アルゴリズム　１７－１２
(1990. 9.29)

部分マップ法を用いたファジー論理関数のファジー主項生成アルゴリズム

山口　健治
工学院大学

あらまし　ファジー論理関数の最小化の問題は通常２値論理関数のそれよりも複雑である。しかしその複雑な問題は、より小さな２値論理関数の最小化の問題に置き換えられる。本稿で述べるアルゴリズムにおいては、最初の大規模な問題をいくつかのより小さな問題に分解し、部分マップ法を用いて２値論理関数の最小化を行って、ファジー主項を得ている。部分マップ法では、カルノー図上に部分マップを作成し、更にその部分マップ上で許容キューブを生成して論理関数の主項を導いている。コンピュータプログラムの実行時間に関して本稿の方法とファジーコンセンサス法とを比較すると、本稿の方法は実行時間が 1/3 ～ 2/3 に減少している。

Fuzzy Prime Implicants Generation Algorithm of Fuzzy Function using Submap Method

Kenji YAMAGUCHI
Kogakuin University
1-24-2 Nishi-Shinjuku, Shinjuku-ku, Tokyo 160 JAPAN

Abstract　Usually, the problem of minimizing fuzzy switching function is complex when it is compared with the problem of minimizing binary switching function. But, that complex problem is replaced the smaller problems of minimizing binary switching function. In the algorithm described in this paper, the original large scaled problem is decomposed into the several smaller problems, and it generates fuzzy prime implicants by to execute the minimization procedure of binary switching function using Submap Method. The execution time of computer program of the algorithm described in this paper has reduced by 1/3 ～ 2/3 when it is compared with the fuzzy consensus method.

1　はじめに

L. A. Zadeh がファジー集合[1]の概念を発表して以来、ファジー論理関数の最小化の研究は、2値論理関数の最小化の研究の分野と密接な関係を持って発展してきた[2]〜[8]。ファジー論理関数の最小化の問題は通常2値論理関数のそれよりも複雑な問題となっている。ファジー論理関数の最小化の方法にはマップ法[2]、テーブル法[3][4]、コンセンサス法[5][6][7] 等あるが、しかしこれらの方法は変数の増大によってやがて処置し難くなるであろう。これに対して、ファジー論理関数の最小化の問題を2値論理関数の最小化の問題に帰着させ、その複雑、大規模な問題をより小さないくつかの問題に分解することによって容易にするという方法が考案された[8]。 この方法では、加法形式のファジー論理関数の最小化は相補項のみについて考えれば良いということと、相補項は同じ相補的文字の組($x_i, \bar{x}_i$) を持つ相補最小項あるいは相補項にのみサブサムされるという2つの性質を利用している。そしてファジー論理関数の最小化は、相補的文字の同じ組を持つ項より、その相補的文字の組を取り除いた項の和を2値論理関数として、この2値論理関数を最小化すれば良いという結論を導いている。

一方、2値論理関数の最小化の分野では、部分マップ法[9] が高速な方法として知られている。部分マップ法はカルノー図等のマップの部分マップの概念を用いている。この方法では、部分マップごとに生成した許容キューブが関数の主項になっているかどうかを調べることにより全ての主項を導出することができる。また無駄な許容キューブの生成を抑制するためセル番号の大きい順に部分マップを作成し、セル番号の小さい順に許容キューブを生成している。

本稿は、ファジー論理関数の最小化の問題をより小さな2値論理関数の最小化の問題に分解するという方法に、部分マップ法を組み合わせた新しいファジー論理関数の最小化アルゴリズムについて述べたものである。本稿のファジー論理関数の最小化アルゴリズムでは、最初の大規模な問題がより小さな問題に置き換えられ、更に2値論理関数に対しては、分割された部分マップごとに主項を導出して行く。従って、高速にファジー主項を求めることができる。コンピュータプログラムの実験では、本稿の方法は、ファジー主項を求めるために多数の繰り返し操作を必要とするコンセンサスを用いる最小化アルゴリズム[6]と比較すると計算時間が 1/3 ないし 2/3に減少しているという結果が得られた。

2　基本的事項

ファジー論理関数に関するいくつかの定義を挙げておく。

〔定義1〕　ファジー命題は、その真理値として実数閉区間〔0, 1〕の任意の値をとる。ファジー命題x をファジー変数（以下変数とする）という。合成命題は、max, minを区間〔0, 1〕の通常の大小関係を表すものとすれば、次の論理演算で求められる。

論理和　$x_1 + x_2 = \max(x_1, x_2)$

論理積　$x_1 \cdot x_2 = \min(x_1, x_2)$（記号・は省略することもある）

否定　　$\bar{x} = 1 - x$

上記の論理演算が定義されている代数系をファジー論理という。変数 x またはその否定 $\bar{x}$ をファジー文字（以下文字とする）といい、x と $\bar{x}$ とは互いに補元であるという。

〔定義2〕　0, 1, x, $\bar{x}$の式と、文字を論理和、論理積で結合して得られる式とをファジー論理式という。ファジー論理式が表現している、真理値集合〔0, 1〕への写像を行う関数をファジー論理関数という。

〔定義3〕　いくつかの異なる文字の積を積項という。積項に、相補的文字の組（ある変数x_iとその否定$\bar{x}_i$の組）が含まれているならば相補積項（相補項）といい、含まれていないならば単積項（単項）という。相補積項は全ての変数を含むとき相補最小項という。また、$x_i + \bar{x}_i \geqq x_j \cdot \bar{x}_j$となることより相補積項は相補最小項の和に展開することができる。

〔定義4〕　α, βを積項とするとき、αの全ての文字がβに存在するならば、βはαをサブサムする（$\beta \leqq \alpha$）といい、$\alpha + \beta = \alpha$が成立してβを省くことができる。

〔定義5〕　ファジー論理関数Fについて、積項αが全ての〔0, 1〕の値に対して$\alpha \leqq$ F なら

ば、積項αをファジーインプリカントという。またファジーインプリカントαは、Fの他のインプリカントをサブサムしなければファジー主項（ファジープライムインプリカント）という。

〔定義6〕　いくつかの積項の和で表されているファジー論理関数をファジー加法標準形という。ファジー加法標準形は、いずれの他の項をもサブサムしないような単項あるいは相補最小項によって構成されるときファジー主加法標準形という。

ファジー論理関数は次のことが言えれば最簡形式と考えられる。

（1）　項数のより少ない他の等価な式がない。

（2）　項数が同じで文字数のより少ない他の等価な式がない。

文字数最小の条件を満たす項はファジー主項で、従って加法形式のファジー論理関数の最簡形式はファジー主項の和で表される。一般的に最簡形式は次のように2段階に分けて求められる。

（1）　関数Fの全てのファジー主項を求める。

（2）　関数Fと等価なファジー主項の最小の組み合わせで表される式を見出す。

上記の手順（2）は（1）で求めたファジー主項の集合の部分集合を求めるという最小カバー問題となるので、結局ファジー論理関数の最小化問題では（1）がその本質的な部分となっている。本稿では上記手順の（1）の部分のみについて扱うこととし、これをファジー論理関数の最小化と呼んでいる。また2値論理関数の最小化もこの意味に用いている。

3　2値論理関数の最小化問題への置換

ファジー論理関数の最小化の問題は2値論理関数の最小化の問題に置き換えることができる。任意の加法形式のファジー論理関数は単項と相補項とによって構成される。そしてこれらの単項と相補項とはそれぞれ以下のような性質を持つ。

最初に単項の性質について述べる。加法形式のファジー論理関数Fに存在する全ての単項は他の単項をサブサムしなければファジー主項である。また単項が関数Fのファジー主項であれば、最簡形式に必ず含まれるような必須項でもある。これらは次のような理由からである。相補項はある文字とその補元を同時に含んでいるので、ファジー論理関数Fに存在する単項は相補項をサブサムするということはあり得ない。またある単項が他の全ての単項をサブサムしなければ、結局関数Fより消去することができずファジー主項となり同時に必須項となる。

このような単項の性質によりファジー論理関数の最小化の手順は次のようになる。

（1）　関数Fに存在する単項同志のサブサムの関係を調べて、サブサムしている単項を全て消去する。

（2）　関数Fに存在する相補積項のみについてファジー主項を求める。

こうしてファジー論理関数の最小化は、単項についてはサブサムの関係だけを調べれば良いので、相補項についてのみ考えれば良いということが言える。

次に相補項の性質について述べる。相補的文字のある組を持つ相補項は、その同じ相補的文字の組を持つ相補最小項（あるいは相補項）によってのみサブサムされる。これは次のような理由からである。上記のことは言い換えると、β, β' を加法形式のファジー論理関数Fに存在する相補項とし、$\beta \leqq \beta'$ とすれば相補的文字の組が互いに他に等しいということと等価である。今x_i, $\bar{x}_i$をβ'の相補的文字の組とする。β'の全ての文字がβに存在するならば$\beta \leqq \beta'$であって、x_i, $\bar{x}_i$はβのなかに存在する。ここでx_i, $\bar{x}_i$の組がβに存在し、x_iかあるいは$\bar{x}_i$の文字がβ' に存在しないと考える。そうすると$\beta \leqq \beta'$と考えているので$\beta \leqq \beta' \leqq F$で、$F = F' + \beta = F' + \beta'$ となる。このことは相補項βの相補的文字x_iか$\bar{x}_i$は取り除くことができるということを意味するが、しかしこれは次のように矛盾する。すなわち、加法形式のファジー論理関数FにおいてF′を相補項β_1が含まれない項の集合とすると、$F = F' + \beta_1 = F' + x_i \bar{x}_i \beta_1'$。ここで$\beta_1$ の文字x_iが消去され得るとすると$F = F' + \bar{x}_i \beta_1'$ となり、$\bar{x}_i \beta_1' \leqq F = F' + x_i \bar{x}_i \beta_1'$となる。そうすると$\bar{x}_i \beta_1'$ が単項であっても相補項であっても $\bar{x}_i \beta_1' \leqq\!\!\!\!/\; x_i \bar{x}_i \beta_1'$なので $\bar{x}_i \beta_1' \leqq F'$でなければならない。従って $x_i \bar{x}_i \beta_1' \leqq \bar{x}_i \beta_1' \leqq F'$となり$x_i \bar{x}_i \beta_1'$（$= \beta_1$）はF′をサブサムすることになり$\beta_1$は消去されてしまう。これは矛盾である。$\bar{x}_i$を

消去し得るとした場合も同様である。それ故相補項βの相補的文字を取り除くことはできず、x_iかあるいは$\bar{x}_i$の文字がβ'に存在しないと考えることはできない。従って$\beta \leqq \beta'$であって、βにx_i, $\bar{x}_i$の組が存在すれば、x_i, $\bar{x}_i$はβ'にも存在する。

相補項のこのような性質により、$x^c = x_{i1}\bar{x}_{i1} x_{i2}\bar{x}_{i2} \cdots\cdots x_{ik}\bar{x}_{ik}$のような相補的文字の組の積を持つ相補最小項$\beta_i$の最小カバーを求めようとするとき、$x^c$と相補的文字の組の同じ積を持つ相補最小項についてのみ考慮すれば良い。そしてこれは相補的文字の異なった組を持つ他の相補最小項に対しては独立に行われる。従ってファジー論理関数の最小化はいくつかのより小さな問題に分解でき、相補的文字の共通積を持つ相補最小項の集合に分けて最小化される。今x^cが相補的文字のある組の積であるとすると次のようになる。

$$x^c\bar{x}_i\alpha + x^c x_i \alpha = x^c(\bar{x}_i + x_i)\alpha = x^c\alpha$$

この最小化の機構はちょうど2値論理関数のそれと同じである。$\beta_{i1}, \cdots\cdots, \beta_{im}$を相補的文字の共通積を持つ相補最小項とすると、

$$\beta_{i1} + \cdots\cdots + \beta_{im} = x^c(\beta_{i1}' + \cdots\cdots + \beta_{im}') = x^c f_1$$

と表される。ここで$\beta_{i1}', \cdots\cdots, \beta_{im}'$は$x^c$に現れる文字を除いて構成された単項である。これより相補最小項$\beta_{i1} + \cdots\cdots + \beta_{im}$の和の最小化は相補的文字の共通積を取り除いた項の和を2値論理関数($f_1 = \beta_{i1}' + \cdots\cdots + \beta_{im}'$)としてこの2値論理関数を最小化することに等しい。

先に述べた単項の性質により、ファジー論理関数の最小化は相補項についてのみ考えれば良かったが、2値論理関数に置き換えられた相補最小項の最小化において、単項はドントケアの入力として用いることができる。今、主加法標準形のファジー論理関数をFとし、Fに存在する相補最小項より相補的文字の共通積x^cを持つものを選び出し、これよりx^cを取り除いた項の和を作り2値論理関数f_1とする。Fに存在する単項よりx^cに含まれる文字を全て取り除いて得られる項の和をf_dとする。そうすると、

$$x^c f_1 \leqq F$$
$$x^c f_d \leqq F$$
$$\therefore \quad x^c(f_1 + f_d) \leqq F$$

となり、f_1の最小化においてf_dをドントケアの入力として用いることができる。

結局より小さな問題に分解されたファジー論理関数の最小化において、その各々は、相補最小項より相補的文字の共通積を取り除いた項の和を最小化することであり、2値論理関数f_dによって表されるドントケアの入力を持つ2値論理関数$f_{1d}(= f_1 + f_d)$を最小化することによって実現される。

【例1】　上記の方法によって次に示すようなファジー論理関数Fのファジー主項を求めてみよう。

$$F = x_1x_2\bar{x}_2x_3 + \bar{x}_1x_2\bar{x}_2\bar{x}_3 + x_1\bar{x}_1x_2x_3 + \bar{x}_1\bar{x}_2x_3$$

上式で相補的文字の組は$x_1\bar{x}_1$と$x_2\bar{x}_2$の2組で、単項は$\bar{x}_1\bar{x}_2x_3$である。

（1）　相補的文字の共通積$x_2\bar{x}_2$を選ぶ。

$$x^c = x_2\bar{x}_2$$
$$f_1 = x_1x_3 + \bar{x}_1\bar{x}_3$$
$$f_d = \bar{x}_1x_3$$
$$\therefore \quad f_{1d} = f_1 + f_d = x_1x_3 + \bar{x}_1\bar{x}_3 + \bar{x}_1x_3 = \bar{x}_1 + x_3$$

f_{1d}の何れかの項がf_1の任意の項にサブサムされないならば、f_1にその項を付加することによってf_1のどの項も消去されないので、そのような項は捨てる。ここではf_{1d}の項$\bar{x}_1$, x_3が共にf_1の項にサブサムされているので2値論理関数f_{1d}の主項として登録する。ファジー主項はx^cを付加して、$\bar{x}_1x_2\bar{x}_2$, $x_2\bar{x}_2x_3$となる。

（2）　相補的文字の共通積$x_1\bar{x}_1$を選ぶ。

$$x^c = x_1\bar{x}_1$$
$$f_1 = x_2x_3$$
$$f_d = \bar{x}_2x_3$$
$$\therefore \quad f_{1d} = x_2x_3 + \bar{x}_2x_3 = x_3$$

ファジー主項は$x_1\bar{x}_1x_3$となる。

以上よりファジー論理関数Fは

$$F = \bar{x}_1x_2\bar{x}_2 + x_2\bar{x}_2x_3 + x_1\bar{x}_1x_3 + \bar{x}_1\bar{x}_2x_3$$

となる。

4　部分マップ法について

前章でファジー論理関数の最小化の問題は2値論理関数の最小化の問題に置き換えられるということを述べた。この章では部分マップ法を用いて2値論理関数の最小化を行っていくこと

について述べる。以下この章では特に断りのない限り論理関数とは２値論理関数を表すものとする。

部分マップ法はカルノー図等のマップの部分マップの概念を用いている。部分マップ法ではカルノー図上に部分マップを作成し、更に部分マップ上に許容キューブを生成して論理関数の主項を導いている。今４変数の場合について考える。カルノー図上の各セルは２進座標で表される（例えばセル5 は(0101)となる）が、セル j の座標成分に対して、否定変数0 を共通に持つセルの集合(セル j も含む）を部分マップ S_j という。S_{15}の場合、セル15は否定変数を持たないが、特別なものとしてマップ全体とする。また S_0は単独のセル0からなる最小の部分マップである。従って部分マップ S_j に対応するセル番号 j が大きい程部分マップは大きくなる。部分マップ S_j はセル j の座標の否定変数を取り除いた残りの変数についてのマップになっている。図１に部分マップ S_{13}の例を示す（２重線枠）。

$x_3x_4 \backslash x_1x_2$	00	01	11	10
00	0	4	12	8
01	1	5	13	9
11	3	7	15	11
10	2	6	14	10

S_{13}

$P_{4,13}$

図１　部分マップと許容キューブの例

部分マップ S_j 上でセル i の座標成分に対して、肯定変数１を共通に持つセルの集合（セル i も含む)を部分マップ S_jの許容キューブ $P_{i,j}$ という。$P_{0,j}$ の場合、セル0 は肯定変数を持たないが、特別なものとして部分マップ S_j 全体とする。また $P_{j,j}$は部分マップ S_j上で単独のセル j よりなる最小の許容キューブである。従って許容キューブ $P_{i,j}$ に対応するセル番号 i が小さい程許容キューブは大きくなる。部分マップ S_j の許容キューブ $P_{i,j}$ はセル i の座標の肯定変数を取り除いた残りの変数についてのマップになっている。図１に許容キューブ $P_{4,13}$ の例を示す（実線枠）。許容キューブ $P_{i,j}$ はマップ上の特定のセルからなる部分であって、$P_{i,j}$に含まれる全てのセルで関数 f が１の値をとるとき、$P_{i,j}$ は f のインプリカントであるという。

論理関数 f の主項は部分マップ法を用いて次のようにして求める。カルノー図上で与えられた関数に対して、部分マップ S_j をTrueのセルのみについてセル番号の大きい順に作成し、各部分マップごとに許容キューブ $P_{i,j}$ をTrueのセルのみについてセル番号 i の小さい順に生成する。こうして生成された許容キューブ $P_{i,j}$ についてインプリカントのみを採用し、既に求められている項をサブサムしなければ主項として登録する。ここで、ある部分マップにおいてある許容キューブが f のインプリカントならば、それに含まれるセル番号の許容キューブは、既に生成したインプリカントである許容キューブに含まれてしまうので、生成する必要はない。またある部分マップにおいて、生成した全ての許容キューブが f のインプリカントならば、その部分マップに含まれるセル番号の部分マップは以後考えなくて良い。その理由は、新たに部分マップを作成し、許容キューブを生成しても前に生成した許容キューブに含まれてしまうからである。なお、セル番号の大きい順に部分マップを作成し、セル番号の小さい順に許容キューブを生成しているのは、インプリカントになる許容キューブに含まれるセルの数の大きい（言い換えるとインプリカントに含まれる文字数が小さい）ものから先に生成しようとしているからである。

【例２】　図２のマップで与えられる関数 f について部分マップ法で主項を導出する。但し丸印のセルでTrue、無印のセルでFalseを表す。

$x_3x_4 \backslash x_1x_2$	00	01	11	10
00	0	4	⑫	8
01	①	⑤	⑬	9
11	③	⑦	15	11
10	2	6	⑭	10

○；True

図２　関数 f

True の最も大きなセル番号14 から部分マップ S_{14}を作成すると図３のようになる。部分マッ

プS_{14} のなかでTrue の最小のセルの番号は12なので許容キューブ $P_{12,14}$={⑫, ⑭}を生成する。この許容キューブは、全てのセルが関数 f に含まれているので f のインプリカントになり、また最初のインプリカントであるから主項として登録する。$P_{14,14}$ はセル14 が$P_{12,14}$に含まれているので生成不要で、またセル12についての部分マップは作成不要となる。次に部分マップS_{13}を作成すると図4のようになる。

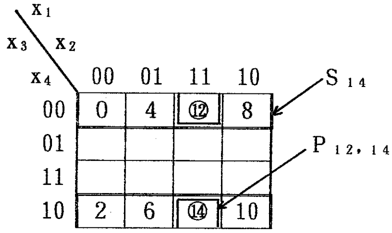

図3　部分マップS_{14}

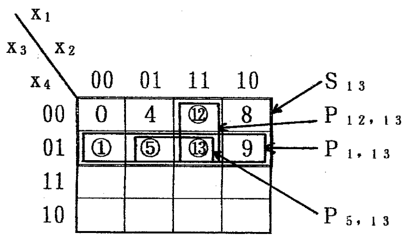

図4　部分マップS_{13}

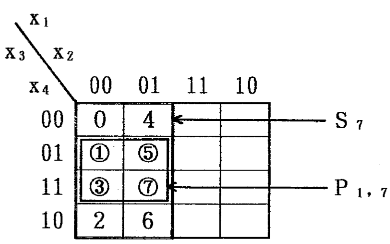

図5　部分マップS_7

図4の部分マップS_{13}では最小のセルの番号は1なのでセル1より順に許容キューブを生成していく。$P_{1,13}$={①, ⑤, 9, ⑬} ではセル9が関数 f に含まれていないのでこの許容キューブは捨てる。$P_{5,13}$={⑤, ⑬}, $P_{12,13}$= {⑫, ⑬} はインプリカントであり同時に既に生成した主項をサブサムしないので主項として登録。$P_{13,13}$は生成不要。部分マップS_7については図5のようになる。許容キューブ$P_{1,7}$= {①, ③, ⑤, ⑦}は主項となる。よって$P_{3,7}$, $P_{5,7}$, $P_{7,7}$は生成不要。またセル1, 3, 5についての部分マップは作成不要となる。これで全ての主項が得られた。主項は$P_{12,14}=x_1x_2\bar{x}_4$, $P_{5,13}=x_2\bar{x}_3x_4$, $P_{12,13}=x_1x_2\bar{x}_3$, $P_{1,7}=\bar{x}_1x_4$の4個となる。

以上は次のように考えても良い。最初に生成された許容キューブ$P_{12,14}$について、部分マップ作成時のセル14の2進座標は(1110)で共通の否定変数は$\bar{x}_4$である。また許容キューブ生成時のセル12の2進座標は(1100)で共通の肯定変数はx_1x_2である。従って共通の否定変数$\bar{x}_4$と肯定変数x_1x_2との積$x_1x_2\bar{x}_4$が$P_{12,14}$ を与える。残りの3個の主項についても同様に考えることができる。

5　アルゴリズム

加法形式のファジー論理関数の最小化アルゴリズムにおいて、ファジー変数は次のような等価10進数で表すことにする。すなわち、x_iに2, $\bar{x}_i$に1, $x_i \cdot \bar{x}_i$に3, 積項にx_i かあるいは$\bar{x}_i$ が現れないときは0を対応させる。今、加法形式のファジー論理関数Fを与える。計算機入力のため関数Fの各々の項に対して1つの行が対応するようなテーブルを構成する。そうすると行a_i は関数Fのi番目の項の変数を含むことになる。このテーブルはまた関数Fの各々の変数に対して1つの列が対応し、列b_j は関数Fのj番目の変数を表している。

【アルゴリズム】

〔1〕　関数Fを単項の和(Fe)と相補項の和(F_c)に分離する。Feでサブサムする項を消去し、F_cの相補項を相補最小項の和に展開する。

〔2〕　F_cにおいて相補的文字の共通積x^c を持つ項を選択しF_{c1}とする。F_{c1}を構成する項よりx^cを消去し、これをf_1とする。更にFeよりx^cに含まれる文字を全て消去してf_dを得る。

〔3〕　2値論理関数f_{1d}(=f_1+f_d)を構成し、各項を最小項に展開する。また各最小項の2進座標に対応する10進表示を求める。

〔4〕　10進数の大きい順に部分マップを作成し、各々の部分マップごとに〔5〕を繰り返す。

〔5〕　〔4〕で作成した部分マップで10進数

の小さい順に許容キューブを作成し、この許容キューブが f_{1d} のインプリカントでかつ f_1 にサブサムされ、更に、既に登録されている主項をサブサムしなければ主項として登録する。

〔6〕 全ての部分マップが作成されていなければ〔4〕より繰り返す。

〔7〕 〔5〕で得られた各項に x^c を付加してファジー主項を得る。$F_c = F_c - F_{c1}$として〔2〕より繰り返す。F_cが空のとき終了する。

【例3】 次のようなファジー論理関数Fについて上記のアルゴリズムでファジー主項を求めてみよう。

$$F = x_1\bar{x}_1x_2\bar{x}_3x_5 + \bar{x}_2x_3x_4\bar{x}_5 + x_1\bar{x}_1\bar{x}_3x_4\bar{x}_4x_5 + x_2\bar{x}_2x_3x_4\bar{x}_4 + \bar{x}_1x_2x_3\bar{x}_3x_4\bar{x}_5$$

（1） 単項は第2項目ただ1つなのでこれがファジー主項となる。

（2） 相補的文字の共通積として$x_1\bar{x}_1$を選ぶ。

$x^c = x_1\bar{x}_1$

$f_1 = x_2\bar{x}_3x_5$

$f_d = \bar{x}_2x_3x_4\bar{x}_5$

$f_{1d} = x_2\bar{x}_3x_4x_5 + x_2\bar{x}_3\bar{x}_4x_5 + \bar{x}_2x_3x_4\bar{x}_5$

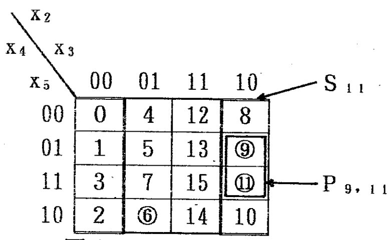

図6 S_{11}と$P_{9,11}$

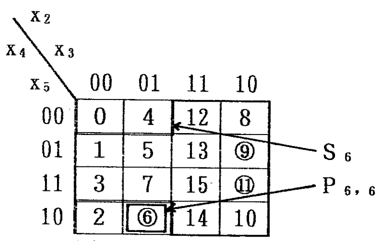

図7 S_6と$P_{6,6}$

f_{1d}のマップより、部分マップS_{11}を作成し、許容キューブ $P_{9,11}$ を生成する（図6）。許容キューブ$P_{9,11}$ = {⑨, ⑪} $= x_2\bar{x}_3x_5$は f_{1d}のインプリカントで、かつ f_1 の項にサブサムされているので主項として登録する。次に部分マップS_6を作成し、許容キューブ$P_{6,6}$ を生成する（図7）。$P_{6,6}$ = {⑥} $= \bar{x}_2x_3x_4\bar{x}_5$ は f_1にサブサムされないので捨てる。従ってファジー主項は$x_1\bar{x}_1x_2\bar{x}_3x_5$となる。

（3） 相補的文字の共通積$x_3\bar{x}_3$を選ぶ。

$x^c = x_3\bar{x}_3$

$f_1 = \bar{x}_1x_2x_4\bar{x}_5$

$f_d = \bar{x}_2x_4\bar{x}_5$

$f_{1d} = \bar{x}_1x_2x_4\bar{x}_5 + x_1\bar{x}_2x_4\bar{x}_5 + \bar{x}_1\bar{x}_2x_4\bar{x}_5$

f_{1d}のマップより部分マップS_{10}, S_6を作成し、許容キューブ$P_{2,10}$ = {②, ⑩} $= \bar{x}_2x_4\bar{x}_5$, $P_{2,6}$ = {②, ⑥} $= \bar{x}_1x_4\bar{x}_5$ を生成する。$P_{2,6}$は主項であり、$P_{2,10}$はf_1にサブサムされないので捨てる。従ってファジー主項は$\bar{x}_1x_3\bar{x}_3x_4\bar{x}_5$となる。

（4） 相補的文字の共通積$x_1\bar{x}_1x_4\bar{x}_4$を選ぶ。

$x^c = x_1\bar{x}_1x_4\bar{x}_4$

$f_1 = \bar{x}_3x_5$

$f_d = \bar{x}_2x_3\bar{x}_5$

$f_{1d} = \bar{x}_2\bar{x}_3x_5 + x_2\bar{x}_3x_5 + \bar{x}_2x_3\bar{x}_5$

f_{1d}のマップより部分マップS_5, S_2を作成し、許容キューブ $P_{1,5}$ = {①, ⑤} $= \bar{x}_3x_5$, $P_{2,2}$ = {②} $= \bar{x}_2x_3\bar{x}_5$を生成する。$P_{1,5}$は主項であり、$P_{2,2}$ はf_1にサブサムされないので捨てる。従ってファジー主項は$x_1\bar{x}_1\bar{x}_3x_4\bar{x}_4x_5$となる。

（5） 相補的文字の共通積$x_2\bar{x}_2x_4\bar{x}_4$を選ぶ。

$x^c = x_2\bar{x}_2x_4\bar{x}_4$

$f_1 = x_3$

$f_d = x_3\bar{x}_5$

$f_{1d} = x_1x_3x_5 + \bar{x}_1x_3x_5 + x_1x_3\bar{x}_5 + \bar{x}_1x_3\bar{x}_5$

f_{1d}のマップより部分マップS_7 を作成し、許容キューブ$P_{2,7}$ = {②, ③, ⑥, ⑦} $= x_3$を生成する。これはf_1にサブサムされるので主項である。従ってファジー主項は$x_2\bar{x}_2x_3x_4\bar{x}_4$となる。

以上より関数Fは

$$F = \bar{x}_2x_3x_4\bar{x}_5 + x_1\bar{x}_1x_2\bar{x}_3x_5 + \bar{x}_1x_3\bar{x}_3x_4\bar{x}_5 + x_1\bar{x}_1\bar{x}_3x_4\bar{x}_4x_5 + x_2\bar{x}_2x_3x_4\bar{x}_4$$

となる。

6 計算結果

以上のようなアルゴリズムに基づいてプログラムを作成し評価を行った。プログラムはFORTRANで記述し、ワークステーション ＤＮ－3000（アポロ社）で実行した。表１にファジーコンセンサスを用いる方法[6] との計算時間の比較を示す。計算は１０変数の関数について行った。本稿の方法はファジーコンセンサス法に対して計算時間が 1/3 ないし 2/3 に減少している。

表１　計算時間の比較

	項 数	本稿の方法(秒)	ファジーコンセンサス法(秒)	コンセンサス法に対する比
1	10	5.3	8.6	0.616
2	11	6.2	9.3	0.667
3	12	7.0	18.8	0.372
4	13	7.4	15.7	0.471
5	14	7.7	18.9	0.407

7　むすび

本稿では、ファジー論理関数の最小化の問題を２値論理関数の最小化の問題に帰着させ、最初の大規模な問題をより小さないくつかの問題に分解し、部分マップ法を用いて２値論理関数の最小化を行って、ファジー主項を生成するという方法を示した。本稿の方法はファジーコンセンサスを用いる方法に比べて計算時間が短縮されているということを示した。また本稿のアルゴリズムでは全てのファジー主項の集合を生成するが、最簡形式に必要でない項も多少含まれる場合がある。このような項の生成をより少なくすることが今後の課題である。

参考文献

[1] L.A.Zadeh : “Fuzzy sets”, Inf.& Control, Vol.8, pp.338-353 (1965)

[2] G. W. Schwede and A. Kandel :“Fuzzy maps”, IEEE Trans. Syst. Man. & Cybern., Vol.SMC-7, No.9, pp.669-674 (Sept.1977)

[3] S. M. Rickman and A. Kandel : “Column table approach for the minimization of fuzzy functions”, Inf. Sciences, Vol.12, pp.111-128(1977)

[4] S.M.Rickman and A.Kandel:“Tabular minimization of fuzzy switching functions”, IEEE Trans. Syst. Man.& Cybern., Vol.SMC-6, No.11, pp.766-769 (Nov. 1976)

[5] A.Kandel:“On minimization of fuzzy functions”, IEEE Trans.Comput., Vol.C-22, No.9, pp.826-832 (Sept.1973)

[6] A.Kandel and J.M.Francioni: “On the properties and applications of fuzzy-valued switching functions”, IEEE Trans.Comput., Vol.C-29, No.11, pp.986-994 (Nov.1980)

[7] 向殿政男 :“Fuzzy論理関数の代数的構造とその最簡形式および既約形式”, 信学論(D), Vol.58-D, No.12, pp.748-755 (昭50-12)

[8] M.Mukaidono:“An improved method for minimizing fuzzy switching functions”, Proc. 14th Int. Sym. on Multiple-Valued Logic, pp.196-201 (May 1984)

[9] 松田秀雄, 宮腰隆: “論理関数の主項を高速に導出する方法--部分マップ法について--”, 信学論(D), Vol. J67-D, No.2, pp.208-215 (昭59-02)