数理解析研究所講究録
第1666巻 2009年 78-86

QE のための数値数式 paritial CAD の実装

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

限量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「限量記号付きの式（一階述語論理式)」を入力とし「等価で限量記号無しの式」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，与えられた多項式系に対して，符号が不変である領域に分割する手法で，限量記号消去問題の基本的な解法のひとつである．そのため CAD は重要な手法であるが計算量が大きく，問題が大きくなると解けなくなることが多い．

Maple 上で実装した QE ツールボックスである SyNRAC の一部として数値数式 full CAD を実装した．代数拡大体上での計算量を抑えるため，数値的な手法を導入し効率化を実現できたが，QEPCAD 等の他の CAD 実装ツールに比べ十分な成果が得られなかった．

本稿では，さらなる効率化のため実装した partial CAD [3] とその実験結果について述べる．

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，CAD による QE アルゴリズムを提案した．

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手続き 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 $f = \{f_i(x_1, \ldots, x_r)\} \subset \mathbb{Q}[x_1, \ldots, x_r]$ とすると，射影段階により多項式列 $f = \mathrm{PROJ}^{(0)}(f)$, $\mathrm{PROJ}^{(1)}(f)$, $\mathrm{PROJ}^{(2)}(f)$, ..., $\mathrm{PROJ}^{(r-1)}(f)$ を得る．それらを既約因子分解して得られる多項式集合 $P \subset \mathbb{Q}[x_1, \ldots, x_r]$

*iwane@jp.fujitsu.com
†anai@jp.fujitsu.com
‡yanami@labs.fujitsu.com

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 の実根の分離により求める．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 $(k = 1, \ldots, r-1)$.

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与えられた代数的命題文を得ることができる．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) を用いて表現する．

SyNRAC の実装では， CAD の出力は標本点である．

定義 1
与えられた代数的命題文の真偽値 (*truth value*) が一定になるように分解する *CAD* を **partial CAD** とよび，それと区別するため，与えられた多項式集合を符号一定の領域に分解する *CAD* を **full CAD** とよぶ．

定義 2
セル $c \subset \mathbb{R}^i$ のとき，i を c の **level** とよぶ．

以下 3 節で full CAD アルゴリズム，4 節でその改良である partial CAD について，アルゴリズムとその実験結果をそれぞれ示す．

3 full CAD

本章では full CAD アルゴリズムとその実験結果について述べる．

3.1 full CAD アルゴリズム

full CAD での持ち上げ段階のアルゴリズムを以下に示す．レベル k のセル c の標本点を持ち上げる場合には，射影因子 $\prod_{p \in P_k} p$ に c の標本点を代入することで得られる一変数多項式の根の分離区間を求めることで得られる．計算効率のためアルゴリズム 1 では $p \in P_k$ の分離区間をそれぞれ求め，分離区間が重複していた場合には同一の代数的数か調べ必要であれば分離区間の精度をあげることで $\prod_{p \in P} p$ の分離区間を求めている．

ここで使用している関数 choose はセルの処理順序を決定するアルゴリズムをあらわす．full CAD においてすべてのセルに対して持ち上げを行うため処理順序は計算時間に影響しない．

アルゴリズム 1
syn_lift$(P = \{P_1, \ldots, P_r\}, D_1)$
入力: P: 射影因子
D_1: $\mathbb{R}^1$ の *CAD*
出力: P が符号一定のセル
$L \leftarrow D_1$
while L *is not empty do*
$c \leftarrow$ choose(L)
$k \leftarrow$ level(c)
for p *in* P_{k+1} *do*
$r_p \leftarrow$ { p の分離区間のリスト }
for p, q *in* P *do*
if r_p と r_q に重複する分離区間が存在する *then*

表 1: full CAD との計算時間の比較（単位：秒）

ex	deg	dim	#	Mathematica	QEPCAD	SyNRAC	detchiage
1	12	2	3	0.12(81)	0.83(58)	24.5(257)	0.604(263)
2-1	8	3	5	1.94(4853)	389 (6835)	>3600	273(70353)
2-2	12	2	3	2.61(5053)	1235(11653)	>3600	> 3600
a-1	4	3	2	1.49(470)	F(306)	2534(8751)	12.1(9387)
a-2	4	3	1	2.19(403)	F(270)	> 3600	2.84(1241)
a-3	3	4	1	1.07(1255)	3.68(1718)	> 3600	14.5(6403)
a-4	4	3	1	1.23(210)	0.34(464)	7.11(763)	0.89(819)
b-1	4	4	1	15.7(8859)	F(1272)	114(13089)	37.1(13861)

if 重複する分離区間は同じ代数的数を表している *then*
　一方の分離区間をリストから削除
else
　重複がなくなるまで分離区間の精度をあげる

$C \leftarrow \{\ r_p$ から構成されるセルのリスト $\}$
$D_{k+1} \leftarrow D_{k+1} \cup C$
if $k < r$ *then*
　$L \leftarrow L \cup C$
return $D = \{D_1, \ldots, D_r\}$

3.2 full CAD の実験結果

他システムとの実行時間の比較を 表 1 に示す．入力については 6 節を参照されたい．

deg は入力の多項式の最大次数，dim は変数の数，# は式の数，SyNRAC は数値数式 full CAD での実行時間．detchiage は Maple 上で実装した精度保証なし数値 full CAD の実行時間．() 内の数字はセルの数，QEPCAD に現れる F(xxx) は xxx 秒時にエラーで停止したことをあらわしている．

detchiage は例えば $\sqrt{2}$ を 1.4 と近似して処理を継続するため正確な解は返さないが，数値数式 CAD よりも必ずはやい結果になるので指標として追加した．

代数拡大体を primitive element で表現する数値処理を使用しない Maple 上での実装では，すべての問題で 3600 秒以上経過しても停止しなかったので，数値数式 CAD の高速化が実現できていることは確認できている．しかし，この実験結果では detchiage と比べても Mathematica と QEPCAD に大きく劣っている．これは，ex1 において，Mathmatica で 81，QEPCAD で 58 に対し，SyNRAC では 257 というようにセルの数が大きく異なることから partial CAD の実装による違いによるものと考えられる．

実際，ex1 は入力の式から $x \geq 0$ の場合の計算は不要であることが簡単にわかるが，$x \geq 0$ の場合に多くの時間を費しており $x < 0$ に限定した CAD を行うと 0.88(70) 秒で完了した．同様に ex2 の場合に $\alpha \geq 0 \wedge \beta \geq 0 \wedge 4(\alpha^2 + \beta^2) < 1$ の条件下で CAD を行うと 63.7(2512) 秒で QEPCAD よりも良い結果が得られた．

4 paritcal CAD

full CAD の場合には，多項式を入力としてすべてのセルを計算していた．本章では partial CAD では代数的命題文を入力として，持ち上げ段階における不要なセルの計算を削減することを考える．

以下では $F(x_1,\ldots,x_r)$ を限量記号がない代数的命題文，$F^* = Q_{s+1}x_{s+1}\cdots Q_r x_r(F(x_1,\ldots,x_r)), 0 \leq s < r$ を QE 問題の入力である限量記号がついた代数的命題文とする．

4.1 不要セルの計算削減手法

4.1.1 定義域の絞込み

前処理として，区間演算を利用して定義域の絞込みを行う．

例 1

$$F(x,y) = \exists x(0 \leq x \wedge x \leq 1 \wedge f(x,y) > 0)$$

このような問題において，$x < 0, x > 1$ に対しては $f(x,y)$ を評価することなく偽と判断できる．

この定義域の区間による絞込みにより不要な区間のサンプルポイントの一致性の比較やセルの持ち上げが不要となり高速化が実現できる．設計最適化問題では，それぞれの変数に区間制限をつけることが多いのでこの処理は有効に働くと考えられる．

4.1.2 真偽値の評価

入力の代数的命題文の真偽値はすべての変数の値が確定しなくても決定することがある．各変数の値が決定した時点で代数的命題文を評価し，持ち上げる必要があるセルを削減する．

例 2
$F(x,y) = F_1(x) \wedge F_2(x,y)$ のとき，$x_1 \in \mathbb{R}^1$ に対して $F_1(x_1)$ の真偽値が偽であることがわかれば，$F(x_1,y)$ の真偽値が偽であることが確定する．

記法 1
c を D_k $(1 \leq k \leq r)$ のセルとする．c を F^* で評価したときの真偽値を $v(c)$ と表記する．

c の level が r より小さい場合に，$v(c)$ は代数的命題文の真偽値を決定できないことがあるので，真偽値と未定義の 3 つの値のどれかを返す．

4.1.3 限量記号がついた変数

定義 3
セル $c \in D_k$ を持ち上げることにより生成される $(k+1)$-*level* のセル $c_1,\ldots,c_n$ を c の**子供**のセルとよび，c を $c_1,\ldots,c_n$ の**親**のセルとよぶ．

定理 4
セル $c \in D_k$ を持ち上げることにより生成される子供のセルを $c_1, \ldots, c_n$ とする．このとき，以下が成立する．

$$Q_{k+1} = \exists \quad \Rightarrow \quad v(c) = \bigvee_{i=1}^{n} v(c_i)$$

$$Q_{k+1} = \forall \quad \Rightarrow \quad v(c) = \bigwedge_{i=1}^{n} v(c_i)$$

限量記号がついた変数の場合，ある c_i の真偽値により c の真偽値が確定し，他のセルの評価，および持ち上げが不要であることがわかる．このことから，定義多項式の次数が低い等の計算量が低いと考えられるセルから計算していくことでセルの数，および計算量の削減が可能となる．

以下に示すアルゴリズム propagation は，入力のセル c の真偽値を用いて，親のセル p とその子供たちのセルの真偽値の情報を確定する．アルゴリズム propagation により確定された p の子供の真偽値は実際に計算した場合には異なる値になりうるが p の真偽値としては正くなるため partial CAD の結果には影響しない．

アルゴリズム 2
propagation(c)
入力: *c*: 真偽値が確定したセル

```
k ← level(c)
if k ≤ s then
    return
p ← {c の親のセル }
if v(p) = 未定義 then
    if v(c) = 真 and Q_k = ∃ then
        v(p および p の子孫) ← 真
    elif v(c) = 偽 and Q_k = ∀ then
        v(p および p の子孫) ← 偽
    elif v(p のすべての子供) ≠ 未定義
        if Q_k = ∃ then
            v(p) ← 偽
        else
            v(p) ← 真

    if v(p) ≠ 未定義 then
        propagation(p)
```

4.1.4 セルの選択順序

4.1.3 節から，限量記号がついていない変数についてはすべてのセルを評価せずに代数的命題文の真偽値が確定することがわかる．そのため計算量が少ないセルから評価していくようにすることで負荷の高い計算の削減をはかる．

アルゴリズム 3

cell_order(X, Y)

入力: X, Y: セル $(X \neq Y)$
出力: $X > Y$ なら正の値, $X < Y$ なら負の値

```
if deg(X) ≠ deg(Y) then
    return deg(X) − deg(Y)
end if
if level(X) ≠ level(Y) then
    return level(X) − level(Y)
end if
return index(X) − index(Y)
```

ここで $\deg(X)$ は X の定義多項式の次数, $\mathrm{level}(X)$ は X の level, $\mathrm{index}(X)$ は X の index を表している. index はセルを作成するときに付けられる位置情報で, 大小関係をつけるために使用している.

4.2 partial CAD アルゴリズム

partial CAD における持ち上げ段階のアルゴリズムを以下に示す. ここで使用している choose は引数のセルのリストから 4.1.4 節で定義した cell_order 関数により順序付けした最小のものを選択する関数である.

アルゴリズム 4

syn_lift(P, D_1, $R = \{R_1, \ldots, R_r\}$)

入力: P: 射影因子
　　D_1: $\mathbb{R}^1$ の CAD
　　R: 真偽値が自明でない区間
出力:

```
L ← D_1
while L is not empty do
    c ← choose(L)
    if v(c) ≠ 未定義 then
        propagation(c)
        continue
    k ← level(c)
    for p in P_k do
        r_p ← {r ∈ {pの分離区間のリスト} | r ∈ R_k } (真偽値が自明な分離区間の除去)
    for p, q in P do
        if r_p と r_q に重複する分離区間が存在する then
            if 重複する分離区間は同じ代数的数を表している then
                一方の分離区間をリストから削除
            else
                重複がなくなるまで分離区間の精度をあげる
```

表 2: 計算時間の比較（単位：秒）

ex	deg	dim	#	free	Mathematica	QEPCAD	SyNRAC (partial)	SyNRAC (full)
1	12	2	3	0	0.12(81)	0.83(58)	0.72(35)	24.5(257)
2-1	8	3	5	0	1.94(4853)	389 (6835)	78(3811)	>3600
2-2	12	2	3	0	2.61(5053)	1235(11653)	352(7797)	>3600
3	6	4	5	1	19.6(32606)	F(3446)	166(10185)	>3600
a-1	4	3	2	3	1.49(470)	F(306)	> 3600	2534(8751)
a-2	4	3	1	3	2.19(403)	F(270)	> 3600	> 3600
a-3	3	4	1	4	1.07(1255)	3.68(1718)	38.3(635)	508(6115)
a-4	4	3	1	3	1.23(210)	0.34(464)	5.14(583)	7.11(763)
b-1	4	4	1	4	15.7(8859)	F(1272)	59.4(6777)	114(13089)

$C \leftarrow \{\ r_p$ から構成されるセルのリスト $\}$
$D_{k+1} \leftarrow D_{k+1} \cup C$
$L \leftarrow L \cup C$
return $D = \{D_1, \ldots, D_r\}$

4.3 partial CAD の実験結果

partial CAD での他システムとの実行時間の比較を表 2 に示す. full CAD の場合に比べ不要なセルの計算の削減により計算時間の短縮が実現できていることが確認できる. またいくつかの問題では数値手法を使用していない QEPCAD よりも効率よく計算できていることも確認できる.

5 まとめ

partial CAD の実装により数値数式手法を使った CAD の実装の有効性を確認することができた.

今後は, QE 問題の結果を構築する式構成 (soluation formula construction) アルゴリズムの実装を予定している.

6 例

実験で使用した問題 [1] を以下に示す.

ex1

$$\forall x, y \in \mathbb{R}\left(x < 0 \wedge x^2 + y^2 < \frac{99438}{100000}\right) \Rightarrow R(x + \mathrm{i}y)R(x - \mathrm{i}y) < 1$$

ここで

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

ex2

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

2-1. Show that

$$\forall\alpha,\beta,C_2\in\mathbb{R}(\alpha\geq 0\wedge\beta\geq 0\wedge 4(\alpha^2+\beta^2)<1)\Rightarrow(B\leq 0\vee D\leq 0).$$

2-2. Show that

$$\forall\alpha,\beta,C_2\in\mathbb{R}(0\leq\alpha\leq 1\wedge 0\leq\beta\leq 1)\Rightarrow A\leq 0\wedge C\leq 0\wedge(B\leq 0\vee D\leq 0).$$

ex3

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

a-1

$$
\begin{aligned}
&-363+177x_1-125x_1x_2^2+444x_2^3-766x_2x_3+477x_1x_2x_3+447x_3^2=0 \quad\wedge \\
&864-55x_1^3x_2-491x_2^2+959x_1x_2^3-465x_1x_3^2>0.
\end{aligned}
$$

a-2

$$566+180x_1^2x_2^2+264x_1^2x_3+919x_2^2x_3+941x_2^3x_3+481x_1x_2x_3^2-519x_2x_3^3\geq 0.$$

a-3

$$634-508x_3+896x_3^3-342x1x_2x_4-462x_2x_3x_4+144x_2x_4^2\geq 0.$$

a-4

$$-120+38x_2^3x_4+207x_3x_4+872x_3^2x_4+165x_2x_3^2x_4+672x_2x_3x_4^2+87x_3^2x_4^2=0.$$

b-1

$$446+511x_3-190x_1^3x_3+132x_2^2x_3+968x_1^2x_2x_4+226x_1x_2x_3x_4+645x_2x_3x_4^2>0.$$

参 考 文 献

[1] Adam W. Strzebonski: Cylindrical Algebraic Decomposition using validated numerics, *Journal of Symbolic Computation*, 41, 2006, 1021–1038

[2] G. E. Collins: Quantifier Elimination for Real Closed Fields by Cylindrical Algebraic Decomposition, *LNCS 33*, 1975, 134–183

[3] G. E. Collins, H. Hong: Partial cylindrical algebraic decomposition for quantifier elimination, *Journal of Symbolic Computation*, 12, 1991, 299–328

[4] G. E. Collins, J. R. Johnson, W. Krandick: Interval Arithmetic in Cylindrical Algebraic Decomposition, *Journal of Symbolic Computation*, 34, 2002, 145–157

[5] J. D. Dora, C. Dicrescenzo, D. Duval: About a new method for computing in algebraic number fields, *LNCS 204*, 1985, 289–290