319200x_1 + 63840400 > 0.$

Parametric robust control design

FUJITSU

Stability with Mixed sensitivity



$C(s) = x_1 + \frac{x_2}{s}$, $P(s) =$

(a) Hurwitz Stability

(b) Sensitivity

$$\| S(s) \|_{[0,1]} \equiv \max_{0 \le \omega \le 1} \| S(i\omega) \| < 0.1$$

(c) Complementary sensitivity

$$\| T(s) \|_{[20,\infty]} \equiv \max_{20 \le \omega \le \infty} \| T(i\omega) \| < 0.05$$

SDC

Specialized QE

(b) $\Rightarrow$ $\forall z > 0 \;\; (x_2^2 - 2x_2 + x_1^2 - 99)z^3 + (3x_2^2 - 4x_2 + 2x_1^2 + 2x_1 - 99)z^2 + (3x_2^2 - 2x_2 + x_1^2 + 2x_1 - 99)z + x_2^2 > 0,$

(c) $\Rightarrow$ $\forall z > 0 \;\; z^3 + (-2x_1 + 1199)z^2 + (-2x_2 - 399x_1^2 - 1598x_1 + 479201)z - 399x_2^2 - 800x_2 - 159600x_1^2 - 319200x_1 + 63840400 > 0.$

Sensitivity $S(s)$:

$$(P_3 \le 0 \wedge x_2 \ne 0) \vee (P_1 \ge 0 \wedge P_2 > 0) \vee (P_5 \ge 0 \wedge P_1 \ge 0 \wedge x_2 \ne 0)$$

where

$P_1 = x_2^2 - 2x_2 + x_1^2 - 99,$
$P_2 = 264627x_2^4 + 7128x_1x_2^3 - 349668x_2^3 - 3596x_1^3x_2^2 + 169274x_1^2x_2^2 + 462528x_1x_2^2 - 13152942x_2^2 + 2392x_1^4x_2 + 7952x_1^3x_2 - 426492x_1^2x_2 - 705672x_1x_2 + 19405980x_2 - 400x_1^6 - 1996x_1^5 + 105419x_1^4 + 352836x_1^3 - 9467766x_1^2 - 15524784x_1 + 288178803,$
$P_3 = x_1 + 11,$
$P_5 = x_1 - 9.$

Complemantary Sensitivity $T(s)$:

$$P_6 < 0$$

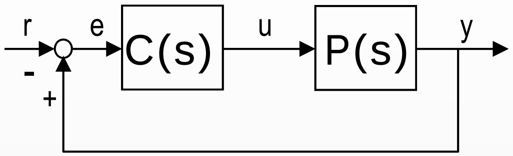
where

$$P_6 = 399x_2^2 + 800x_2 + 159600x_1^2 + 319200x_1 - 63840400.$$

Parametric robust control design

FUJITSU

Stability with Mixed sensitivity



$$C(s) = x_1 + \frac{x_2}{s}, \quad P(s) = \ldots$$

Specialized QE

(a) Hurwitz Stability

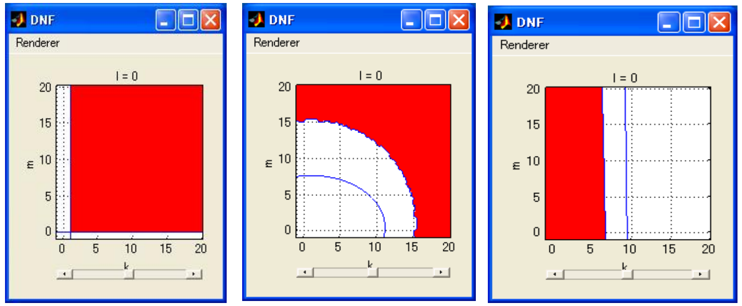
SDC

(b) Sensitivity

$$\| S(s) \|_{[0,1]} \equiv \max_{0 \le \omega \le 1} \| S(\boldsymbol{i}\omega) \| < 0.1$$

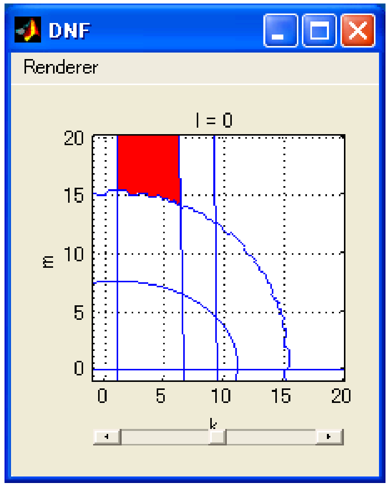
(c) Complementary sensitivity

$$\| T(s) \|_{[20,\infty]} \equiv \max_{20 \le \omega \le \infty} \| T(\boldsymbol{i}\omega) \| < 0.05$$



(a) (b) (c)

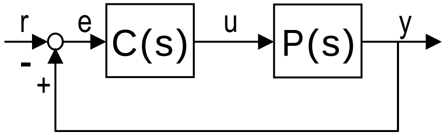
Superposing



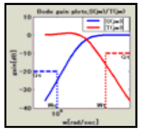
Robust control design

Our approach

Anai & Hara (ACC2000,IFAC02)



specification



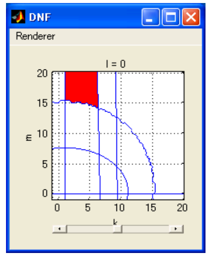
reduction

Multivariate Polynomial Inequalities (parametric)

Symbolic optimization

(QE)

Feasible regions



Frequency domain properties

(a) Hurwitz Stability
(b) H∞-norm constraint
(c) Gain/Phase margin
(d) Pole assignment
(e) Stability radius

Sign Definite Condition

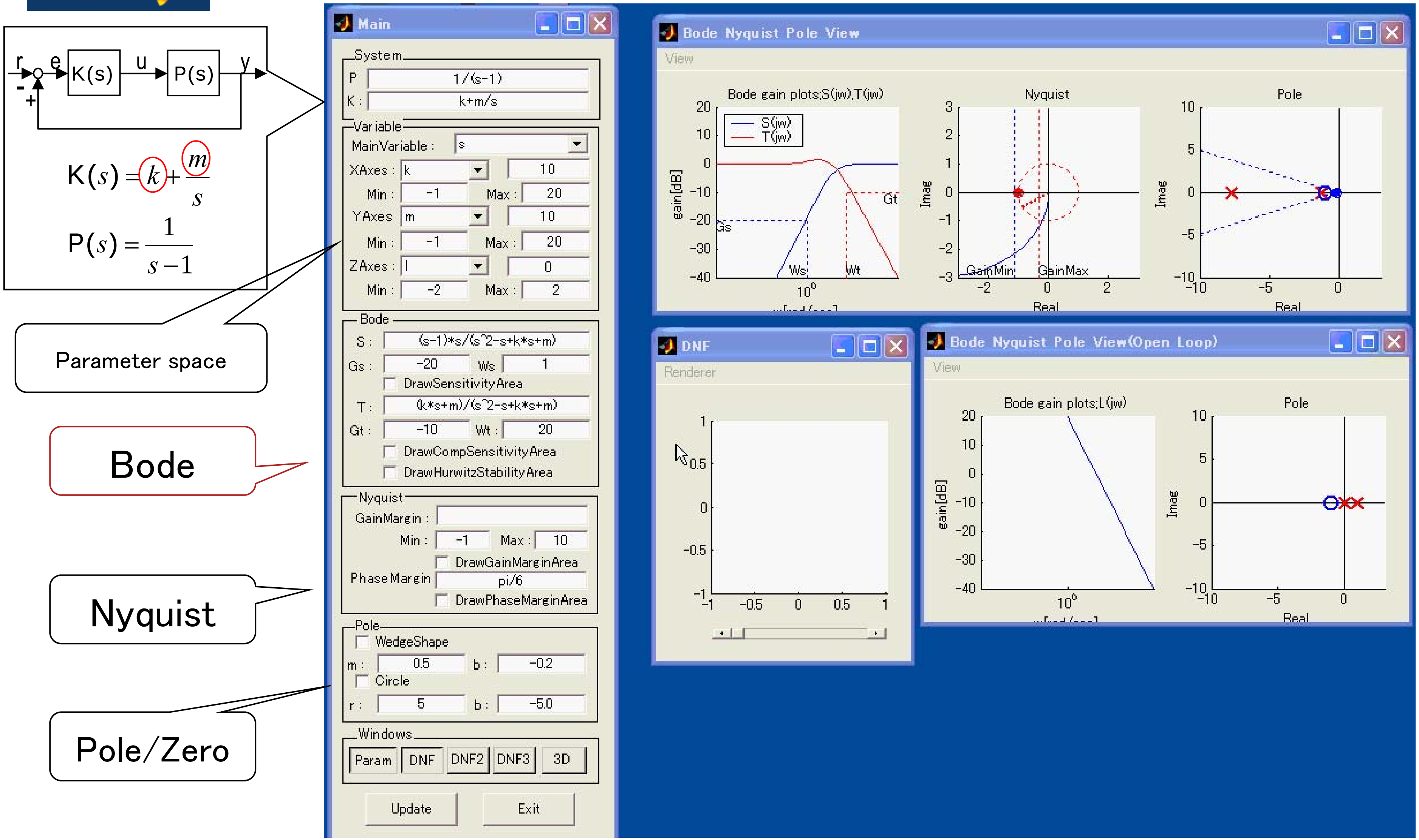
$$\forall x \; (x \geq 0 \rightarrow f(x) > 0)$$

Specialized QE for SDC
(Sturm-Habicht sequence)

Tractability

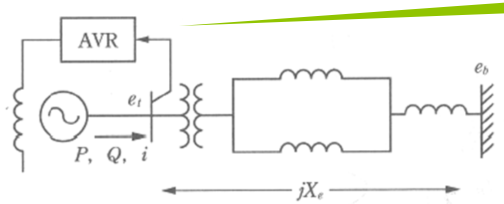
- **PI/PID for a plant with order 10 : < 1h**

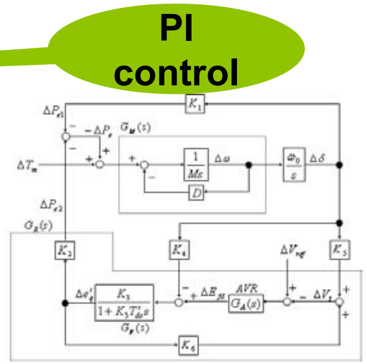
Parametric Robust Control Toolbox

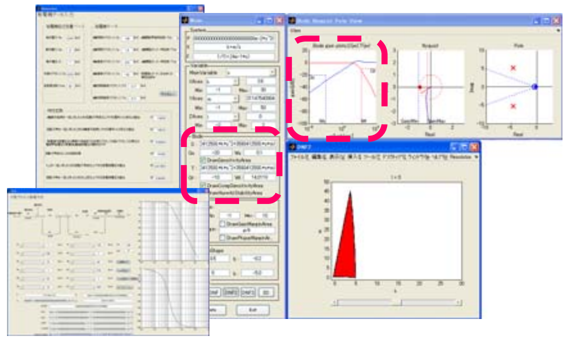


Parametric robust control design

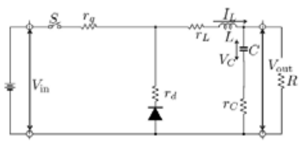
- Parametric robust control design by QE has been successfully applied to nontrivial industrial problems.
  - **Electric generating facility**
    - **generator excitation control design** (Yoshimura et al. 2008)

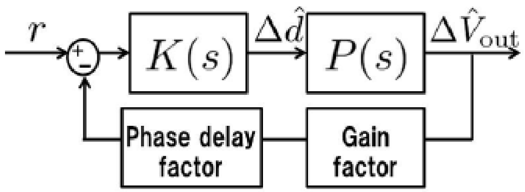




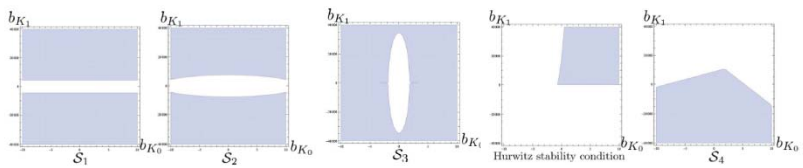


  - **Power supply units**
    - **digital controller design** (Matsui et al. 2013)





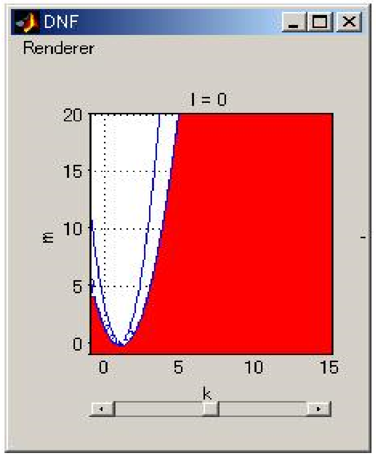
- **Specification 0** The closed-loop system is internal stable.
- **Specification 1** The gain> 45 dB when $0 \le \omega \le 1$.
- **Specification 2** The gain> 25 dB when $1 \le \omega \le 100$.
- **Specification 3** The gain crossover frequency >3000.
- **Specification 4** The phase margin (PM) >45 degree.
- **Specification 5** The gain margin (GM) >7 dB.

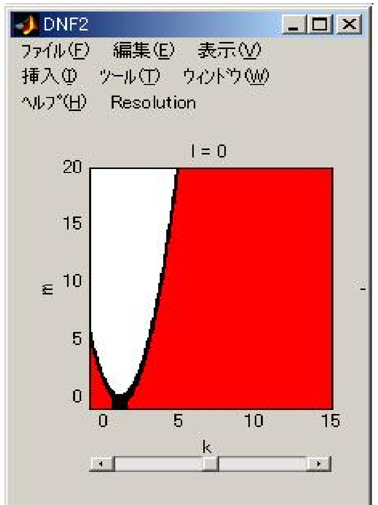


Approximate feasible parameter regions

FUJITSU

- Validated numerical method to solve first-order formula φ
  - approximately (but with guarantee) using interval arithmetic.
  - Repeated refinement of boxes and verification of T/F/U
    - **T={T implies that φ is true for all elements of B}**
    - **F={F implies that φ is false for all elements of B}**
    - **U={undecided }**

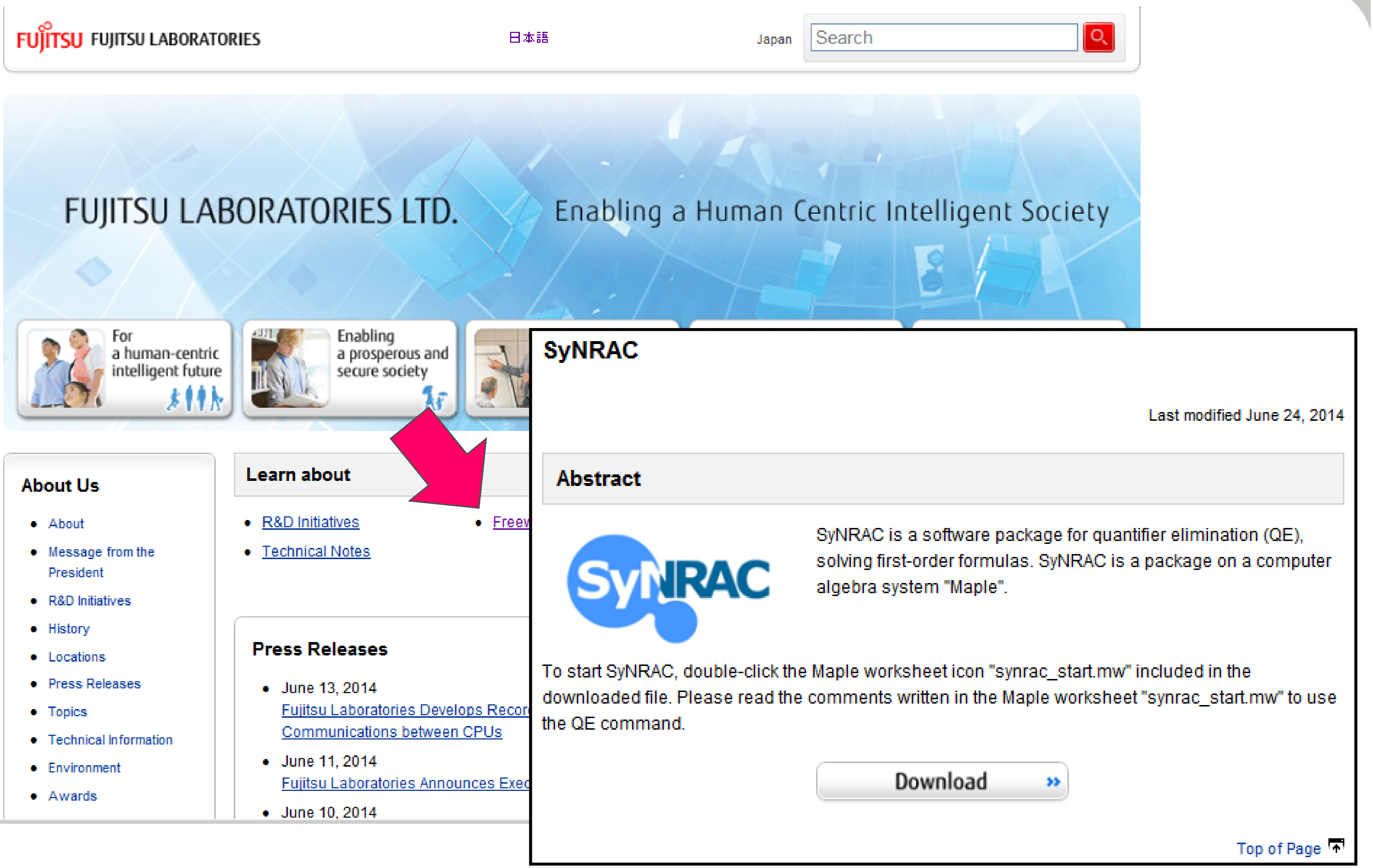




- Reference:
  - Approximate Quantified Constraint Solving by Cylindrical Box Decomposition (S. Ratschan, 2008)

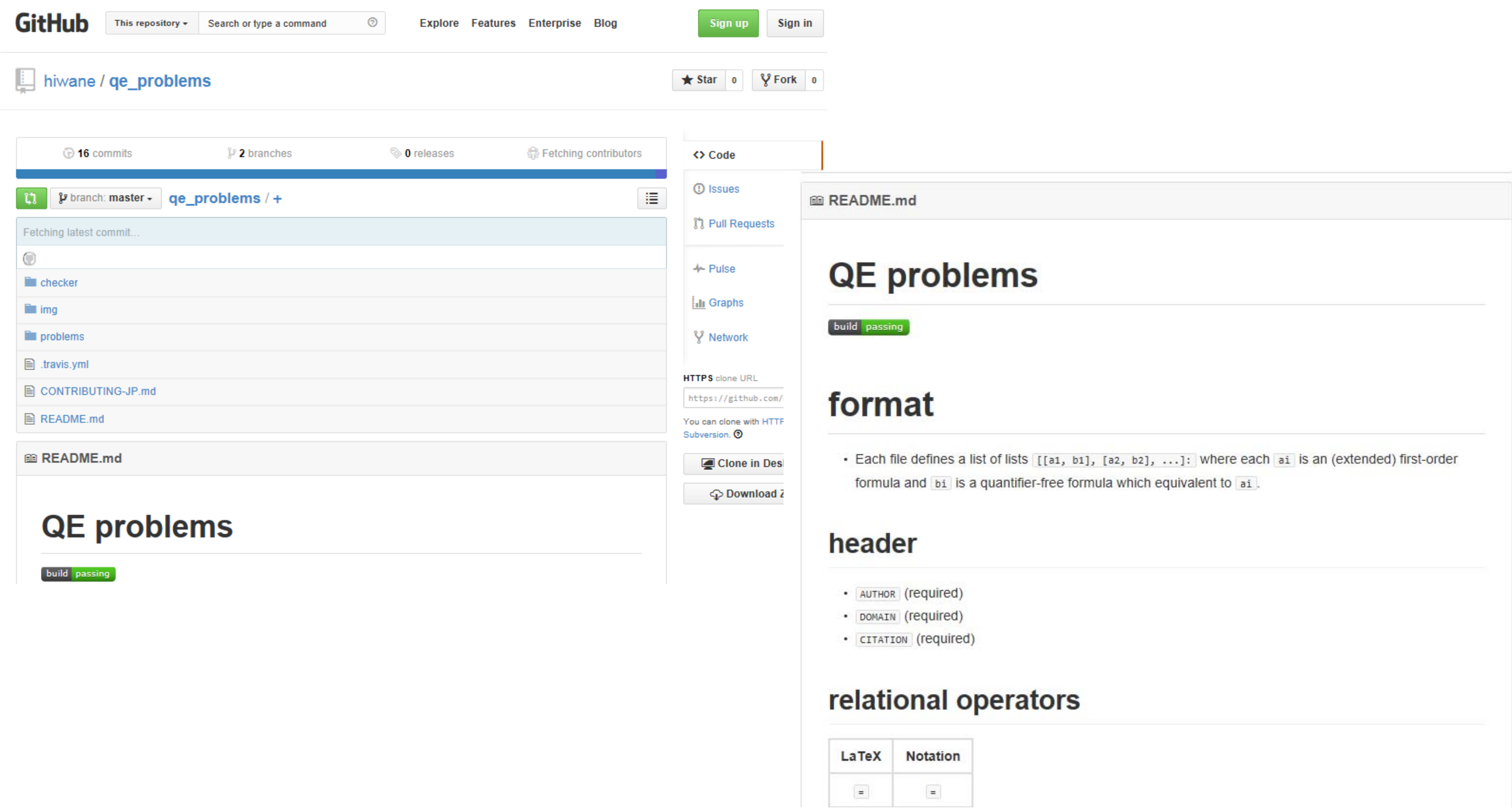
References

SyNRAC


FUJITSU FUJITSU LABORATORIES
日本語
Japan
Search
FUJITSU LABORATORIES LTD.
Enabling a Human Centric Intelligent Society
For a human-centric intelligent future
Enabling a prosperous and secure society
About Us
About
Message from the President
R&D Initiatives
History
Locations
Press Releases
Topics
Technical Information
Environment
Awards
Learn about
R&D Initiatives
Technical Notes
Freew
Press Releases
June 13, 2014
Fujitsu Laboratories Develops Recor
Communications between CPUs
June 11, 2014
Fujitsu Laboratories Announces Exec
June 10, 2014
SyNRAC
Last modified June 24, 2014
Abstract
SyNRAC
SyNRAC is a software package for quantifier elimination (QE), solving first-order formulas. SyNRAC is a package on a computer algebra system "Maple".
To start SyNRAC, double-click the Maple worksheet icon "synrac_start.mw" included in the downloaded file. Please read the comments written in the Maple worksheet "synrac_start.mw" to use the QE command.
Download
Top of Page

QE Benchmark problems

- GitHuB
  - https://github.com/hiwane/qe_problems



References

- H. Anai and S. Hara. A parameter space approach to fixed-order robust controller synthesis by quantifier elimination. International Journal of Control, 79(11):1321--1330, 2006.
- H. Anai and S. Hara. Fixed-structure robust controller synthesis based on sign definite condition by a special quantifier elimination. In Proceedings of American Control Conference, 2000, vol.2, 1312--1316, 2000.
- H. Anai and K. Yokoyama. Algorithms of Quantifier Elimination and their Applications: Optimization by Symbolic and Algebraic Methods (in Japanese). University of Tokyo Press, 2011.
- C. W. Brown. QEPCAD B: A program for computing with semi-algebraic sets using CADs. SIGSAM BULLETIN, 37:97--108, 2003.
- B. F. Caviness and J. R. Johnson, editors. Quantifier elimination and cylindrical algebraic decomposition. Texts and Monographs in Symbolic Computation. Springer, 1998.
- G. E. Collins. Quantifier elimination and cylindrical algebraic decomposition, pages 8--23. In Caviness and Johnson [5], 1998.
- A. Dolzmann and T. Sturm. REDLOG: Computer algebra meets computer logic. SIGSAM Bulletin 31(2):2--9.

References

- A. Dolzmann, T. Sturm, and V. Weispfenning. Real quantifier elimination in practice. In Algorithmic Algebra and Number Theory, pages 221--247. Springer, 1998.
- L. González-Vega. Applying quantifier elimination to the Birkhoff interpolation problem. Journal of Symbolic Computation, 22:83--103, July 1996.
- L. González-Vega. A combinatorial algorithm solving some quantifier elimination problems, pages 365--375. In Caviness and Johnson [5], 1998.
- L. González-Vega, T. Recio, H. Lombardi, and M.-F. Roy. Sturm-Habicht sequences determinants and real roots of univariate polynomials, pages 300--316. In Caviness, Bob F. and Johnson, Jeremy R [5], 1998.
- N. Hyodo, M. Hong, H. Yanami, S. Hara, and H. Anai. Solving and visualizing nonlinear parametric constraints in control based on quantifier elimination. Applicable Algebra in Engineering, Communication and Computing, 18(6):497--512, 2007.
- H. Iwane, H. Higuchi, and H. Anai. An effective implementation of a special quantifier elimination for a sign definite condition by logical formula simplification. In Proceedings of CASC 2013: 194--208, 2013.

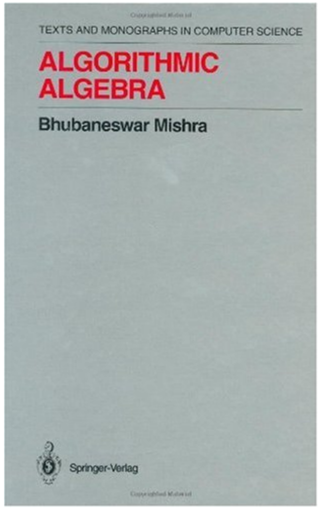
References

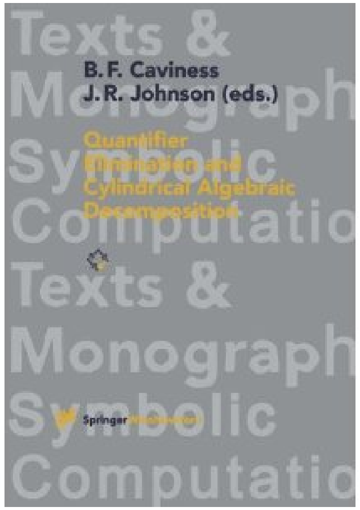
- H. Iwane, H. Yanami, and H. Anai. A symbolic-numeric approach to multi-objective optimization in manufacturing design. Mathematics in Computer Science, 5(3):315--334, 2011.
- H. Iwane, H. Yanami, H. Anai, and K. Yokoyama. An effective implementation of symbolic-numeric cylindrical algebraic decomposition for quantifier elimination. Theor. Comput. Sci. 479: 43--69, 2013.
- R. Loos and V. Weispfenning. Applying linear quantifier elimination. The Computer Journal, 36(5):450--462, 1993.
- Y. Matsui, H. Iwane, and H. Anai. Two controller design procedures using SDP and QE for a Power Supply Unit , "Development of Computer Algebra Research and Collaboration with Industry". COE Lecture Note Vol. 49 (ISSN 1881-4042), Kyushu University, 43--52, 2013.
- T. Matsuzaki, H. Iwane, H. Anai and N. Arai. The Most Uncreative Examinee: A First Step toward Wide Coverage Natural Language Math Problem Solving. In Proceedings of 28th Conference on Artificial Intellegence (AAAI 2014), to appear, 2014.
- T. Sturm. New domains for applied quantifier elimination. In V. G. Ganzha, E. W. Mayr, and E. V. Vorozhtsov, editors, Computer Algebra in Scientific Computing, volume 4194 of Lecture Notes in Computer Science, pages 295--301. Springer, 2006.

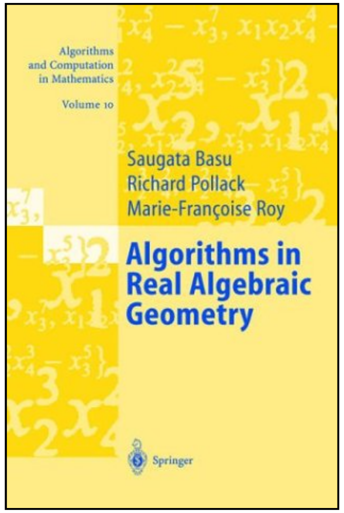
References

- T. Sturm and V. Weispfenning. Rounding and blending of solids by a real elimination method. In Proceedings of the 15th IMACS World Congress on Scientific Computation, Modelling, and Applied Mathematics (IMACS 97), pages 727--732, 1997.
- T. Sugimachi, A. Iwasaki, M. Yokoo, and H. Anai. Polynomial representation of auction mechanisms and automated mechanism design by quantifier elimination (in Japanese). In Proceedings of the autumn meeting of the Operation Research Society of Japan, 2012(30), 198--199, 2012
- A. Tarski, A decision method for elementary algebra and geometry. Prepared for publication by J. C. C. McKinsey. Berkeley, 1951.
- H. Yanami and H. Anai. The Maple package SyNRAC and its application to robust control design. Future Generation Comp. Syst. 23(5): 721--726, 2007.
- H. Yanami. Multi-objective design based on symbolic computation and its application to hard disk slider design. Journal of Math-for-Industry (JMI2009B-8) Kyushu University, pp.149--156, 2009.
- S. Yoshimura, H. Iki, Y. Uriu, H. Anai, and N. Hyodo. Generator Excitation Control using Parameter Space Design Method. In Proceedings of 43rd International Universities Power Engineering Conference (UPEC2008), pp.1--4, 2008.
- V. Weispfenning. Simulation and optimization by quantifier elimination. Journal of Symbolic Computation, 24:189--208, August 1997.

References







■ Algorithmic Algebra
(Texts & Monographs in Computer Science S.)
B. Mishra (1993)

■ Quantifier Elimination and Cylindrical Algebraic Decomposition
(Texts and Monographs in Symbolic Computation)
Bob F. Caviness , J. R. Johnson (1998)

■ Algorithms in Real Algebraic Geometry
(Algorithms and Computation in Mathematics, V. 10)
Saugata Basu, Richard Pollack, Marie-Francoise Roy (2006)

Reference

- **ISSAC 2004 Tutorial: Cylindrical Algebraic Decomposition**
  - Christopher W. Brown
  - http://www.usna.edu/Users/cs/wcbrown/index.html

C. W. Brown's Homepage

- Home
- Teaching
- Research
- Personal
- CS Faculty

Research Information

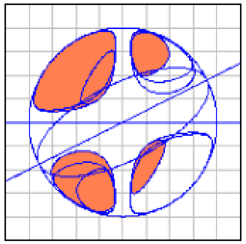
My primary research interest is in *Computer Algebra*, although I also have an interest in *Computer Science Education* and intelligent tutoring systems.

General	Software	Publications & Reports
• Curriculum Vitae • ISSAC '04 Tutorial • Tutorials	• QEPCAD • RegeXeX • SLFQ • ROLLE	• Refereed or Invited Publications • Technical Reports • Thesis

Quantifier Elimination & Computing with Semi-Algebraic Sets
One of the great contributions of Descartes was to connect algebra and geometry. Many of the most common geometric objects - like spheres, cubes, cones, and planes - can be defined with polynomials. Computers can deal with shapes like these by manipulating the polynomials that define them; reasoning about a circle, for example, by reasoning about the equation *x^2 + y^2 = 1*.

I'm interested in computing with the geometric objects that polynomials define, objects called *semi-algebraic sets*. This involves a wide variety of topics, from the theory behind algorithms for manipulating such objects to the practical issues of implementing systems that can really perform these computations. Such computations are so demanding that current programs are unable to solve interesting application problems within a reasonable amount of time and space. My ultimate goal is to provide systems for computing with semi-algebraic sets that are powerful enough to be used in scientific and industrial applications.

Much of my research into algorithms ends up implemented in the QEPCAD system for real quantifier elimination and formula simplification by cylindrical algebraic decomposition.



FUJITSU

shaping tomorrow with you