節展開法のクリーネ論理関数族への拡張と最簡形について

4H－6　荒木智行[+]　巽 久行[+]　向殿政男[++]　大前義次[+]
+神奈川工科大学 情報工学科　++明治大学 理工学部 情報科学科

あらまし 本報告では、Nelsonの定理を拡張した定理が成立するクリーネ論理関数族、特に多値クリーネ論理関数、定数係数を持ったファジィ論理関数について節展開法が拡張できることを示し、主項展開、最簡形を求めるための効率の良いアルゴリズムを示す。

1. はじめに

２値論理関数の主項を求めるための手法としてNelsonの定理[3]がよく知られている。Nelsonの定理は、乗法形式で与えられた論理関数を加法形式に展開することを必要とする。節展開法[4]は、これを効率良く行うための手法である。クリーネ関数族の中には、Nelsonの定理を拡張した定理が成立する関数があることが知られている[6][7]。本報告では、これらの関数の主項展開を求める方法として節展開法が拡張できることを示す。節展開法は山口によりファジィ節展開法[5]と名付けられ、クリーネ論理関数族の一つであるファジィ論理関数に対して拡張されているが、主項展開を求めるものではなく、最簡形を求める中間の段階としての拡張がなされている。本報告では、節展開法を主項展開を求める手法として位置付け、ファジィ論理関数をその特殊な場合として含む多値クリーネ論理関数、定数係数を持ったファジィ論理関数に対して拡張を行う。また、一般に論理関数は加法形式で取り扱われることが多く、入力も出力も加法形式で表現する場合が少なくない。節展開法は入力として乗法形式を仮定しており、もし入力を加法形式で与えたい場合、一度乗法形式に変換する必要があり、効率的でない。本報告では最簡形式を求める際に、論理関数をある特殊な標準形に変換することにより、その効率を上げげられる可能性のある手法を提案する。また、この手法により最簡形を求める中間の段階で生成される不必要項の数を減少させることができる可能性がある。

2. 諸準備

ここでは、基本的な定義、定理について述べる。これらは既に文献[1][2][6][7]に述べられているが必要最少限を示すことにする。

[定義1] n変数の多値クリーネ論理関数、定数係数を持ったファジィ論理関数とは、次のように定義された論理式で表現された写像である。

(D1.1) 定数c_1, c_2, …, c_tは論理式である。

(D1.2) 変数x_1, x_2, …, x_nは論理式である。

(D1.2) F, Gが論理式であれば、F・G, F∨G, ～Fは論理式である。

但し、論理演算は次のように定義される。

$x \cdot y = \min(x, y)$,　$x \vee y = \max(x, y)$,　$\sim x = 1 - x$

また、変数x_iのとる真理値および定数c_iの値は、m値クリーネ論理関数の場合は0, 1/(m-1), 2/(m-1), …, (m-2)/(m-1), 1であり、定数係数を持ったファジィ論理関数の場合、実数閉区間[0, 1]の任意の値である。□

以降、多値クリーネ論理関数のことをMVKF、定数係数を持ったファジィ論理関数のことをFF/Cと略記し、論理式とそれの表す関数を同一視する。

[定義2] 変数x_iとその否定$\sim x_i$、および定数のことを<u>文字</u>という。定数を含めないときは<u>変数の文字</u>という。□

[定義3] MVKF, FF/Cには次のように３つのタイプの積項と和項がある。積項については、

(Type1) 定数(＞1/2)を含み、すべてのｉについてx_i, $\sim x_i$が同時現れないもの。

(Type2) 定数(≦1/2)を含み、すべてのｉについてx_i, $\sim x_i$が同時現れないもの。

(Type3) あるｉについてx_i, $\sim x_i$を同時に含むもの。

Type1の積項を<u>単積項</u>といい、Type2, Type3の積項を<u>相補項</u>という。和項については、

(Type1) 定数(＜1/2)を含み、すべてのｉについてx_i, $\sim x_i$が同時現れないもの。

(Type2) 定数(≧1/2)を含み、すべてのｉについてx_i, $\sim x_i$が同時現れないもの。

(Type3) あるｉについてx_i, $\sim x_i$を同時に含むもの。

Type1の和項のことを<u>単和項</u>といい、Type2, Type3の和項を<u>相補和項</u>という。□

[定義4] 相補項(相補和項)ですべての変数を含むものを<u>相補最小項</u>(<u>相補最大項</u>)という。□

[定義5] MVKFおよびFF/Cが単積項と相補最小項の和で表現されているとき<u>主加法標準形</u>という。また、単和項と相補最大項の積のときは<u>主乗法標準形</u>という。□

[定理1] 任意のMVKFおよびFF/Cの主加法標準形(主乗法標準形)は一意に表現できる。□

[定理2] ｆをMVKF またはFF/Cとする。ｆが主加法標準形 $f = Fsp \vee Fcp$ (但し、Fspは単積項の和、Fcpは相補最小項の和)で、また主乗法標準形 $f = Fsc \cdot Fcc$ (但し、Fscは単和項の積、Fccは相補最大項の積)で表現されているとする。この時ｆは次のような標準形で一意に表現される。

(T2.1)　$f = Fsp \vee (1/2) \cdot Fsc$

(T2.2)　$f = Fsc \cdot (Fsp \vee 1/2)$　□

[定義6] ｆをMVKFまたはFF/Cとし、Vをその真理値の集合とする。ｆの積項αについて $\alpha(\mathbf{a}) \leqq f(\mathbf{a})$　$(\forall \mathbf{a} \in V^n)$が成立するときαを<u>内項</u>という。ｆの内項のうち、それから文字を取り除く、または定数を大きくしたら、もはやｆの内項でなくなるものを<u>主項</u>という。□

[定義7] すべての主項の和を<u>主項展開</u>という。□

[定理3] ｆをMVKF またはFF/Cとする。ｆが主乗法標準形 $f = C_1 \cdot C_2 \cdots C_m$ (但し、C_iは単和項または相補最大項)で表現されているとき、これを加法形式に展開して $f = \alpha_1 \vee \alpha_2 \vee \cdots \vee \alpha_t$を得たとする。このとき$\alpha_1 \vee \alpha_2 \vee \cdots \vee \alpha_t$はｆの主項展開である。(拡張されたNelsonの定理)□

[定義8] ｆをMVKFまたはFF/Cとする。このときｆを表現する加法形式のうちで項数最少、更に項数が同じなら文字数が最少のものをｆの<u>最簡形</u>という。□

[定理4] 最簡形は主項の和で表される。□

[定義9] ｆのいかなる最簡形にも現れる主項を<u>必須主項</u>という。また、ｆのいかなる最簡形にも現れない主項を<u>不必要項</u>という。□

[定理5] ｆをMVKFまたはFF/Cとする。ｆが乗法形式 $f = C_1 \cdot C_2 \cdots C_s$ (但し、C_iは和項)で表現されているとき、これを加法形式に展開して $f = \alpha_1 \vee \alpha_2 \vee \cdots \vee \alpha_t$を得たとする。このとき$\alpha_1 \vee \alpha_2 \vee \cdots \vee \alpha_t$は主項の和である。もしｆの主項である$\alpha_0$が$\alpha_1 \vee \alpha_2 \vee \cdots \vee \alpha_t$の中に現れない場合は$\alpha_0$は不必要項である。□

[定理6] 単積項はすべて必須主項である。□

[定理7] ｆが定理2(T2.1)で与えられた標準形で表されたとする。ｆを加法形式に展開して $f = \alpha_1 \vee \alpha_2 \vee \cdots \vee \alpha_t$を得たとする。このとき$\alpha_1 \vee \alpha_2 \vee \cdots \vee \alpha_t$は主項の和である。もしｆの主項である$\alpha_0$が$\alpha_1 \vee \alpha_2 \vee \cdots \vee \alpha_t$の中に現れない場合は$\alpha_0$は不必要項である。□

On extension of Clause Selection Method to a family of Kleenean switching functions and their minimization

Tomoyuki ARAKI[+]　Hisayuki TATSUMI[+]　Masao MUKAIDONO[++]　Yoshitsugu OHMAE[+]
+Kanagawa Institute of Technology　1030, Shimoogino, Atsugi-shi, Kanagawa 243-02, Japan
++Meiji University　1-1-1, Higashimita, Tama-ku, Kawasaki-shi, Kanagawa 214, Japan

[性質1] MVKF, FF/Cにおいてαを相補項、Cを相補和項とするとき$\alpha \cdot C = \alpha$である。 □

3.節展開法の拡張

定理3により、主乗法標準形で表現されたMVKFおよびFF/Cを加法形式に、拡張された節展開法を用いて展開することにより、その主項展開を求めることができる。また、性質1は、最終的に得られる積項が相補項になる場合には、相補和項に対する展開は不要であることを示している。この事実により木の節点数を減少させるための工夫が以下のアルゴリズムのS1, S2選択の部分で適用できる。

[アルゴリズム1] (拡張された節展開法)

[Step1] 主乗法標準形で表現されたfについて各和項を構成する文字の集合を集めたものを節点とする。この節点は木の根になり非終端節点である。

[Step2] 木の非終端節点をSとする。

(S1選択) もしSに至る直前の枝の文字が定数≦1/2であった場合Sから相補和項以外で文字数最少、頻度和最大の和項を選びC_0とする。

(S2選択) もしSにそれ以前の枝の変数の文字の否定(x_iか$\sim x_i$)を含む和項が存在し、更にこの和項以外に相補和項があれば、このx_iか$\sim x_i$を含む和項をC_0とする。そのような和項が複数ある場合は、それらの内で文字数最少、頻度和最大の和項を選ぶ。

(S3選択) S1, S2選択に該当しない場合、Sから文字数最少、頻度和最大の和項を選びC_0とする。

以上で選ばれたC_0に含まれる文字を頻度順に並べたものをO_sとする。$O_s = \langle L_1, L_2, \cdots, L_i, \cdots \rangle$

[Step3] Sから枝出しを以下のように行う。

(a) SよりO_sの順位の高い文字から枝出しを行い、その枝にはL_iとラベルを付ける。新しい節点$S_k(L_i)$はL_iを含む和項を消去し、更にS1, S2選択のときには$S_k(L_i)$の中のすべての相補和項も消去する(性質1による)。$S_k(L_i)$が空ならば有効節点とする。

(b) $S_k(L_i)$より$L_1, L_2, \cdots, L_{i-1}$を取り除く。この結果、空となる和項があれば無効節点とする。

(c) 操作(a), (b)の結果、有効節点でも無効節点でもなければ、その節点は非終端節点である。

[Step4] 枝出しされてない非終端節点が無くなるまでStep2, 3を繰り返す。

[Step5] 得られた木について、根から有効節点へ至る道の枝のラベルとなっている各文字の積をすべての有効節点について求める。これらはすべてfの内項であり、他の項に包含されるものをすべて取り除けば、残ったものがfのすべての主項である。 (アルゴリズム終わり)

4.最簡形について

定義8のように論理関数fの最簡形を定義した場合、通常、簡単化は定理4より次のような2つのステップをとおして行われることが多い。

① fのすべての主項を見出す。

② fを被覆する最少の主項の組み合わせを見いだす。

しかしながら、簡単化を考える場合①のスッテプでは必ずしもすべての主項を見出す必要はなく、不必要項はなるだけ除去されているほうが望ましい。従って、①のスッテプを行う手法として定理5より次のようなアルゴリズムが考えられる。

[アルゴリズム2]

[Step1] 乗法形式で表現されたfについて各和項を構成する文字の集合を集めたものを節点とする。この節点は木の根になり非終端節点である。

[Step2]〜[Step4]はアルゴリズム1と同じ。

[Step5] 得られた木について、根から有効節点へ至る道の枝のラベルとなっている各文字の積をすべての有効節点について求める。これらはすべてfの内項であり、他の項に包含されるものをすべて取り除けば、fの最簡形に現れ得る主項はすべて含まれている。

(アルゴリズム終わり)

これはファジィ節展開法の拡張となっている。

アルゴリズム2では入力として乗法形式を仮定しており、①のスッテプの出力としては主項を加法形式で②のステップに渡すことになる。通常②のスッテプは、与えられた論理関数の最少項展開(MVKF, FF/Cの主乗法標準形に対応する)に含まれる積項を被服する最小の主項の組み合わせを求める最小被服問題に帰着される。もしステップ①への入力が乗法形式で与えられた場合、②のステップを行うために、与えられた関数の最小項展開(主加法標準形)を別の工程で作ってやらねばならない。即ち、最小項展開(主加法標準形)に一度変換することを必要とする。一般に論理関数は加法形式で取り扱われることが多く、入力も出力も加法形式で表現したい場合が少なくない。以下では定理2、定理6、定理7に基づき主加法標準形で与えられた論理関数の最簡形を求めるためのアルゴリズムを示す。

[アルゴリズム3]

[Step1] 主加法標準形で表現されたfの積項について、単積項の集合をT_s相補項の集合をT_cとする。

[Step2] fの各積項を構成する文字の集合を集めたものを節点とする。この節点は木の根になり非終端節点である。

[Step3] 木の非終端節点をPとする。

Pから文字数最少、頻度和最大の積項を選びD_0とし、またD_0に含まれる文字を頻度順に並べたものをO_pとする。

$O_p = \langle L_1, L_2, \cdots, L_i, \cdots \rangle$

[Step4] Pから枝出しを以下のように行う。

(a) PよりO_pの順位の高い文字から枝出しを行い、その枝にはL_iとラベルを付ける。新しい節点$P_k(L_i)$はL_iを含む積項を消去する。$P_k(L_i)$が空となれば有効節点とする。

(b) $P_k(L_i)$に至る直前の枝の文字が定数>1/2であった場合は無効節点となる。

(c) $P_k(L_i)$に至る道の枝に相補的な変数の文字(x_iと$\sim x_i$)が存在すれば無効節点となる。

(d) $P_k(L_i)$より$L_1, L_2, \cdots, L_{i-1}$を取り除く。この結果、空となる積項があれば無効節点となる。

(e) 操作(a), (b), (c), (d)の結果、有効節点でも無効節点でもなければ、その節点は非終端節点である。

[Step5] 枝出しされてない非終端節点が無くなるまでStep3, 4を繰り返す。

[Step6] 得られた木について、根から有効節点へ至る道の枝のラベルとなっている各文字の積をすべての有効節点について求める。これらはfを乗法形式で表したときの単和項であり、他の和項に包含されるものは除去しておく。

[Step7] Step6で得られた乗法形式の関数をアルゴリズム2により展開し、得られた積項すべてに対して1/2との積をとり、その集合とT_0とする。

[Step8] T_0からT_sの積項に包含されるものを除去する。

[Step9] T_0とT_cの間で最小被覆をとり、T_0から最簡形に必要となる相補項の主項を選び出す。

[Step10] T_sとStep9で選び出された主項で構成される関数が最簡形である。 (アルゴリズム終わり)

5.あとがき

アルゴリズム3には更に改善できると予想される部分が多く残っており、今後は計算機による実験も含めて更に良いアルゴリズムとしていきたい。

参考文献

[1] 高木, 向殿:"多値クリーネ関数の基本的性質", 信学論(D-I), Vol.J74-D-I, No.12, pp.797-804 (1991)

[2] 高木, 向殿:"多値クリーネ関数の論理式表現", 信学論(D-I), Vol.J75-D-I, No.2, pp.69-75 (1992)

[3] Nelson, R.J.:"SIMPLEST NORMAL TRUTH FUNCTION", J. Symbolic Logic, Vol.20, No.2, pp.105-108 (1954)

[4] 上林, 岡田, 矢島:"節展開法を用いた論理関数の主項の生成", 信学論(D), VOL.J62-D, No.2, pp.89-96 (1979)

[5] 山口:"ファジィ節展開法を用いたファジィ論理関数のファジィ主項生成アルゴリズム", 信学論(D), Vol.J-73-D-1, No.8, pp.657-663 (1990)

[6] Araki, T., Tatsumi, H., Mukaidono, M and Ohmae, Y., "On the Fuzzy Switching Function with Arbitrary Constans and Generalization of Nelson Theorem for Fuzzy Logic", AFSS'93, submitted

[7] 荒木, 巽, 向殿:"Nelsonの定理の拡張について", 多値論理研究ノート, Vol.15, No.12, pp.12-1〜12-10 (1992)