3Ｅ－６

二分決定グラフを用いた不確定入力に対する論理関数の評価アルゴリズム

井口幸洋　向殿政男
明治大学 理工学部 情報科学科

概要：あるｎ変数の２値論理関数ｆに対して入力変数のうち幾つかは０または１に確定しているが，残りの幾つかの値は不確定，即ち，０か１か不明であることがある．この様な不確定な入力を含む場合の論理関数の値の定め方と，論理関数が２段のAND-OR論理式によって表現されている時，その値を求めるアルゴリズムについては既に報告されている[1][2]．一方，AkersやBryantらによって提案されている二分決定グラフ[3][4]を用いた論理関数の表現方法が実際の大規模な論理回路の表現方法として有望視されている[5]．本報告では論理関数の表現方法に二分決定グラフを用いることで，不確定入力に対する論理関数の評価を高速に行うアルゴリズムを提案する。

１．不確定入力に対する論理関数の評価

論理関数を評価する場合，入力変数のうち幾つかが０か１か不明である場合がある。典型的な例としては内部状態が不明な順序回路の論理シミュレーションがある．．

真理値０と1/2と１からなる集合をＶとする．ここで1/2は０か１か不明であることを表す真理値とする．

$$V = \{0,\ 1/2,\ 1\}$$

対象とする論理関数はｎ入力１出力の論理関数とする．今，入力 $a \in V^n$ の1/2を０か１で置き換えて得られる確定した入力の集合を a^* とする．

［例１］入力 $a = (1/2, 1, 1/2, 0)$ ならば

$a^* = \{(0,1,0,0), (0,1,1,0), (1,1,0,0), (1,1,1,0)\}$.

入力ａに対する３値論理関数 $\underline{f}(a)$ を次の様に定める

定義１[1]：
$$\underline{f}(a)=0 \iff f(a^*)=\{0\}$$
$$\underline{f}(a)=1 \iff f(a^*)=\{1\}$$
$$\underline{f}(a)=1/2 \iff f(a^*)=\{0,1\}$$

定義１は，不確定な入力がもし確定したとする時にとり得る全ての入力パターンに対して，そのすべての出力値が０(１)であれば出力値を０(１)とし,出力値に０と１の両方の場合があれば出力も不確定として1/2を割り付けることを意味している。

２．データ構造と評価アルゴリズム

論理関数がAND-OR２段の論理式で表されている時，不確定入力に対する論理関数の値を求めるアルゴリズムについては既に報告されている[1][2]．ここでは，不確定入力を論理関数に加えた時の出力値を求めたい場合，論理関数を二分決定グラフで表現することにより効率よく値を求める方法を提案する。

［例２］図１のカルノー図で表される論理関数を二分決定グラフで表した例を図２に示す。図１上のループは不確定入力の被覆を表している。例えば，入力 a_4 は $a_4=(x_1,x_2,x_3,x_4) = (1/2, 1, 1/2, 0)$ であるから，例１に示したように(0,1,0,0), (0,1,1,0), (1,1,0,0), (1,1,1,0) の４つの確定した入力で構成されている．それらを１つのループでまとめて表示している．

二分決定グラフの上で不確定入力に対する論理関数の値を求めてみよう。グラフ上のノード内の数字は変数を表し（例えば①は x_1 を意味している），$\boxed{0}$と$\boxed{1}$は２値論理関数の値０と１をそれぞれを表している。今，すべての入力値が確定していれば，一番上のノードからその変数に対応する入力の値に従って進めば論理値０か１のどちらかに必ず到達し，値が決定される。問題は入力値が 1/2 のときの処理である．ここでは簡単のために図２(a)(b)に示した各入力に対してその論理値を求めてみよう。

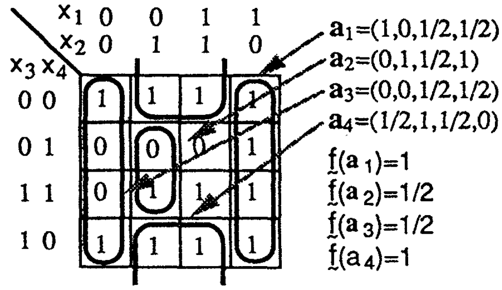
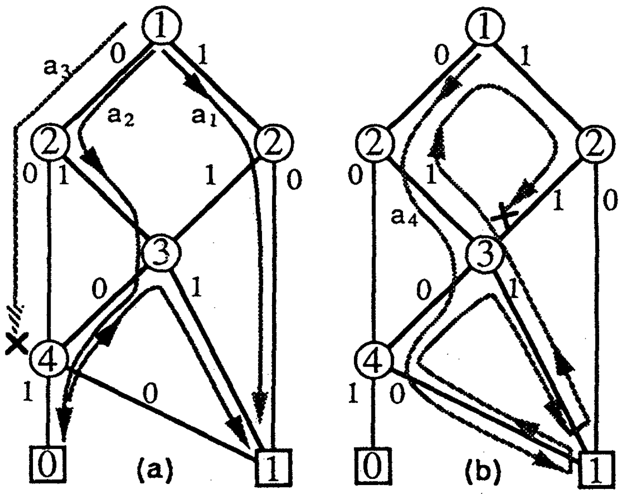


図１　４変数の論理関数の例



二分決定グラフの上で不確定入力に対する論理関数の値を求める過程を図示したもの，但し，×印は計算を途中で打ち切ることが出来たことを示す。

図２　図１の論理関数を二分決定グラフで表した例

An Evaluation Algorithm of Switching Functions for an Unknown Input Based on Binary Decision Diagrams
Yukihiro IGUCHI, Masao MUKAIDONO
Meiji University

(1) $\mathbf{a}_1$=(1,0,1/2,1/2) に対しては，まず最初のノード①で x_1 への入力は１であるから１側に進み②のノードへ移る．更に x_2＝０であるから，０側に進み $\boxed{1}$ に到達し $\underline{f}(a_1)$=1 が決定する．

(2) $\mathbf{a}_2$=(0,1,1/2,1) に対しては，まず最初のノード①で x_1 =0であるから0側に進み②へ移る。次は x_2＝１であるから，１側に進み③に移る．次に, x_3=1/2であるから０側と１側の両方を調べなければならない．とりあえず０側に進み④に移り x_4=1であるから $\boxed{0}$ のノードに行き着く，更に未探索の③のノードまでバックトラックして１側に進むと今度は $\boxed{1}$ のノードに到達する． よって, $\boxed{0}$ と $\boxed{1}$ の両方に到達したので $\underline{f}(a_2)$=1/2 が決定する．

(3) $\mathbf{a}_3$=(0,0,1/2,1/2)に対しては最初のノード①で０側に次の②のノードで０側に進み，④のノードに到達する．ここで入力x1 とx2 以外は1/2であり,確定した入力についてはすべてノードを通過してきたにもかかわらず, $\boxed{0}$にも$\boxed{1}$にも到達しないのでこの時点で $\underline{f}(a_3)$=1/2 が決定する．

(4) $\mathbf{a}_4$=(1/2,1,1/2,0) に対しては，図２(b)に示したように順時探索していくと，既に前の探索で訪れたノードであるノード③に到達する．この場合，これ以降の探索は不必要なので計算を打ち切ることが出来 $\underline{f}(a_4)$=1 が決定する．

以上，入力に1/2 が含まれている時の処理の例を示した．ここで特徴的な事をまとめてみよう．まず，(1) の例は入力が1/2になっている変数のノードを通過することなしに値が決定する場合があることを示している。(2) は典型的な1/2 の処理であって， 1/2 の変数のノードで２つに枝分かれして探索を行っている．一方，(3) では枝分かれせずに1/2 が決定する例である。(4) も枝分かれして探索を次々に行っている．入力の1/2 の個数が増えれば(2) や(4) のように枝分かれしてそれだけ探索量が爆発的に増加しそうである。しかし，(4) のように以前訪れたノードはもう１回調べることは不要であるから計算を打ち切ることができることと，一般に1/2 の個数が増えれば入力の被覆が大きくなるので出力値が1/2となる可能性が増え，探索中に $\boxed{0}$ と $\boxed{1}$ の両方が出てきた時点で，出力値が1/2 に決定され，それ以上の計算が不要になること．更に(1)や(3) の例のように入力に1/2 があっても必ずしも枝分かれするわけではないことなど不確定入力の個数が増えてもそれほど探索量が増えないことが期待できる．

本評価法を実現するアルゴリズムを図３に示す．

３．まとめ

不確定入力を論理関数に加えた時の出力値を求めたいとき，論理関数を二分決定グラフで表現することによりその値を求める方法を提案した。本評価法において論理関数を表現する二分グラフがより小さい方が，メモリの点だけでなく，評価にかかる時間の点においても優れていると思われる。二分グラフの大きさは展開する変数の順序によって大きく変化することが知られており，より小さいグラフを求める手法や工夫を加えたグラフが提案されている[4]-[6].

今後の課題としては，現在，実際の論理関数に対して計算機上で実験を行っているところであるので計算機実験をもとに上記のことを含めて検討を行いたい。論理関数にdon't careがある場合についても若干の変更を加えることで求めることができるが，実験結果を含めて稿を改めて報告する予定である。

参考文献

[1] 向殿:不確定入力に対する論理関数の評価法について,情報処理学会設計自動化研究会夏期シンポジウム (1981).

[2] 井口,向殿：不確定入力に対する論理関数の評価の拡張と比較,情報処理学会設計自動化研究会資料21-3 (1984).

[3] S.B.Akers: Binary Decision Diagrams, IEEE Trans. Comput., Vol. C-27, No.6, pp. 509 -516 (1978).

[4] R. E. Bryant: Graph Based Algorithms for Boolean Function Manipulation, IEEE Trans. Comput., Vol. C-35, No.8, pp. 677-691 (1986).

[5] 藤田,藤沢,川戸：２分決定グラフを用いた論理照合アルゴリズムの評価と改良,情報処理学会設計自動化研究会資料43-2 (1988).

[6] 湊,石浦,矢島:論理関数の共有二分決定グラフによる表現とその効率的処理手法,情報処理学会論文誌 Vol. 32, No.1(1991).


```
p は２分決定グラフのルートノードを指すポインタ;
a は不確定入力を含む入力パターン;
S ← 確定している入力変数の集合;
{定数関数 f = 0(1)の場合はどのような入力に対しても出力は常に0(1)であるから以下の手続きを適用しない}
lastValue ← 未決定;
eval(p, s) で出力値が求まる.
-----------------------------------------------------------
function eval(p,S):V;
begin
  x ← p が指しているノードの値;
  if  x = [0]   then  return(0);
  if  x = [1]   then  return(1);
  if  S = 空集合   then return(1/2);
  if そのノード は 以前に訪れたノード then
    return (lastValue);
  if  x ∈ S  then
    begin
      S ← S - {x};
      if a(x) = 0 then
        eval(0 側のノードへ, S)
      else
        eval(1 側のノードへ, S)
    end
  else
    begin
      zValue ← eval(0 側のノードへ, S);
      if zValue = 1/2 then return(1/2);
      if lastValue = 未決定 then
        lastvalue ← zValue
      else
        if lastValue<>zValue then
          return(1/2);
      oValue ← eval(1 側のノードへ, S);
      if oValue = 1/2 then return(1/2);
      if zValue = oValue then
        return(oValue)
      else
        return(1/2)
    end
end; {of eval}
```

図３ 評価アルゴリズム