論　文

冠頭標準形有理数プレスブルガー文の真偽判定アルゴリズムの提案

柴田　直樹[†]　　岡野　浩三[†]　　東野　輝夫[†]　　谷口　健一[†]

A Decision Algorithm for Prenex Normal Form Rational Presburger Sentences

Naoki SHIBATA[†], Kozo OKANO[†], Teruo HIGASHINO[†], and Kenichi TANIGUCHI[†]

あらまし　本論文では，加算をもつ有理数の理論（有理数変数，有理数定数，$+,-,=,<,\wedge,\vee,\forall,\exists$ からなる理論）の冠頭標準形閉論理式に対する，時間計算量が $r\cdot\alpha^{\beta d}n^{\gamma d(b+1)^a}$（$\alpha,\beta,\gamma$ は定数，n は入力の式に含まれる不等式の個数，d は変数の個数，a は限定子交替数，b は同じ限定子が続く最大の個数，r は入力の式の各係数と定数の分母，分子のビット数）の計算幾何学の手法を利用した真偽判定アルゴリズムを提案する．従来知られていた真偽判定アルゴリズムの最良の時間計算量は $r\cdot 5^{\epsilon d}n^{\zeta d(2b+1)^a}$（$\epsilon,\zeta$ は定数）である．

キーワード　加算をもつ有理数の理論，真偽判定アルゴリズム，組合せ幾何学，投影

1. まえがき

加算をもつ有理数の理論（有理数変数，有理数定数，$+,-,=,<,\wedge,\vee,\forall,\exists$ からなる理論）の冠頭形の閉論理式（以降 PRP 文と呼ぶ）の真偽判定ルーチンはプロトコルのテスト，ハードウェアのタイミング検証などに利用される [1].

加算をもつ有理数の理論の時間計算量，領域計算量については数々の研究がなされている [2]〜[5]．時間計算量の下界は Fischer と Rabin らによって非決定性 $O(2^{cl})$（c は定数，l は入力の長さ）という結果が得られている [6]．上界に関しては，文献 [7] で Ferrante と Rackoff が時間計算量 $O(2^{2^{dl}})$（d は定数）の決定性の PRP 文真偽判定アルゴリズムを提案している．この時間計算量を入力の PRP 文に含まれる不等式の個数 n，変数の個数 d，限定子交替数 a，同じ限定子が続く最大の個数 b，入力の式の各係数と定数の分母，分子の最大のビット数 r を用いて精密に評価すると，$r5^{\epsilon d}n^{\zeta d(2b+1)^a}$（$\epsilon,\zeta$ は定数）となる．このアルゴリズムは筆者らの知る限り最も時間計算量が少ない．本論文では時間計算量が $r\alpha^{\beta d}n^{\gamma d(b+1)^a}$（$\alpha,\beta,\gamma$ は定数）の計算幾何学の手法を利用した PRP 文真偽判定アルゴリズムを提案する．双方のアルゴリズムの計算量で支配的なのは 2 重指数の部分であり，提案するアルゴリズムの計算量は Ferrante, Rackoff のアルゴリズムのものに比べてこの部分が改善されている．

アルゴリズムの概要は次のとおりである．入力の PRP 文に含まれる不等式の集合から d 次元のアレンジメント（d 次元空間を平面で分割して得られる部分空間すべての集合）を作る．ここで分割で得られる個々の部分空間（フェースと呼ぶ）内では PRP 文の母式の真偽は変わらない，という性質がある．そこで，まず，各フェースに母式と真偽が同じになるように真偽を割り当て，次に，内側から連続する同じ限定子に束縛された変数分だけ，このアレンジメントをより小さな次元の空間へ（限定子が存在記号なら真偽値の論理和を，全称記号なら論理積をとって）縮体させる（この操作を投影と呼ぶ）．最後に得られた 0 次元のアレンジメントから式全体の真偽を判定する．

2. アルゴリズム

2.1 アルゴリズムの概要

以下では，$F=\forall x\exists y\exists z\{x\geqq 0\wedge y\geqq 0\wedge[(z-x\geqq 0\wedge y+z\leqq 3/4)\vee(x\leqq 1/4\wedge z\leqq 1/4)]\}$ の真偽判定の様子を例に，このアルゴリズムの真偽判定の概要について説明する．また，必要な概念を適宜定義していく．

有理数変数，有理数定数，$+,-,=,<,\wedge,\vee,(,)$ からなる論理式を PRP 式と呼ぶことにする．冠頭形の閉論理式である PRP 式を特に PRP 文と呼ぶ．

最初に入力の PRP 文の母式から空間を分割するア

[†] 大阪大学大学院基礎工学研究科情報数理系専攻，豊中市
Division of Informatics and Mathematical Science, Graduate School of Engineering Science, Osaka University, Machikaneyama 1–3, Toyonaka-shi, 560–8531 Japan

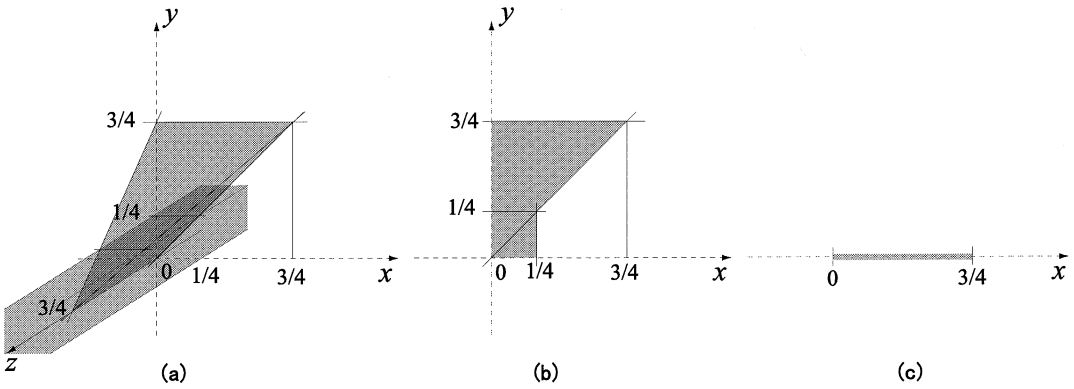

図 1　アルゴリズムの適用によって変化していくアレンジメント
Fig. 1　Transformation of an arrangement in execution of the algorithm.

レンジメントを作る．まず，3 次元の場合について必要な概念を定義する．それぞれの平面で空間を平面の一方，もう一方，そして平面に含まれる空間の三つに分割することを考える．これにより空間は，点，両端を含まない線分，外周を含まない多角形，外面を含まない多面体に分割される．こうして分割されたそれぞれの部分をフェースと呼ぶ．点，両端を含まない線分，外周を含まない多角形，外面を含まない多面体をそれぞれ 0-フェース，1-フェース，2-フェース，3-フェースと呼ぶ．空間全体を平面の集合 H でフェースの集合 S に分割したとき，S を H のアレンジメントと呼ぶ．また，点を 0-フラット，直線を 1-フラット，平面を 2-フラット，空間全体を 3-フラットと呼ぶ．以上 3 次元で説明したが一般に d 次元に拡張して $(d-1)$-フェース，$(d-1)$-フラット，などという用語も使う．d 次元空間における $(d-1)$-フラットを超平面と呼ぶ．PRP 文 F には三つの変数が含まれるので，3 次元の図で考える．F の母式である PRP 式 $E = x \geqq 0 \wedge y \geqq 0 \wedge [(z - x \geqq 0 \wedge y + z \leqq 3/4) \vee (x \leqq 1/4 \wedge z \leqq 1/4)]$ が真になる領域は図 1 (a) の灰色の部分で表され，この領域は E 中の各不等式の表す平面で 3 次元空間を分割してできるアレンジメントに含まれるフェースの部分集合である．

以上の事柄を一般次元に拡張して定義すると，次のようになる．以下の定義は文献 [9] に従う．

[定義 1]（軸に垂直なフラット，平行なフラット） 直交座標 $v_1, \ldots, v_d$ を用いたときフラット fl が v_i 軸に垂直なある直線を含んでいるならば，fl は v_i 軸に垂直であるという．フラット fl が v_i 軸に平行なある直線を含んでいるならば，fl は v_i 軸に平行であるという．　□

[定義 2]（k-フェース） 一般性を失うことなく v_d 軸に垂直でない超平面 h を考える．h 上の任意の点 $x = (x_1, x_2, \cdots, x_d)$ について $x_d = \eta_d + \sum_{i=1}^{d-1} \eta_i x_i$ が成り立つというような実数の組 $\eta_1, \cdots, \eta_d$ がただ一つ存在する．点 $p = (\pi_1, \cdots, \pi_d)$ に対し π_d が $\eta_d + \sum_{i=1}^{d-1} \eta_i \pi_i$ より大きい，等しい，小さい場合にそれぞれ，p は h より上にある，h に乗っている，h より下にあるということにする．h^+ は h より上にある点の集合を，h^- は h より下にある点の集合を表す．超平面の集合 $H = \{h_1, \cdots, h_n\}$ に含まれるどの超平面もすべて v_d 軸に垂直でないとする．超平面 h_i 及び点 p に対し $v_i(p)$ を以下のように定める．

$$v_i(p) = \begin{cases} +1 & (p \in h_i^+) \\ 0 & (p \in h_i) \\ -1 & (p \in h_i^-) \end{cases}$$

$(v_1(p), \cdots, v_n(p))$ の値が同じ点 p の集合をフェースという．k-フラットに含まれて，$(k-1)$-フラットに含まれないフェースを k-フェースと呼ぶ．特に 0-フェースを頂点と呼ぶ．　□

[定義 3]（アレンジメント） d 次元空間内の超平面の有限集合 H は d 次元以下の種々の次元のフェースに空間を分割する．このフェースの集合を H によって E^d を分割してできる（又は単に H の）アレンジメントと呼ぶ．H のアレンジメント A に含まれるフェースを A を構成するフェースと呼ぶ．ある k-フラット fl がアレンジメント A を構成する k-フェース f を

含むとき，fl は A から定まるフラットであるという．□

次に，E より得られたアレンジメント A の各フェース f に，f に含まれる任意の点の座標を母式に代入して得られる真偽値を割り当てる（この操作を A に E の真偽を割り当てると呼ぶ）．これによって，空間の任意の点 p に対し，p の座標を E に代入して得られる真偽値と p が含まれるフェースに割り当てられた真偽値が一致する．図 1 (a) 中の灰色のフェースが，真が割り当てられたフェースである．

$\exists z$ を消去する操作を行う．これは E を真にする z が存在する x, y の値の領域を xy 平面上に図示する操作である．これによって得られた図が図 1 (b) である．これは，図 1 (a) に z 軸に並行な光を当ててできる真になる部分の影である．この影の部分を表す真偽を割り当てたアレンジメントを真偽を割り当てたアレンジメントの投影と呼ぶ．

投影を作る操作は図 1 (a) 中の 1-フラット（直線）の xy 平面への投影（直線）すべての集合から 2 次元のアレンジメントを作り，図 1 (a) の真になる部分の影に含まれるフェースにのみ真を割り当てることで行う．

[定義 4]（点の投影） 点 $p = (V_1, V_2, \ldots, V_d)$ の $(v_1, v_2, \ldots, v_{d-s})$ 空間への投影は $(V_1, V_2, \ldots, V_{d-s})$ である．□

[定義 5]（フェースの投影） フェース f の $(v_1, v_2, \ldots, v_{d-s})$ 空間への投影は f に含まれる点の $(v_1, v_2, \ldots, v_{d-s})$ 空間への投影すべての集合である．□

[定義 6]（フラットの投影） フラット fl の $(v_1, v_2, \ldots, v_{d-s})$ 空間への投影は fl に含まれる点の $(v_1, v_2, \ldots, v_{d-s})$ 空間への投影すべての集合である．□

[定義 7]（アレンジメントの投影） アレンジメント B の $(v_1, v_2, \ldots, v_{d-s})$ 空間への投影は，B から定まる $(d-1-s)$-フラット $(v_1, v_2, \ldots, v_{d-s})$ 空間での超平面）の $(v_1, v_2, \ldots, v_{d-s})$ 空間への投影 fl' のうち fl' が $(d-1-s)$-フラットとなる fl' の集合のアレンジメントである．□

一般に，あるアレンジメント A に属するフェース f の投影に対応する集合 $\mathcal{F}$ があって，次の条件を満たす：$\mathcal{F}$ は A の投影 **pr** A 上のフェースの集合であり，かつ f の投影に含まれる点の集合と $\mathcal{F}$ に含まれるフェースから構成される点の集合は一致する（**3.3** の補題 1 参照）．

[定義 8]（真偽を割り当てたアレンジメントの限定子 Q による投影） 真偽を割り当てたアレンジメント A の $(v_1, v_2, \ldots, v_{d-s})$ 空間への Q による投影 **pr** A は定義 7 の A の $(v_1, v_2, \ldots, v_{d-s})$ 空間への投影 A' に属する各フェースに次のように真偽を割り当てたものである：

$Q =$‘∃’ の場合　A' に属するフェース f' に対して，A に属する真が割り当てられたあるフェース f が存在して f の投影が f' を含むときかつそのときのみ，f' の値は真である．

$Q =$‘∀’ の場合　A' に属するフェース f' に対して，A に属する偽が割り当てられたあるフェース f が存在して f の投影が f' を含むときかつそのときのみ，f' の値は偽である．□

入力の論理式の限定子の並びは $\forall x \exists y \exists z$ なので，同様に $\exists y$ を消去して，1 次元の真偽を割り当てたアレンジメントを得る．これは図 1 (c) で表される．実際のアルゴリズムでは連続する同じ限定子で束縛された変数は一度に消去する．すなわち，図 1 (a) から直接図 1 (c) を得る．

$\exists$ を消去する操作が，真になる領域に座標軸に並行な光を当ててできる影を真とする領域であるのに対し，$\forall$ を消去する操作は，偽になる領域に座標軸に並行な光を当ててできる影を偽とする領域である（定義 8, **2.2.3** 参照）．残った $\forall x$ を消去し，0 次元の真偽が割り当てられたアレンジメントを得る．これは偽が割り当てられた一つの頂点からなる．したがって，式全体は偽と判定される．

2.2 アルゴリズムの詳細

まずサブルーチンとなる ARRANGE, ASSIGN, PROJECT について順に述べる．ここで，ARRANGE はアレンジメントを表すデータ構造を作るものであり，メインルーチン MAIN と PROJECT に用いられるサブルーチンである．ASSIGN はアレンジメントと PRP 式から真偽を割り当てたアレンジメントを作る．PROJECT は真偽を割り当てたアレンジメントの投影を作るものである．**2.2.4** において主ルーチンである MAIN について述べる．

2.2.1 ARRANGE

d 次元空間上の超平面を表すデータ構造として，超平面上にある d 個のアフィン独立な点（$P = \{\vec{p_0}, \cdots, \vec{p_k}\}$ を d 次元空間内の有限点集合とする．$x = \Sigma_{i=0}^{k} \lambda_i \vec{p_i}$ かつ $\Sigma_{i=0}^{k} \lambda_i = 1$ であるとき，x は P のアフィン結合であるという．$p_i \in P$ が $P - \{p_i\}$ のアフィン結合

になるような p_i が存在しないとき，P はアフィン独立であるという)(注1)の座標の集合を使う．

アレンジメントのデータ構造を次に述べる．

それぞれのフェース f は定数サイズの付加情報と f に接続する次元が一つ上のフェースへのポインタ群，次元が一つ下のフェースへのポインタ群で表される．図2のアレンジメント全体のデータは図3のようになる．ただし，図3のそれぞれの長方形がフェースを表す．フェース同士を結ぶ枝は，結ばれた両方のフェースが接続していることを表す．

付加情報には次のような情報が含まれる．

- フェースの次元数
- 0-フェースの場合は，頂点の座標
- フェースに割り当てられた真偽値

なお有界でないフェース（図2のフェースAG等）については十分大きな値を頂点の座標として用いる（例えばAの座標として半直線GA上の十分Gから離れた点をとる）．

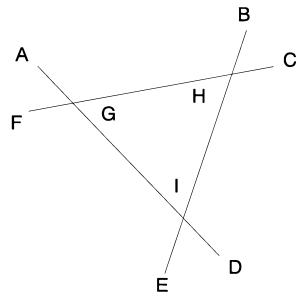

図2　アレンジメントの例
Fig. 2　An example of an arrangement.

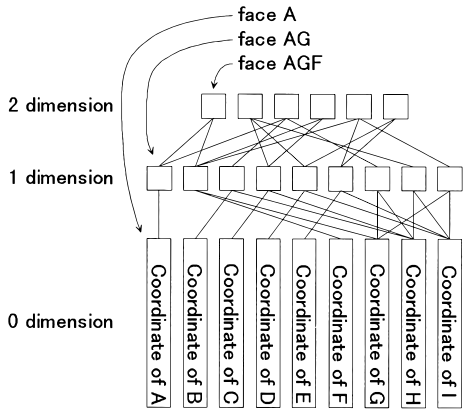

図3　図2のアレンジメントに対するデータ構造
Fig. 3　The data structure of Fig. 2.

Algorithm ARRANGE
◇ 入力 超平面の集合 H
◇ 出力 H のアレンジメント

アルゴリズムは文献[9]第7章参照．

2.2.2　ASSIGN

サブルーチン ASSIGN は入力のアレンジメントに入力の PRP 式の真偽を割り当てるルーチンである．以下に用いる関数 INNER は（頂点の集合で表される）フェース f から，f の内部にある点（例えば重心）の座標を求めるものである．

Algorithm ASSIGN
◇ 入力 与えられた入力論理式の母式 E, E の不等式の表す超平面集合を含む超平面集合のアレンジメント A
◇ 出力 A に E の真偽を割り当てたアレンジメント
▷ A に含まれるフェースに対して E の真偽を割り当てる
1　**for** each $f \in A$
2　　f に E(INNER(f)) の真偽値を割り当てる；
3　**next**
4　**return** A;
5　**end**

2.2.3　PROJECT

サブルーチン PROJECT はアレンジメントの投影のアレンジメントを作るルーチンである．以下に用いる関数 EXT はフェースからフラットを作るものである．

Algorithm PROJECT
◇ 入力 次の条件を満たす真偽を割り当てたアレンジメント A：すべての座標軸に対してその軸に垂直でありかつ A から定まる超平面 hp がある，整数 s，限定子 q
◇ 出力 A の投影
1　$S := \{\mathrm{EXT}(f) | f \in A, f$は $(d-1-s)$-フェース $\}$;
2　$A' := \mathrm{ARRANGE}(S)$;
3　**if** $q =$ '∃' **then** $b :=$ true **else** $b :=$ false;
▷ 各 $f' \in A'$ に対し最初に $\neg b$ に初期設定する
4　**for** each $f' \in A'$ **do** f' に $\neg b$ を割り当てる **next**
▷ A' に含まれるフェースに対して，真偽を割り当てる
5　**for** $f \in A$ s.t. f に b が割り当てられている
6　　**for** each $f' \in A'$
7　　$\vec{x_1}, \ldots, \vec{x_k}$ を f の頂点の投影とする．
▷ f' に含まれる適当な1点 (INNER(f')) が $sh\ f$ に含まれるかどうか調べる
8　　**if** (INNER(f') $= \sum_{m=1}^{k} a_m \vec{x_m} \wedge 0 < a_m \wedge a_m < 1 \wedge \sum_{m=1}^{k} a_m = 1$) を満たす $a_1, \ldots, a_k$ が存在する
9　　**then** f' に b を割り当てる．**endif**
10　　**next**
11　**next**
12　**return** A';
13　**end**

(注1)：場合によってはそれらに加えて超平面上のいくつかの点．

2.2.4 MAIN

MAIN は提案するアルゴリズムのメインルーチンで，与えられた PRP 文の真偽を判定する．以下に用いる関数 AFFINE は不等式 in の表す超平面 h の上にあるような点であって，かつ h を表現する十分な数のアフィン独立な点の集合に in を変換するものである．

Algorithm MAIN
◇ 入力：PRP 文 F
◇ 出力：F の真偽値

```
1     入力の PRP 文の限定子の並びを Q_1,...,Q_d とする．
      それぞれの限定子が束縛する変数を v_1,...,v_d とする．
2     S := (F に含まれる不等式の集合) ⋃ (⋃_{i=1}^{d} {v_i = 0});
3     H := ∅;
4     for each h ∈ S
5         t := AFFINE(h);
6         H := H ⋃ {t};
7     next
8     A := ARRANGE(H);
9     A := ASSIGN (A, F の母式) ;
      ▷ i は残っている限定子の数
10    i := d;
11    while i ≧ 1 do
12
13        if Q_i = '∀' then q := '∃' else q := '∀'
14
          ▷ 内側の連続した Q_i の個数を s 個に代入.
15        if q ∈ {Q_1,...,Q_i} then
16            determine s, s.t. Q_{i-s} = q∧ for each
              i - s + 1 ≦ k ≦ i, Q_k ≠ q;
17        else
18            s := i;
19        endif
20
21        A := PROJECT(A, s, Q_i);
22
23        i := i - s;
24    endwhile
25
      ▷ A は一つの点だけで構成される
26    return A の原点の真偽値 ;
27 end
```

3. 正当性の証明と時間計算量の解析

ここでは，各ルーチンの仕様を再掲し，それぞれの正当性の証明，時間計算量，及びその解析について述べる．

以下，明示しない限り左の表記で右の事柄を表す．

- n　入力の PRP 文に含まれる不等式の数
- d　入力の PRP 文に含まれる異なる変数の数
- l　入力のサイズ
- a　入力の PRP 文の限定子交替数
- b　入力の PRP 文の同じ限定子が続く最大の個数

3.1 ARRANGE

仕様：

入力　超平面の集合 H
出力　H のアレンジメント A

正当性の証明：文献 [9] 参照.

時間計算量：H に含まれる各超平面は超平面を表す頂点の集合の各座標の分母，分子がそれぞれたかだか g ビットの整数で表されているとする．時間計算量及び出力のサイズは $O(g \cdot \gamma^d |H|^d)$（ただし γ は定数）[9].

時間計算量の解析：アレンジメントに含まれるフェースの数は $O(|H|^d)$ 個 [9]．出力の各頂点の座標は入力の超平面を表現するための各有理数を係数とする d 元連立 1 次方程式の解である．したがって，出力の各頂点の座標はたかだか $g \cdot \gamma^d$ ビットの整数を分母，分子にもつ有理数である．時間計算量はフェースの数と出力の各頂点の座標のビット数の積であるから，時間計算量及び出力のサイズは $O(g \cdot \gamma^d |H|^d)$ となる．

3.2 ASSIGN

仕様：

入力　PRP 式 E, E の不等式の表す超平面集合を含む超平面集合のアレンジメント A
出力　E の真偽を割り当てたアレンジメント A'

正当性の証明：定義 3，PRP 式の定義，真偽を割り当てたアレンジメントの定義から明らか．

時間計算量：A のサイズ l に対して時間計算量は l の多項式オーダ，出力のサイズは $O(l)$.

時間計算量の解析：1 行目から 3 行目までのループを 1 回回るのに必要な時間計算量は l に対してたかだか多項式オーダである．ループを回る回数は $O(l)$ であるから，1 行目から 3 行目までの時間計算量は l の多項式オーダとなる．したがって，このアルゴリズム全体の時間計算量 l の多項式オーダである．出力のサイズは，A のサイズと等しいので，$O(l)$ である．

3.3 PROJECT

仕様：

入力　各軸に垂直な超平面が定まる，真偽を割り当てたアレンジメント A，アレンジメントを縮退させる次元の数 s，限定子 Q
出力　A の次元を d' とするとき，A の $d'-s$ 次元空間への限定子 Q による投影 A'

正当性の証明：後述の補題 1 より，f' が **sh** f に含まれることと f' に含まれる 1 点が **sh** f に含まれることは等価である．

8 行目の正当性について述べる．凸多面体 C の頂点の座標ベクトルの集合を $\{\vec{x}_1,\ldots,\vec{x}_k\}$ とするとき，ある座標 $\vec{p}$ が C に含まれることは，式 (1) を満たすような $a_1,\ldots,a_k$ が存在することと同値である．ここで，$\{\vec{x}_1,\ldots,\vec{x}_k\}$ の代わりに，$\{\vec{x}_1,\ldots,\vec{x}_k,\vec{y}_1,\ldots,\vec{y}_l\}$（ただし，$\vec{y}_1,\ldots,\vec{y}_m$ はすべて C に含まれる）に対しても同様のことがいえる（証明は文献 [11] を参照）．

$$\vec{p}=\sum_{m=1}^{k} a_m\vec{x_m} \wedge 0<a_m<1 \wedge \sum_{m=1}^{k} a_m=1 \qquad (1)$$

したがって，成り立つ．その他に関しては定義 4，定義 5，定義 9 から明らか．

時間計算量：A を構成するフェースの数を m, A の各頂点の座標がたかだか r ビットの整数を分母，分子にもつ有理数，α,η,θ を定数とするとき，出力 A' のデータサイズはたかだか $r\alpha^{b+1}m^{b+1}$．全体の時間計算量はたかだか $r\alpha^{\eta(b+1)}m^{\theta(b+1)}$．

時間計算量の解析：A を H のアレンジメントとする．A を構成するフェースの数は $O(|H|^d)$ である．また，A から定まる $d-s-1$-フラットの数はたかだか ${}_{|H|}C_{s+1}$ である．$s\leqq b$ より，A から定まる $d-s-1$-フラットの数はたかだか $|H|^{b+1}$ である．A の投影 **pr** A は A から定まる $(d-s-1)$-フラットすべてと同じ数の超平面の集合 H' のアレンジメントであるので，**pr** A にはたかだか $O(|H|^d)^{b+1}$ 個のフェースが含まれている．したがって，入力 A を構成するフェースの数 $O(|H|^d)$ を m とおくと，出力 A' を構成するフェースの数はたかだか m^{b+1} となる．

A の各頂点の座標がたかだか r ビットの整数を分母，分子にもつ有理数であるとすると，**pr** A の各頂点の座標の分母，分子のビット数はたかだか $r\alpha^{b+1}$ である．

8 行目の処理にかかる時間に関しては，頂点の数がたかだか m^{b+1} 個なので，解が存在するかどうかを判定する式中の不等式と等式の数はたかだか $d+2m^{b+1}+1$ 個，変数の数はたかだか m^{b+1} 個になる．これは線形計画問題なので，式の判定にかかる時間は $O((m^{b+1})(d+2m^{b+1}+1)^3r)$ となる [10]．

5 行目から 11 行目までの処理は A に属するフェースの数が m, A' に属するフェースの数が m^{b+1} であるから，m^{b+2} の多項式オーダとなる．

したがって，全体の時間計算量は $r\alpha^{\eta(b+1)}m^{\theta(b+1)}$ となる．

［補題 1］ **pr** A を構成するフェースの集合を S とする．各座標軸に対してその座標軸に平行でない超平面がアレンジメント A から定まる場合に(注 2)，A を構成する各フェース f の $(v_1,\ldots,v_{d-s})$ 空間への投影 **sh** f に対し，**sh** f と（点の集合として）一致する S の部分集合が存在する． □

補題 1 の証明は文献 [11] を参照．

3.4 MAIN

仕様：

入力　PRP 文 F

出力　F の真偽値

正当性の証明：MAIN の正当性は下の各命題が証明されれば十分である．

［定義 9］（関数 $\mathcal{VAL}$） 関数 $\mathcal{VAL}$ は真偽を割り当てたアレンジメント A と A に属する 1 点の座標 p を引数に取り，p が含まれる A に属するフェースに割り当てられた真偽値を返す．

［定義 10］（関数 $\mathcal{FML}$） A を真偽を割り当てたアレンジメント，b をブール変数，$Q_1,\ldots,Q_i$ を限定子とする．関数 $\mathcal{FML}$ を次のように定義する．

$$\mathcal{FML}(A,(Q_1,\ldots,Q_i)) = Q_1v_1,\ldots,Q_iv_i\mathcal{VAL}(A,(v_1,\ldots,v_i))$$

［命題 1］ 最初に MAIN の 12 行目に到達したときに $\mathcal{FML}(A,(Q_1,\ldots,Q_i))$ の真偽が F の真偽と一致している，かつ A に各座標軸に垂直な超平面が含まれていること．

［命題 2］ MAIN の 12 行目において $\mathcal{FML}(A,(Q_1,\ldots,Q_i))$ の真偽が F の真偽と一致している，かつ A に各座標軸に垂直な超平面が含まれていること．

［命題 3］ 最初に MAIN の 25 行目に到達したとき A が一つの点だけで構成され，$i=0$ であり，t が入力の PRP 文の真偽と一致していること．

［命題 4］ while 文からいつか抜け出ること．

（命題 1 の証明） アレンジメント A 内の，同一のフェースに属する座標では，E の真偽が変わらないことと，関数 $\mathcal{FML}$ の定義，MAIN の 2 行目で A に各軸に垂直な超平面を付加していることから，題意は成立する． □

(注 2)：この条件が必要である理由を以下に例を挙げて説明する．3 次元空間に平面 $x=0$ のみからなる平面の集合のアレンジメントがあり，このアレンジメントの (x,y) 平面への投影をとる場合に投影のアレンジメントから直線 $x=0$ が定まるようにしたい．もとのアレンジメントに z 軸に平行でない平面 $z=0$ を加えればこの要求が満たされる．

（命題 2 の証明） a_1, a_2 をそれぞれ 12 行目，22 行目の時点の $\mathcal{FML}(A,(Q_1,\ldots,Q_i))$ の値とする．サブルーチン PROJECT の正当性と補題 2 より $a_1 = a_2$ が成り立つ．

各座標軸に垂直な超平面をアレンジメントから定まるフラットとしてもつアレンジメントの投影は各座標軸に垂直な超平面をアレンジメントから定まるフラットとしてもつことは補題 1 より明らか．したがって，題意が成立する． □

［補題 2］ 各軸に垂直な超平面を定めるアレンジメント A の限定子 Q による $(v_1,\ldots,v_{i-s})$ 空間への投影を $\mathbf{pr}\ A$ とすると，任意の $V_1,\ldots,V_{i-s}$ に対し下の式が成り立つ．

$$\begin{aligned}&\mathcal{VAL}(\mathbf{pr}\ A,(V_1,\ldots,V_{i-s}))\\&\quad= Q\ v_{i-s+1},\ldots,Q\ v_d\\&\qquad\mathcal{VAL}(A,(V_1,\ldots,V_{i-s},v_{i-s+1},\ldots,v_d))\end{aligned}$$

（証明） 補題 1 より，$\mathbf{sh}\ f$ が含む $\mathbf{pr}\ A$ 上のフェースの集合を $\mathcal{F}$ とすると，$\mathcal{F}$ に含まれるフェースに含まれる点すべての集合と，$\mathbf{sh}\ f$ は一致する．$Q =$‘∃’ の場合について述べる．アルゴリズムより f に真が割り当てられていれば，$\mathcal{F}$ に含まれるフェースには真が割り当てられるので，真が割り当てられたフェース f に含まれる点 p に対して，$\mathbf{sh}\ p$ は真が割り当てられたフェースに含まれている．f に偽が割り当てられていれば，$\mathcal{F}$ に含まれる各フェース f' に真が割り当てられる場合と偽が割り当てられる場合がある．f' に偽が割り当てられる場合は $f' \subseteq \mathbf{sh}\ f''$ となるような A を構成するフェース f'' にはすべて偽が割り当てられている．f' に真が割り当てられている場合，ある f'' に対して真が割り当てられている．$\mathbf{sh}\ p'$ が偽が割り当てられたフェースに含まれていれば，$\mathbf{sh}\ p'' = \mathbf{sh}\ p'$ となるような p'' はすべて偽が割り当てられたフェースに含まれている．

また，$Q =$‘∀’ の場合にも同様に証明できる．

以上により，題意が成り立つ． □

（命題 3 の証明） 最後に 22 行目を通った時点で A は 0 次元である． □

（命題 4 の証明） 22 行目の時点で $s > 0$ である．したがって，while 文を 1 回回るごとに必ず i の値は減少する．よって，題意は示される． □

時間計算量：入力の式の各係数と定数の分母，分子のビット数 r，定数 α,β,γ に対して，$r\cdot\alpha^{\beta d}n^{\gamma d(b+1)^a}$．

時間計算量の解析：5 行目の時間計算量はたかだか l に対して多項式オーダである．8 行目の時間計算量及び代入される A のサイズは $O(r\cdot\gamma^d n^d)$ である．9 行目の ASSIGN の返り値 A のサイズは $O(l\cdot\gamma^d n^d)$ となる．

11 行目から 24 行目までのループの時間計算量について考える．12 行目の時点での A を構成するフェースの数を m，各頂点の座標がたかだか r' ビットの整数を分母，分子にもつ有理数で表されているとする．21 行目の時間計算量は定数 α,η,θ に対し，$O(r'\alpha^{\eta(b+1)}m^{\theta(b+1)})$ であり，返り値の A のサイズは $r'\alpha^{b+1}m^{b+1}$ である．ループに入る前の A に含まれるフェースの個数を m_s，A の各頂点の座標の分母，分子のビット数を r_s とおくと，ループを回る回数は a であるので，a 回目のループの 21 行目の PROJECT の時間計算量は定 $r_s\alpha^{\eta ab}m_s^{\gamma d(b+1)^a}$ である．$d = O(ab)$ より，全体の時間計算量は定数 α,β に対して，$r\alpha^{\beta d}n^{\gamma d(b+1)^a}$ となる．

4. 評価実験と考察

提案するアルゴリズムによる判定ルーチンと Ferrante, Rackoff のアルゴリズムによる判定ルーチンのそれぞれを実装し，評価実験を行った．それぞれのアルゴリズムの実装に際して，実装上の最適化は一切行わなかった．各測定は SunOS 5.6 の動作するワークステーション（UltraSPARC-II 248 MHz, メモリ 512 M バイト，スワップ領域 1 G バイト）上で行った．

提案するアルゴリズムの **4.** で述べた時間計算量オーダが実際の判定時間とどれほど一致するか，また Ferrante, Rackoff のアルゴリズムとの実際の判定時間の差について調べた．

双方のアルゴリズムは入力の式中に出てくる各不等式の形と限定子にのみ計算量が依存し，各不等式がどのように論理演算子で結合されているかには依存しないという特徴がある．したがって，入力の PRP 文の母式の形をいくつかの不等式の論理積に限定しても，そうでない場合と同じ結果が得られる．

最初に，入力の PRP 文の係数と定数のビット長のみを変化させ，判定時間の変化を調べた．

実験では下の式の係数と定数 A から L までの分子，分母それぞれのビット長を 5 ビットから 30 ビットまで変化させ，判定時間を測定した．係数と定数 A か

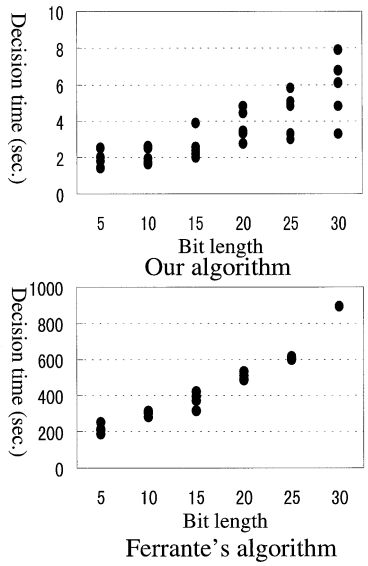

図 4　入力の係数と定数のビット長と判定時間
Fig. 4　Relation between bit length of coefficients and decision time.

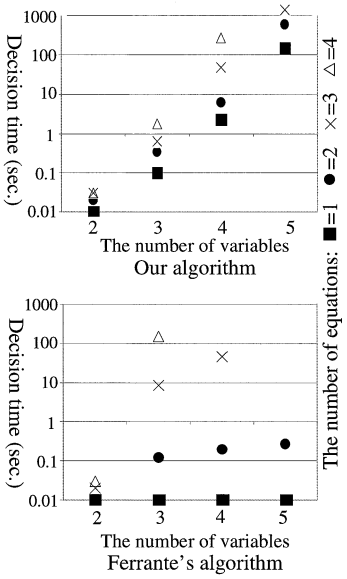

図 5　限定子交替数が 1 の場合の判定時間
Fig. 5　Decision times under the condition : the number of quantifier alternation is 1.

ら L をランダムで生成した 5 個の PRP 文に対して判定時間を測定した．結果を図 4 に示す．

$$\exists x \forall y \{ Ax + By < C \wedge Dx + Ey < F \wedge Gx + Hy < I \wedge Jx + Ky < L \}$$

実験の結果から，双方のアルゴリズムの判定時間は入力の PRP 文の係数と定数のビット長にほぼ比例しており，これは解析により求めた時間計算量から予想される結果と一致している．次に，入力の PRP 文の限定子交替数と，含まれる変数の数，不等式の数を変化させ，判定時間の変化を調べた．母式は不等式の単純な論理積であり，係数と定数のビット長は 7 ビットに統一した．

まず，最も外側の変数のみ ∀，その他の変数はすべて ∃ で束縛された場合（限定子交替数 =1）について，変数の数を 2 から 5 まで，不等式の数を 1 から 4 まで変化させ，判定時間を測定した．各係数と定数を 10 進数で分母 2 けた，分子 2 けたのランダムな値として，5 回測定した平均値を図 5 に示す．

次に，外側から 2 番目の変数のみが ∀，その他の変数はすべて ∃ で束縛された場合（限定子交替数 =2）について，変数の数を 3 から 5 まで，不等式の数を 1 から 4 まで変化させて判定時間を測定した．上の場合と同様に 5 回の測定値の平均を図 6 に示す(注 3)．なお，限定子の並び方，変数の数，不等式の数が同じであれば，係数と定数が変わっても，いずれのアルゴリズムでも判定時間は数倍程度しか変わらないようである(注 4)．

双方のアルゴリズムで変数の数と不等式の数が増えるに従い判定時間が指数的に増えているが，提案するアルゴリズムでは判定時間の増え方が Ferrante, Rackoff のアルゴリズムよりも少ないといえる．

また，双方のアルゴリズムの判定時間に関して次のようなことが観測された [12]．

まず，入力の PRP 文の各不等式に現れる変数の数が少ない場合は，Ferrante, Rackoff のアルゴリズムによって高速に判定できる．これは，Ferrante, Rackoff のアルゴリズムの性質からもわかることである．

次に，限定子交替数や同じ限定子が連続する最大の個数が同じ場合でも内側の限定子が交替する間隔が短い(注 5)といずれのアルゴリズムでも判定に時間がかかった．例えば，変数の数が 4，限定子交替数が 1，不

(注 3)：すべての組合せについて測定を行ったが，それぞれについて判定時間が 3000 秒を超えた時点で測定を打ち切った．また，その場合は結果は載せていない．
(注 4)：ただし，係数と定数がすべて同じであるなど，特殊な係数と定数の組合せに対してはこの限りではない．
(注 5)：例えば ∃∃∀∃∀ という限定子の並びは，∀∃∀∃∃ よりも内側の限定子が交替する間隔が短い．

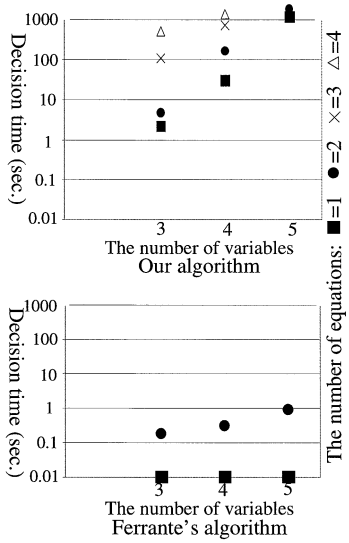

図 6　限定子交替数が 2 の場合の判定時間
Fig. 6　Decision times under the condition : the number of quantifier alternation is 2.

等式の数が 3 の場合，提案するアルゴリズムでの判定時間が約 1500 秒，Ferrante, Rackoff のアルゴリズムだと 3000 秒を超えた(注6)．これは，次の理由によるものであると思われる．外側の限定子が交替する間隔が短いと，次元の数が急激に減るので変数の数が多い式や次元の多いアレンジメントを処理する回数が少ない．変数の数が少ない式や，次元の少ないアレンジメントを処理する過程で，全く同じ不等式やフェース，フラットが数多く出現し，重複する不等式やフェース，フラットは 1 回だけ処理されるため，その分計算時間が減る．したがって，外側の限定子が交替する間隔が短いと計算時間が減る．

以上の実験から次のことがいえる．まず，アルゴリズムの実際の判定時間の変化は計算で求めた結果とだいたい一致する．また，外側の限定子が交替する間隔が短い場合には提案するアルゴリズムのほうが Ferrante, Rackoff のアルゴリズムよりも短い時間で判定できる．

5. むすび

本論文では時間計算量が $r\alpha^{\beta d}n^{\gamma d(b+1)^a}$（$\alpha, \beta, \gamma$ は定数，n は入力の PRP 文に含まれる不等式の個数，d は変数の個数，a は限定子交替数，b は同じ限定子が続く最大の個数，r は入力の式の各係数と定数の分母，分子のビット数）の計算幾何学の手法を利用した PRP 文真偽判定アルゴリズムを提案した．

今後の課題として，計算量オーダの定数部分を改良し，更に大きな問題に適用できるようにすることが挙げられる．アルゴリズムの改良の余地がある箇所としては，アレンジメントの処理のうち真偽判定にかかわってこない部分を省略するなどが挙げられる．

(注6)：係数と定数を変えて 5 回測定した平均値.

文　献

[1] 東野輝夫，北道淳司，谷口健一，“整数上の線形制約の処理と応用,” コンピュータソフトウェア，vol.9，no.6，pp.31–39，1992.
[2] A.R. Bruss and A. Meyer, “On time-space classes and their relation to the theory of real addition,” Thoret. Comput. Sci., vol.11, pp.59–69, 1980.
[3] L. Berman, “The complexity of logical theories,” Theoret. Comput. Sci., vol.11, pp.71–77, 1980.
[4] V. Weispfenning, “The complexity of linear problems in fields,” J. Symbolic Computation, vol.5, pp.3–27, 1988.
[5] C. Hosono and Y. Ikeda, “A formal derivation of the decidability of the theory SA,” Thoret. Comput. Sci., vol.127, pp.1–23, 1994.
[6] M.J. Fischer and M.O. Rabin, “Super exponential complexity of Presburger Arithmetic,” SIAM-AMS Proc. **VII**(AMS,Providence,RI), 1974.
[7] J. Ferrante and C. Rackoff, “A decision procedure for the first order theory of real addition with order,” SIAM J. Comput., vol.4, pp.69–76, 1975.
[8] J.E. Hopcroft and J.D. Ullmann, “Introduction to automata theory, languages and computation,” Addison-Wesley, 1979.
[9] H. Edelsbrunner, “Algorithms in Combinatorial Geometry,” Springer-Verlag, 1987.
[10] L.G. Khachiyan, “Polynomial algorithms for linear programming,” Dokl. Akad. Nauk SSSR vol.244, pp.1093–1096, 1979.
[11] 柴田直樹，岡野浩三，東野輝夫，谷口健一，“Tarski 算術における冠頭標準形の閉論理式の真偽判定アルゴリズムの提案,” 信学技報，COMP97-47, 1997.
[12] 柴田直樹，“冠頭標準形有理数プレスブルガー文の真偽判定アルゴリズム,” 大阪大学基礎工学研究科修士学位論文，1998.
[13] A. Nakata, T. Higashino, and K. Taniguchi, “Time-action alternating model for timed LOTOS and its symbolic verification of bisimulation equivalence,” Proc. of FORTE/PSTV’96, pp.279–294, 1996.

（平成 10 年 5 月 6 日受付，12 月 4 日再受付）

柴田　直樹

平 8 阪大・基礎工・情報中退．現在同大大学院博士後期課程在学中．ハードウェア，ソフトウェアの形式的検証等に興味をもつ．情報処理学会会員．

岡野　浩三　（正員）

平 2 阪大・基礎工・情報卒．平 5 同大大学院博士後期課程中退．同年同大情報工学科助手，現在に至る．工博．代数的手法によるプログラム開発，分散システムなどの研究に従事．情報処理学会会員．

東野　輝夫　（正員）

昭 54 阪大・基礎工・情報卒．昭 59 同大大学院博士課程了．同年同大学助手．平 2，6 モントリオール大学客員研究員．現在，同大大学院基礎工学研究科助教授，工博．分散システム，通信プロトコル等の研究に従事．情報処理学会，ACM 各会員．IEEE Senior Member.

谷口　健一　（正員）

昭 40 阪大・工・電子卒．昭 45 同大大学院博士課程了．同年同大助手．現在，同大大学院基礎工学研究科教授．工博．この間，計算理論，ソフトウェアやハードウェアの仕様記述・実現・検証の代数的手法及び支援システム，関数型言語の処理系，分散システムや通信プロトコルの設計・検証法等に関する研究に従事．