

# 符号情報利用による数値数式 CAD の効率的な実装

岩根秀直

(株) 富士通研究所\*

HIDENAO IWANE

FUJITSU LABORATORIES LTD

穴井宏和

(株) 富士通研究所/九州大学†

HIROKAZU ANAI

FUJITSU LABORATORIES LTD/KYUSHU UNIVERSITY

屋並仁史

(株) 富士通研究所‡

HITOSHI YANAMI

FUJITSU LABORATORIES LTD

## 1 はじめに

限定記号消去アルゴリズム (quantifier elimination (QE) algorithm) とは, 与えられた形式理論 (formal theory) について「限定記号付きの式 (一階述語論理式)」を入力とし「等価で限定記号無しの式」を出力するアルゴリズムのことである. QE アルゴリズムは最適化問題等の様々な適用分野があり非常に有効である. Cylindrical Algebraic Decomposition (CAD) [2] は, 与えられた多項式集合に対して, 符号が不変である領域に分割する手法で, 限定記号消去問題の基本的な解法のひとつである.

我々は 2009 年に Dynamic Evaluation (DE) と数値数式手法の組み合わせによる効率的な CAD の実装を提案した [6]. DE [4, 5] は拡大体の表現方法の一つである. DE は定義多項式の既約性を必要とせず, 無平方性のみを要求する. そのため代数拡大体上での因数分解を回避でき計算の効率化につなげることができた. 数値手法については, CAD 計算で代数拡大体上での実根の分離が必要となるが, 代数的数をそれを含む区間で置き換え, 区間係数一変数多項式の実根の分離として扱うようにした. すべての数値演算で区間演算を利用することで正確性を失わずに効率化を実現することができた.

数値手法を利用することで多くの記号計算を回避することができたが, 区間が 0 を含む場合には記号計算による判定が必要であるし, DE が無平方性を要求するために無平方因子分解の処理が回避できない.

本稿では CAD が再帰的に射影因子の符号不変な領域を構築することを利用し, 数値数式 CAD における記号計算をさらに回避する実装方法について述べる.

## 2 CAD アルゴリズム

1975 年に, G. E. Collins が, 与えられた多項式集合に対して変数空間を各多項式の符号が不変であるセル (cell) とよばれる領域に分割する新しい代数的方法である Cylindrical Algebraic Decomposition (CAD) [2] を提案した.

---

\*iwane@jp.fujitsu.com

†anai@jp.fujitsu.com

‡yanami@labs.fujitsu.com

CAD アルゴリズムは射影段階, 底段階, 持ち上げ段階 の 3 つの段階から成る. 射影段階では, 入力の多項式集合に射影演算子 PROJ を繰り返し適用することで 1 変数ずつ消去していく. 入力の多項式集合を  $F = \{f_i(x_1, \dots, x_r)\} \subseteq \mathbb{Q}[x_1, \dots, x_r]$  とすると, 射影段階により多項式列  $F_r = F, F_{r-1} = \text{PROJ}(F_r, x_r), F_{r-2} = \text{PROJ}(F_{r-1}, x_{r-1}), \dots, F_1 = \text{PROJ}(F_2, x_2)$  を得る.  $\cup_{i=1}^r F_i$  を既約因子分解して得られる多項式集合  $P \subseteq \mathbb{Q}[x_1, \dots, x_r]$  を射影因子という. ここで,  $P_k = \{p \in P \mid \deg_{x_k}(p) > 0, \deg_{x_i}(p) = 0 (i > k)\}$  とする. 底段階では,  $\mathbb{R} (= \mathbb{R}^1)$  の分解を行う. これは射影段階により得られた 1 変数多項式の集合  $P_1$  の実根を求めることで  $\mathbb{R}^1$  を分割する. 最後の持ち上げ段階では,  $\mathbb{R}^k$  の分解  $D_k$  と  $P_k$  を用いて  $\mathbb{R}^{k+1}$  の分解  $D_{k+1}$  を得る ( $k = 1, \dots, r-1$ ).

CAD により得られるセルでは多項式集合の符号が一定なので, 標本点 (sample point) と呼ばれる任意の一点の符号を評価することで入力の多項式集合の符号を評価することができる. 標本点の各座標は代数的数になり, その定義多項式とその根を唯 1 つ含む分離区間 (isolating interval) を用いて表現し, 分離区間を求めることを実根の分離と言う.

以下に持ち上げ段階のアルゴリズムを示す. ここでは拡大体上での実根の分離処理をそれぞれの射影因子に対して行い, 異なる実根を表す分離区間が重複しないようにする必要があり, 分離区間が重複した場合の共通根判定処理を行う. 判定処理により異なることが確認できた場合には, 標本点  $S$  を定める分離区間の区間幅を小さくすることで, 分離区間が重複しないようにする.

### アルゴリズム 1

LIFT( $P_k, S$ )

入力:  $P_k \subseteq \mathbb{Q}[x_1, \dots, x_k]$

セル  $C_{k-1}$  の標本点  $S = (s_1, \dots, s_{k-1}) \in \mathbb{R}^{k-1}$

出力:  $C_{k-1}$  を持ち上げた結果のセルの標本点

for  $f_i$  in  $P_k$  do

$L_i \leftarrow \text{REALROOTISO}(f_i, S)$  (分離区間のリスト)

for  $L_i, L_j$  do

if  $L_i, L_j$  の分離区間に重複がある then

if 同じ代数的数を表している then

一方の分離区間を削除する

else

区間の重複がなくなるまで精度を上げる

CAD の計算において  $D_{k+1}$  を構築する場合には  $P_k$  に対して符号一定な CAD  $D_k$  を用いる. このとき  $P_k$  は  $P_{k+1}$  に対する射影演算により生成されたもので,  $P_k$  の符号情報を利用することで  $P_{k+1}$  に関する情報を得ることができる. 本稿ではこれを利用して記号計算を回避することを考える.

## 2.1 数値数式 CAD

本稿は数値数式 CAD [6] の実装に対する効率化であるため, その実装方法について簡単に述べる.

数値数式 CAD では持ち上げ段階における実根の分離処理を数値演算に置き換えることで効率化を実現した. 以下に数値数式手法による実根の分離処理を示す.

### アルゴリズム 2

REALROOTISO( $f, S$ )

入力:  $f \in \mathbb{Q}[x_1, \dots, x_{k-1}, x_k]$  ( $k \geq 2$ )

セル  $C_{k-1}$  の標本点  $S = (s_1, \dots, s_{k-1}) \in \mathbb{R}^{k-1}$   
 出力:  $f(s_1, \dots, s_{k-1}, x_k)$  の分離区間

```

g ← SUBSTSAMPLEPOINT(f, S)
L ← INTVREALROOTISO(g)
if error then
  f ← SYMBOLICSQRFREE(f, S)
  do
    g ← SUBSTSAMPLEPOINT(f, S)
    L ← INTVREALROOTISO(g)
    if error then
      S の精度をあげる
    else
      break
return L

```

ここで, SUBSTSAMPLEPOINT は  $f$  に標本点を表す分離区間を代入し, 区間係数の一変数多項式を返す関数である. INTVREALROOTISO は区間係数一変数多項式の実根の分離処理で, Krawczyk 法 [7] を基本とした実装としている. Krawczyk 法は無平方でない場合には正しく実根を分離できないためエラーを返す. 我々の実装では DE を利用しているため, DE が要求する定義多項式の無平方性を満たす必要がある. そこで Krawczyk 法での実根の分離処理が失敗した場合にのみ記号計算による代数拡大体上での無平方因子分解 SYMBOLICSQRFREE を呼び出すようにすることで不要な演算を回避することを実現した. しかし, INTVREALROOTISO は係数の区間が十分に小さくない場合にも実根の分離ができないためにエラーを返すのでその場合には不要な記号演算を呼び出すことになる. ここで複素根の重根については除去できないが CAD の実装では実根のみを使用するため影響がない.

SUBSTSAMPLEPOINT では分離区間の精度が十分でない場合には係数が膨張してしまうことがあるため, 復帰する多項式の係数は符号が確定しているようにした. つまり, 係数が本当に 0 となる場合以外には係数の区間が 0 を含まないようにした. 以下に代入処理のアルゴリズムを示す.

### アルゴリズム 3

SUBSTSAMPLEPOINT( $f, S$ )

入力:  $f \in \mathbb{Q}[x_1, \dots, x_{k-1}, x_k]$  ( $k \geq 2$ )

セル  $C_{k-1}$  の標本点  $S = (s_1, \dots, s_{k-1}) \in \mathbb{R}^{k-1}$

出力:  $f$  に標本点の分離区間を代入して得られる区間係数一変数多項式

```

for  $i = 0$  to degree of  $f$  in  $x_k$ 
   $c_i \leftarrow$  coefficient of  $x_k^i$  in  $f$ 
   $I_i \leftarrow c_i(s_1, \dots, s_{k-1})$ 
  if  $I_i$  が 0 を含む then
    if SYMBOLICZEROCHECK( $c_i, S$ ) then
       $I_i \leftarrow [0, 0]$ 
    else
       $I_i$  が 0 を含まなくなるまで  $S$  の精度をあげる
return  $\sum_i I_i x^i$ 

```

標本点の代入した結果が 0 でない場合にも分離区間を代入する際にその精度が十分でない場合に入力の多項式  $f$  の係数が大きい場合等に 0 を含んでしまうため、記号演算によりチェックが必要となることがあった。

これまでの話をまとめると、数値数式 CAD の持ち上げ段階において記号計算が発生する箇所は以下の 3 箇所である。

1. 係数に標本点を代入した際に発生する零判定 (区間が 0 を含む場合)
2. DE が要求する無平方因子分解 (実根の分離に失敗した場合)
3. 異なる射影因子から生成される代数的数の共通根判定 (区間が重複した場合)

次節では CAD の計算結果を利用することによりこれらの記号計算を回避する方法について述べる。

### 3 符号情報を利用した数値数式 CAD の実装

CAD の計算量に大きく影響するため、これまで多くの研究が射影因子を減らすことに注力されてきた。本稿で提案する手法は使用する射影演算に依存する。ここでは McCallum 射影演算子 [9] を例にして紹介する。

#### アルゴリズム 4

$\text{PROJMC}(P_k, x_k)$   
 入力:  $P_k \subseteq \mathbb{Q}[x_1, \dots, x_{k-1}, x_k]$  ( $k \geq 2$ )  
 $x_k$  主変数  
 出力: 射影因子  $P_{k-1} \subseteq \mathbb{Q}[x_1, \dots, x_{k-1}]$   
 $G \leftarrow \emptyset$   
 for  $f$  in  $P_k$  do  
 $G \leftarrow G \cup \{f \text{ の } x_k \text{ に関するすべての係数}\}$   
 $G \leftarrow G \cup \{f \text{ の } x_k \text{ を主変数とみたときの判別式}\}$   
 for  $f, g$  in  $P_k$  do ( $f \neq g$ )  
 $G \leftarrow G \cup \{f \text{ と } g \text{ の } x_k \text{ を主変数とみたときの終結式}\}$   
 return  $G$

PROJMC により得られる射影因子には係数、判別式、終結式の情報が含まれている。この符号情報を利用することで多くの記号計算の回避が実現できる。

#### 3.1 標本点の分離区間の代入処理

ここではアルゴリズム 3 における符号情報の利用について述べる。PROJMC では入力の多項式のすべての係数の情報を射影因子として出力する。つまり、持ち上げ段階において多項式  $f$  に標本点  $S$  を代入した結果の符号は一定で、その値は既知である。従ってアルゴリズム 3 において発生する記号演算を完全に回避することができる。

#### アルゴリズム 5

$\text{NEWSUBSTSAMPLEPOINT}(f, S)$   
 入力:  $f \in \mathbb{Q}[x_1, \dots, x_{k-1}, x_k]$  ( $k \geq 2$ )  
 セル  $C_{k-1}$  の標本点  $S = (s_1, \dots, s_{k-1}) \in \mathbb{R}^{k-1}$

出力:  $f$  に標本点の分離区間を代入して得られる区間係数一変数多項式

```

for  $i = 0$  to degree of  $f$  in  $x_k$ 
   $c_i \leftarrow$  coefficient of  $x_k^i$  in  $f$ 
   $I_i \leftarrow c_i(s_1, \dots, s_{k-1})$ 
  if  $I_i$  が  $0$  を含む then
    if 射影因子の符号から  $c_i(s_1, \dots, s_{k-1})$  が  $0$  である then
       $I_i \leftarrow [0, 0]$ 
    else
       $I_i$  が  $0$  を含まなくなるまで  $S$  の精度をあげる
return  $\sum_i I_i x^i$ 

```

### 3.2 代数拡大体上での無平方因子分解

数値数式 CAD における実根の分離処理の部分では符号情報を利用することで不要な無平方因子分解を回避することができる。

#### アルゴリズム 6

NEWREALROOTISO( $f, S$ )  
 入力:  $f \in \mathbb{Q}[x_1, \dots, x_{k-1}, x_k]$  ( $k \geq 2$ )  
       セル  $C_{k-1}$  の標本点  $S = (s_1, \dots, s_{k-1}) \in \mathbb{R}^{k-1}$   
 出力:  $f(s_1, \dots, s_{k-1}, x_k)$  の分離区間

```

 $g \leftarrow$  SUBSTSAMPLEPOINT( $f, S$ )
 $L \leftarrow$  INTVREALROOTISO( $g$ )
if error then
   $c_i \leftarrow$  leading coefficient of  $f$  in  $x_k$ 
  if  $f$  の判別式が  $0$  then
     $f \leftarrow$  SYMBOLICSQRFREE( $f, S$ )
  do
     $g \leftarrow$  SUBSTSAMPLEPOINT( $f, S$ )
     $L \leftarrow$  INTVREALROOTISO( $g$ )
    if error then
       $S$  の精度をあげる
    else
      break
return  $L$ 

```

ここでは  $f$  の  $x_k$  に関する主係数が  $S$  で  $0$  になる場合に、判別式の値は必ず  $0$  になるために不要な無平方因子分解が発生することがある。それでも、従来の方法では発生していた不要な無平方因子分解の多くを回避できるため、計算効率の改善が実現できる。

### 3.2.1 次数 2 の多項式の無平方因子分解

SYMBOLICSQRFREE は代数拡大体上での無平方因子分解を行う。次数 2 の多項式が平方因子をもつと分かっている場合には以下のように置き換えることができる。

$$c_2x^2 + c_1x + c_0 = w(2c_2x + c_1)^2. \quad (1)$$

ここで、 $w$  は  $c_2$  と符号が同じ値となる定数である。

CAD 計算では符号情報のみに興味があるため定数倍の違いは結果に影響しないので、 $|w| = 1$  として良い。また  $c_2$  の符号は射影因子の符号からすぐに得ることができる。

### 3.3 共通根判定処理

ここでは異なる多項式から得られた分離区間が重複した場合の記号計算回避について述べる。

$f, g \in \mathbb{Q}[x_1, \dots, x_k]$  に標本点  $S \in \mathbb{R}^{k-1}$  を代入して得られる一変数多項式の分離区間が重複する場合には、記号計算で共通根であるか判定する必要がある。ここでは終結式の符号情報が得られているため、終結式の値が 0 でない場合には記号計算を行うことなく標本点の精度を上げることで実根を分離することが可能となる。

## 4 計算機実験

ここで提案した手法は SyNRAC [6] 上に実装した。表の SyN2009 は 2009 年に提案した SyNRAC の実装で Maple ユーザ言語で実装している。SyN2011 は本稿で提案した内容を C 言語で実装したものである。他の CAD を基本とする QE ツールである QEPCAD B 1.58 [1], Mathematica 8.0 [10] との比較も行った。良く知られた QE のベンチマーク問題 (adam1 から adam3) と、多目的最適化問題を QE を用いて解く問題 (moorea, wilson, lampinen), 社内で発生した最適化問題 (opt1, opt2) を比較のために使用した。

表 1 は計算時間 (CPU 時間 (秒)) を表している。使用した環境は Intel(R) Core(TM) i3 CPU U330 1.20 GHz で 2.92 GByte メモリである。2009 年に提案した SyNRAC の実装に比べても記号計算の回避による効率化の実現が確認できる。またいくつかの問題では Mathematica よりも良い結果を得ることができた。

表 2 は CAD 計算における記号計算の呼出回数を表している。符号情報の利用により多くの記号計算を削減できていることが確認できる。

表 3 は SyNRAC 上で実装した、次数 2 の場合の無平方因子分解処理を通常の拡大体上での演算と、式 (1) による演算による実行時間を表している。扱った問題では多くの呼出が発生し効率化が実現できていることを確認できる。

## 5 まとめ

数値数式 CAD 計算における符号情報の利用による記号計算の回避方法について述べた。計算結果を利用することによる記号計算の回避であるため負荷はほとんど発生せず、これにより多くの不要な記号計算を回避でき、大きな効率化を実現できた。

本稿では McCallum 射影演算子に対する手法について述べたが、部分終結式主係数 (principal subresultant coefficient) を用いた Collins 射影演算子等を利用するとより多くの情報が得られる。例えば、重根の数が正

problem	SyN2009	SyN2011	Mathematica	QEPCAD
adam1	0.01	0.22	0.71	0.90
adam2-1	36.97	3.02	2.93	356
adam2-2	154	4.69	3.16	1086
adam3	90.39	3.96	18.32	>3600
mooea	>3600	29.79	4.22	>3600
wilson	>3600	17.79	5.44	>3600
lampinen	>3600	10.26	6.99	137
opt1	>3600	29.00	49.51	>3600
opt2	>3600	28.60	1054	>3600

表 1: Computing time (sec)

	零判定		無平方因子分解	
	SyN2009	SyN2011	SyN2009	SyN2011
adam1	1	0	0	0
adam2-1	155	0	1640	5
adam2-2	902	27	8340	131
adam3	1054	499	5678	4

表 2: 関数の呼出回数

確に分かるため数値計算のみで CAD 構築を実現できる可能性がある。ただし、部分終結式主係数を用いた射影演算子は McCallum が提案したものにくらべ、一般に大きな射影因子集合を構築するために全体としてのセルの数が増加してしまう傾向にある。扱う問題や数値手法の効率化によっては部分終結式主係数を用いたもののほうが良い結果をもたらす可能性もあるが、我々の実験では McCallum 射影演算子を用いたものが良い結果が確認できている。

## 参 考 文 献

- [1] C. W. Brown. QEPCAD B: A program for computing with semi-algebraic sets using CADs. *SIGSAM BULLETIN*, 37:97–108, 2003.
- [2] G. E. Collins. *Quantifier elimination and cylindrical algebraic decomposition*, pages 8–23. Texts and Monographs in Symbolic Computation. Springer, 1998.
- [3] K. Deb. *Multi-Objective Optimization using Evolutionary Algorithms*. Wiley-Interscience Series in Systems and Optimization. John Wiley & Sons, Chichester, 2001.
- [4] J. D. Dora, C. Dicrescenzo, and D. Duval. About a new method for computing in algebraic number fields. In *Research Contributions from the European Conference on Computer Algebra-Volume 2*, EUROCAL '85, pages 289–290, London, UK, 1985. Springer-Verlag.
- [5] D. Duval. Algebraic numbers: An example of dynamic evaluation. *Journal of Symbolic Computation*, 18(5):429–445, 1994.

problem	符号情報利用なし (秒)	符号情報利用あり (秒)	呼出回数
mooea	30.65	27.70	584
wilson	21.95	19.68	297
lampinen	12.91	11.80	71
opt1	30.67	32.99	31
opt2	36.72	35.75	876

表 3: 次数 2 の無平方因子分解処理

- [6] H. Iwane, H. Yanami, H. Anai, and K. Yokoyama. An effective implementation of a symbolic-numeric cylindrical algebraic decomposition for quantifier elimination. In *Proceedings of the 2009 International Workshop on Symbolic-Numeric Computation*, volume 1, pages 55–64, 2009.
- [7] R. Krawczyk. Newton-algorithmen zur bestimmung von nullstellen mit fehlerschranken. *Computing*, pages 187–201, 1969.
- [8] J. Lampinen. *Multiobjective Nonlinear Pareto-Optimization*. Laboratory of Information Processing, 2000.
- [9] S. McCallum. An improved projection operator for cylindrical algebraic decomposition. In B. F. Caviness and J. R. Johnson, editors, *Quantifier elimination and cylindrical algebraic decomposition*, Texts and Monographs in Symbolic Computation, pages 242–268. Springer, 1998.
- [10] A. W. Strzeboński. Cylindrical algebraic decomposition using validated numerics. *Journal of Symbolic Computation*, 41(9):1021–1038, 2006.
- [11] B. Wilson, D. Cappelleri, T. W. Simpson, and M. Frecker. Efficient Pareto frontier exploration using surrogate approximations. *Optimization and Engineering*, 2:31–50, 2001.

## A 例

実験で使用した問題を以下に示す。

adam1 [10] Stability of Dormand-Prince fifth-order embedded seven-stage method (Example 4.4 from Hong, et al. (1997))

$$\forall x \forall y ((x < 0 \wedge x^2 + y^2 < \frac{99438}{100000}) \Rightarrow R(x + iy)R(x - iy) < 1),$$

where

$$R(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6} + \frac{z^4}{24} + \frac{z^5}{120} + \frac{z^6}{600}.$$

adam2 [10] Stability of a six-point upwind-based second-order accurate scheme for approximating a two-dimensional advection equation (Example 5.4 from Hong, et al. (1997))

$$2-1 \quad \forall \alpha \forall \beta \forall C_2 ((\alpha \geq 0 \wedge \beta \geq 0 \wedge 4(\alpha^2 + \beta^2) < 1) \Rightarrow (B \leq 0 \vee D \leq 0)),$$

$$2-2 \quad \forall \alpha \forall \beta \forall C_2 ((0 \leq \alpha \leq 1 \wedge 0 \leq \beta \leq 1) \Rightarrow A \leq 0 \wedge C \leq 0 \wedge (B \leq 0 \vee D \leq 0)),$$

where

$$\begin{aligned}
A &= C_2^4(\alpha - \beta + 1)(\alpha - \beta - 1)(\alpha - \beta)^2, \\
B &= 2C_2^4\beta(3\alpha^2\beta - 2\alpha^2 - 2\alpha\beta^2 + \alpha + \beta^3 - \beta) \\
&\quad + 4C_2^3\alpha\beta(\alpha^2 - \alpha + \beta^2 - \beta) + 2C_2^2\alpha(\alpha^3 - 2\alpha^2\beta + 3\alpha\beta^2 - \alpha - 2\beta^2 + \beta), \\
C &= C_2^4\beta^2(\beta^2 - 1) + 4C_2^3\alpha\beta^2(\beta - 1) + \alpha^2(\alpha^2 - 1) \\
&\quad + 2C_2^2\alpha\beta(3\alpha\beta - 2\alpha - 2\beta + 1) + 4C_2\alpha^2\beta(\alpha - 1), \\
D &= C_2^2R + 2C_2S + T, \\
R &= 8\alpha^2\beta^2 - 12\alpha^2\beta + 5\alpha^2 - 8\alpha\beta^3 + 8\alpha\beta^2 + 2\alpha\beta - 4\alpha + 4\beta^4 - 4\beta^3 - 3\beta^2 + 4\beta, \\
S &= 4\alpha^3\beta - 2\alpha^3 - 4\alpha^2\beta^2 - 2\alpha^2\beta + \alpha^2 + 4\alpha\beta^3 - 2\alpha\beta^2 + 2\alpha\beta - 2\beta^3 + \beta^2, \\
T &= 4\alpha^4 - 8\alpha^3\beta - 4\alpha^3 + 8\alpha^2\beta^2 + 8\alpha^2\beta - 3\alpha^2 - 12\alpha\beta^2 + 2\alpha\beta + 4\alpha + 5\beta^2 - 4\beta.
\end{aligned}$$

adam3 [10] Robust multi-objective feedback design (Example 4.2 from Dorato, et al. (1997))

Find the set of  $n$  satisfying:

$$\begin{aligned}
&\exists q_1 \exists q_2 \forall w (q_1 > 1 \wedge q_2 > 0 \wedge n > 0 \wedge \\
&\quad (n - q_1^2)w^4 + (n((q_1 + 1)^2 - 2q_2) - (q_1^2 + q_2^2))w^2 + (n - 1)q_2^2 \geq 0 \wedge \\
&\quad (n - q_1^2)w^4 + (n((q_1 - 1)^2 - 2q_2) - (q_1^2 + q_2^2))w^2 + (n - 1)q_2^2 \geq 0).
\end{aligned}$$

mooea: extended problem of example 1 in [3, p. 11]

$$\begin{aligned}
\exists x_1 \exists x_2 \exists x_3 \quad &(f = x_1^2 + x_2^2 + x_3 \wedge g = (x_1 - 1)^2 + x_2^2 + x_3 \wedge \\
&-2 \leq x_1 \leq 2 \wedge -2 \leq x_2 \leq 2 \wedge -1 \leq 10x_3 \leq 1).
\end{aligned}$$

wilson [11]

$$\begin{aligned}
\exists x_1 \exists x_2 \quad &(f = (x_1 - 2)^2 + (x_2 - 1)^2 \wedge g = x_1^2 + (x_2 - 6)^2 \wedge \\
&2/5 \leq x_1 \leq 8/5 \wedge 2 \leq x_2 \leq 5).
\end{aligned}$$

lampinen [8, p. 6]

$$\exists x_1 \exists x_2 \quad (f = x_1^2 + x_2 \wedge g = x_1 + x_2^2 \wedge -10 \leq x_1 \leq 10 \wedge -10 \leq x_2 \leq 10).$$