数理解析研究所講究録
943巻 1996年 29-34

否定数限定反転回路の複雑さ

西野 哲朗 (Tetsuro Nishino)

電気通信大学　電子情報学科

概要　使用できるNOTゲートの個数を制限した組み合わせ論理回路を、否定数限定回路という。ブール関数の否定数限定回路計算量を評価する際には、否定数が限定された反転回路（すべての入力変数の否定を計算する回路）の複雑さが重要となる。本稿では、この否定数限定反転回路のサイズの上界と下界に関する、筆者らによる最近の研究結果 [3, 8] を要約して述べる。

1　回路計算量理論

Shannonは1949年の論文で、**関数の複雑さ**を、その関数を計算する最小ブール回路の素子数で測ることを提案した。当時はまだ計算機のハードウェアが非常に高価であり、Shannonの動機は、計算に必要なハードウェアの量を最小化することであった。一般に、ブール回路を用いて関数の複雑さを解析する分野を、**回路計算量理論**という。

値として0または1を取る変数を**ブール変数**という。$B = \{0,1\}$ とする。関数 $f : B^n \to B^m$ を n 入力 m 出力の**ブール関数**という。以下では、n 入力1出力のブール関数のことを、単に n 変数ブール関数と呼ぶ。ブール関数の値はブール回路によって計算できる。**ブール回路**は非循環有向グラフで表される。入次数が0の頂点は**入力**と呼ばれ、変数 x_i または定数（0または1）でラベル付けされる。入次数 $k > 0$ の頂点は**ゲート**と呼ばれ、k 変数ブール関数でラベル付けされる。頂点の入次数を**ファンイン**と言い、出次数を**ファンアウト**と言う。特に断らないかぎり、ブール関数としてはAND, OR, NOTのみを考える。回路内の1つの頂点が**出力**頂点として指定される。回路の**サイズ**とは、その回路に含まれるゲートの個数のことであり、回路の**深さ**とは、入力から出力への回路内の最長経路長のことをいう。

$\mathbf{N}$ を自然数の集合とし、$\{0,1\}^n$ を長さ n の2進列の集合とする。さらに、$\{0,1\}^*$ で長さが有限のすべての2進列の集合を表す。$f : \{0,1\}^n \to \{0,1\}$ としよう。このとき、f の値を計算する最小回路のサイズを $C(f)$ で表す。また、$g : \{0,1\}^* \to \{0,1\}$ とし、$h : \mathbf{N} \to \mathbf{N}$ としよう。g の**回路計算量**が h であるとは、すべての n に対して $C(g_n) = h(n)$ となるときをいう。ただし、g_n は g を $\{0,1\}^n$ 上に制限した関数である。

また、$\{0,1\}^*$ の部分集合を**言語**という。つまり、言語とは2進列の集合である。**言語の回路計算量**とは、その言語の特性関数の回路計算量のことをいう。ここで、言語 L の特性関数 f_L とは、$x \in L$ に対しては $f_L(x) = 1$ となり、$x \notin L$ に対しては $f_L(x) = 0$ となる関数のことをいう。

回路計算量とTuring機械上の計算量との関係は、Savageによって初めて明かにされた。次の定理は、PippengerとFischerによって示された（証明は [6] 参照）。

定理 1　言語 A が $\mathrm{TIME}(T(n))$ に属するならば、A の回路計算量は $O(T(n)\log(T(n)))$ である。　□

したがって、P に属する言語は多項式サイズのブール回路によって認識できる（しかし、この逆は成り立たない [6]）。定理 1 からもわかるように、ブール回路は、計算能力に関して Turing 機械と密接な関係を持っており、回路サイズに対する十分大きな下界は、Turing 機械の時間計算量に対する下界をただちに与える。

定理 1 より、$P \subsetneq NP$ であることを示すには、例えば、NP に属するある具体的な関数について、その関数の回路計算量の超多項式（多項式を超える）下界を示せればよいが、一般に、具体的な関数に対する、回路計算量の強い下界を求めることは非常に難しい。

具体的な関数の回路計算量の下界が示せれば、それは特定の関数の本質的な難しさを示したことになる。その意味でこのような下界を求めることは大変重要であるにもかかわらず、現在までに知られている最良の下界は、Blum があるブール関数に対して示した $3n$ の下界に過ぎない。つまり、現在我々は、一般の回路サイズに関しては、非常に弱い下界しか示すことができない。

ある問題が実際に難しいことを証明するための 1 つのアプローチは、「計算モデルに制限を加え、可能なアルゴリズムのクラスを制限する」ことである。実際、1980 年代に入り、回路の形状に制約を課すことにより、回路計算量のいくつかの強い下界が示された。特に、Razborov は、「n 変数**クリーク関数**を計算する**単調回路**は、n に関する指数オーダの個数のゲート（AND, OR ゲート）を必要とする」ことを示した。次節では、この結果について紹介する。回路計算量理論の詳細については、[4, 6] 等を参照されたい。

2 単調回路

単調回路とは、AND ゲートと OR ゲートから成り、NOT ゲートは含まない回路のことをいう。ここで、各ゲートのファンインは 2 であり、ファンアウトは無制限であるとする。ブール関数 f が**単調**であるとは、通常のブール順序の下で、$x \leq y$ ならば $f(x) \leq f(y)$ となるときをいう。単調回路で計算可能な関数だけが単調関数である。単調関数の**単調回路計算量**とは、その関数を計算する最小の単調回路のサイズのこととする。

計算量理論において重要な多くの関数は単調である。例えば、次のようなクリーク関数を考えてみよう。**クリーク関数**（$\mathrm{CLIQUE}_{k,n}$ と記す）は、$\binom{n}{2}$ 個の変数を持ち、各変数は n 頂点から成るグラフの可能な各辺に対応する。$\mathrm{CLIQUE}_{k,n}$ は、入力されたグラフ中に、ある k 頂点から成るクリーク（完全グラフ）が含まれるときに値 1 を取り、さもなければ値 0 を取る。クリーク関数は単調である。なぜならば、入力グラフに新たに辺を付け加えても、入力グラフ中にそれ以前に存在したクリークが消えることはないからである。

単調回路に対しては、強力な下界が知られている。Razborov は、クリーク関数の単調回路計算量に関するサイズ $n^{\Omega(\log n)}$ の超多項式下界を得た。すなわち、Razborov は以下の定理を証明した。

定理 2 $k \leq n^{1/4}$ に対し、関数 $\mathrm{CLIQUE}_{k,n}$ の単調回路計算量は $n^{\Omega(\sqrt{k})}$ である。 □

この結果以前に知られていた、具体的な単調関数の単調回路計算量に関する最良の下界は、Tiekenheinrich による $4n$ にすぎなかった。その直後に Andreev は、Razborov と類似の手法を用いて、NP に属するある単調問題に対する（ただの超多項式ではな

く）指数下界を示した。このことは、クリーク関数に対する指数下界の存在を意味している。というのは、クリーク関数は「単調 NP」において（多項式単調射影に関して）完全だからである。Alon と Boppana は、Razborov が用いた組合せ論の議論を強め、$\mathrm{CLIQUE}_{k,n}$ （ただし $k = n^{2/3}$）の単調回路計算量の $2^{\Omega((n/\log n)^{1/3})}$ 下界を示した。

もし単調関数を計算する（NOT ゲートを含む）一般の回路を、サイズを多項式倍に増やすだけで等価な単調回路に変換できることが示せれば、上の Razborov の下界は一般の回路計算量に対しても適用でき、したがって、$\mathrm{P} \subsetneq \mathrm{NP}$ が示されることになる。しかし、Razborov 自身がこの可能性を否定した。すなわち彼は、クリーク関数の下界のときと同様の手法を用いて、P に属することが知られている 2 部グラフのマッチング問題が、超多項式サイズの単調回路を必要とすることを証明した。さらに Tardos は、P に属する他の単調関数に関して、このギャップを真に指数にまで広げた。

単調回路計算量と一般回路計算量のあいだに指数的ギャップがあるにもかかわらず、2 つの計算量が多項式的に関係しているような特殊な関数のクラスが存在する。これは Berkowitz によって提案されたスライス関数のクラスである。関数 f は、ある整数 k に対し、x 中の 1 の個数が k より少ないときは $f(x)$ の値が 0 であり、1 の個数が k より多いときは $f(x)$ の値が 1 である（しかし 1 の個数がちょうど k のときは $f(x)$ の値は任意でよい）とき**スライス関数**と呼ばれる。スライス関数は制限されているように見えるけれども、NP 完全なスライス関数が実際に存在する。Berkowitz は、スライス関数を計算する一般回路は、多項式個のゲートを付け加えるだけで単調回路に変換できることを示した。したがって、明示的なスライス関数に対する**単調**回路計算量の超多項式下界は、$\mathrm{P} \neq \mathrm{NP}$ を意味する。

3 否定数限定回路計算量について

上で見たように、使用できる NOT ゲートの個数に何の制限もない一般の回路の場合には、具体的な関数の回路計算量については線形下界しか知られておらず、一方、NOT ゲートを 1 個も使用できない単調回路の場合には、クリーク関数の単調回路計算量の指数下界が知られている。しかし、一般の回路計算量と単調回路計算量の間には指数的なギャップが存在する場合がある。それでは、使用できる NOT ゲートの個数を次第に少なくしていったら、回路計算量はどのように変化するだろうか？以下では、この問題に対する、筆者らの最近の結果について紹介する。本節の内容の詳細については、[3, 8] を参照されたい。

F を、$\{0,1\}^n$ 上で定義されたブール関数系 $f_1, \ldots, f_n$ とする。r 個以下の NOT ゲートを含む回路を、r-回路と呼ぶ。$C^r(F)$ で、F を計算する最小の r-回路のサイズを表す。もし F が、r 個だけの NOT ゲートでは計算できない場合には、$C^r(F)$ は未定義とする。**反転回路** I_n とは、ブール関数系 $f_1, \ldots, f_n$ のこととする。ただし、各 $1 \leq i \leq n$ に対し、$f_i(x_1, \ldots, x_n) = \neg x_i$ とする。

F を n 変数ブール関数系とする。ブール束 $\{0,1\}^n$ 内の鎖 C とは、増加列 $a^1 < \ldots < a^k \in \{0,1\}^n$ のことをいう。C 上の F の**減少量**とは、ある j に対して、$F_j(a^{i-1}) > F_j(a^i)$ となるような $i \leq k$ の個数である。$d(F)$ を、任意の鎖 C 上の F の最大減少量のことと定義する。$d(F) \leq n$ であり、かつ、$d(F) = n$ となるのは $F = I_n$ のときであることに注意せよ。

$b(n) = \lceil \log_2(n+1) \rceil$ とする。Markov は、ブール関数系 F を計算するには、$b(d(F))$ 個の NOT ゲートが必要かつ十分であることを示した。したがって、$r \geq b(n)$ に対しては、$C^r(F)$ は常に定義される。以下では、$C^{b(d(F))}(F)$ を関数系 F の**否定数限定回路計算量**と呼ぶ。

筆者らは、[3] において、以下の関係式を示した。

$$C^{b(n)}(F) \leq 2C(F) + O(n \log n)$$

上でも述べたが、現在、具体的なブール関数 f に対する、$C(f)$ の超線形下界は知られていない。上の関係式から、ある具体的なブール関数 f に対する $C^{b(n)}(f)$ の $\omega(n \log n)$ 下界が得られれば、$C(f)$ の $\omega(n \log n)$ 下界も得られることになる。上の関係式の右辺に現れる $O(n \log n)$ の項は、実は、$C^{b(n)}(I_n)$ の値に対応している。$C^{b(n)}(I_n)$ の上界の変遷については次節で述べる。

否定数限定回路計算量は、単調回路計算量とは異なり、すべての関数に対して定義できる。また、使用できる NOT ゲートの個数は、一般の回路では無制限だが、否定数限定回路においては、$b(d(F))$ 個である。さらに、$C(f)$ と $C^{b(n)}(f)$ は、上で示した関係式から非常に密接な関係にあることがわかる。回路計算量の下界を示す場合には、NOT ゲートの個数が制限されていた方が取扱い易いであろうから、具体的な関数について、否定数限定回路計算量の下界を求めることが興味深い研究課題となる。

筆者らは、すでに n 変数**パリティ関数**（入力中の 1 の個数の奇偶を判定する関数）の否定数限定回路計算量の $4n + 3\log(n+1) - c$ 下界など、いくつかの下界を示している [3]。しかし、具体的な関数について、否定数限定回路計算量の強い下界を示すことは、今後の課題である。このような方針で、もし NP に属する具体的な関数の否定数限定回路計算量の超多項式下界が得られれば、上で述べた関係式から、P $\neq$ NP が示されたことになる。

4 否定数限定反転回路のサイズ

1958 年に Markov [7] は、単調関数と $b(n)$ 個の否定を用いて反転回路を構成した。しかし、彼は使用した単調関数の複雑さを考慮しなかった。Akers [2] は 1968 年に、否定数限定反転回路の明示的な構成法を初めて与えた。彼の回路は、$b(n)$ 個の NOT ゲートと正の重みを持つしきい値ゲートを用いており、サイズは $O(n)$ で、深さは $O(\log n)$ である。

以下では、AND, OR, NOT ゲートから成る回路に話を限定する。1974 年に Fischer は、$b(n)$ 個の NOT ゲートを用いて、サイズ $O(n^2 \log^2 n)$、深さ $O(\log^2 n)$ の回路 I_n を構成した。

定理 3 (Fischer [5]) $C^{b(n)}(I_n) = O(n^2 \log^2 n)$ □

Fischer の構成法（とそれに続くすべての構成法）では、ソーティング回路が重要な役割を果たす。Ajtai, Komlós & Szemerédi [1] は、n ビットをソートするサイズ $O(n \log n)$、深さ $O(\log n)$ の単調回路を構成した。もし、Fischer の構成法において、Ajtai-Komlós-Szemerédi ソーティング回路を用いれば、サイズは $O(n^2 \log n)$ に、深さは $O(\log n)$ に、それぞれ減少する。

1994 年に Tanaka と Nishino [8] は、反転回路の複雑さについて研究し、深さが $\theta(\log^2 n)$ となる構成法を用いて、サイズの $O(n \log^2 n)$ 上界を与えた。

定理 4 (Tanaka & Nishino [8]) $C^{b(n)}(I_n) = O(n\log^2 n)$ □

Fischer と Tanaka-Nishino の両構成法では、まず n 個の入力ビット $x_1,\ldots,x_n$ をソートする。そして、ソーティング回路の出力 $y_1,\ldots,y_n$ は、以下の性質を持つ部分回路 M_n （Fischer [5] による）に与えられる。

1. M_n は n 個の 2 進入力 $y_1,\ldots,y_n$ と、n 個の 2 進出力 $z_1,\ldots,z_n$ を持つ。

2. M_n はサイズ $O(n)$、深さ $O(\log n)$ であり、$b(n)$ 個の NOT ゲートを含む。

3. $y_1 \geq y_2 \geq \ldots \geq y_n$ ならば、$1 \leq i \leq n$ なるすべての i に対し、$z_i = \neg y_i$ となる。

x_i の否定は、x_i, y_i, z_i の単調関数として計算することができる。Fischer はこの計算を、$O(n)$ 個のソーティング回路を並列に用いることにより行った。したがって、もし Ajtai-Komlós-Szemerédi ソーティング回路を用いれば、Fischer の構成法により、サイズ $O(n^2\log n)$、深さ $O(\log n)$ の回路が得られる。一方、Tanaka と Nishino は、Valiant [9] の単調 $(n,2n)$-反転回路を用いた。この回路は、$2n$ 個の入力を、それらのうちのちょうど n 個が 1 のときに反転する。この条件は、例えば $x_1,\ldots,x_n,z_1,\ldots,z_n$ に対して成り立つ。Valiant の回路は、サイズ $O(n\log^2 n)$、深さ $O(\log^2 n)$ の単調回路である。

Fischer および Tanaka-Nishino のどちらの構成法においても、回路は次の 3 つの段階に分かれている。ソーティング回路、部分回路 M_nと、出力を計算する最終段階である。上でも述べたように、ソーティング回路はサイズ $O(n\log n)$、深さ $O(\log n)$ の単調回路で実現でき、M_n はサイズ $O(n)$、深さ $O(\log n)$ で実現できる。また、Fischer および Tanaka-Nishino のどちらの構成法においても、最終段階への入力は x_i, y_i, z_i のみである。

1995 年に Beals, Nishino & Tanaka が構成した否定数限定反転回路は、サイズが $O(n\log n)$、深さが $O(\log n)$ であり、それ以前の I_n に対する否定数限定回路のサイズを、少なくとも $\log n$ 分の 1 に改善した [3]。

定理 5 (Beals, Nishino & Tanaka [3]) $C^{b(n)}(I_n) = O(n\log n)$ □

以前の深さ $O(\log n)$ の唯一の構成法では、サイズが $O(n^2)$ よりも真に大きかったことに注意せよ。

Beals-Nishino-Tanaka の否定数限定反転回路も 3 段階から成り、最初の 2 段階はやはりソーティング回路と Fischer の M_n であるが、最終段階において新たな工夫がなされた。すなわち、最終段階の計算においてソーティング回路の中間結果を用いることで、最終段階がサイズ $O(n\log n)$、深さ $O(\log n)$ の回路で構成できることを示した。そのときの最終段階は、上下逆の Ajtai-Komlós-Szemerédi ソーティング回路と考えることができる。

上の 3 つの構成法の違いは、M_n 部分回路の出力のとらえ方の違いに起因している。もちろん、3 つの構成法すべてにおいて、M_n の出力はまったく同一の n 個の関数になっている。しかし、3 つのアプローチそれぞれにおいて、これらの出力はまったく異なる扱い方をされている。Fischer の最初の否定数限定反転回路においては、M_n の入力と出力は、それぞれ（$x_1,\ldots,x_n$ の）**しきい値関数とその否定**と考えられた。Fischer はこれらのビットを、入力中の 1 の個数 k にしたがって、適当な部分回路の出力を選択するのに用いた。このような部分回路は、$0 \leq k \leq n$ なる各 k に対して $n+1$ 個存在する。

Tanaka と Nishino は、$x_1,\ldots,x_n$ と M_n の出力を合わせると、ちょうど n 個の 1 を含む $2n$ ビットの列が得られることに注目して、$n+1$ 個の異なる場合を並列に考えることを避けた（このような $2n$ ビットの列は、Valiant の単調 $(n,2n)$-反転回路を用いて反転することができる）。彼らの構成法では、M_n の出力は、入力 $x_1,\ldots,x_n$ の釣り合いを保つために使われている。Beals-Nishino-Tanaka の反転回路では、M_n の出力は x_i の否定の順列であると考えられた。したがって、これらのビットを適当な位置に並べ変えることが、最終段階の仕事となった。この並べ変えは、第 1 段階で行われたソーティングによる並べ変えを、順次もとに戻していくことに対応している。

筆者は、Beals-Nishino-Tanaka 否定数限定反転回路のサイズの $O(n\log n)$ 上界は、おそらく最適であろうと予想している。一方、$C^{b(n)}(I_n)$ の下界として知られているのは、筆者らによる $5n+3\log(n+1)-c$ のみである [8]。

References

[1] Ajtai M., Komlós J., and Szemerédi E., An $O(n\log n)$ sorting network, in *Proceedings of the 15th Annual ACM Symposium on the Theory of Computing*, 1983, pp.1-9.

[2] Akers S. B., On maximum inversion with minimum inverters, *IEEE Trans. Comput.*, **17** (1968), pp.134-135.

[3] Beals, R., Nishino, T., and Tanaka, K., More on the complexity of negation-limited circuits, in *Proceedings of the 27th Annual ACM Symposium on the Theory of Computing* (May-June 1995).

[4] Dunne, P. E., *The Complexity of Boolean Networks*, Academic Press (1988).

[5] Fischer, M. J., The complexity of negation-limited networks – a brief survey, in *Lecture Notes in Compter Science*, **33**, Springer-Verlag, Berlin, 1974, pp.71-82.

[6] J. van Leeuwen (Ed.), *Handbook of Theoretical Computer Science, Vol. A, Algorithms and Complexity*, The MIT Press / Elsevier (1990)（邦訳：「コンピュータ基礎理論ハンドブック I」, 丸善, 第 14 章「有限関数の複雑さ」, by R. B. Boppana and M. Sipser, 西野哲朗訳）.

[7] Markov, A. A., On the inversion complexity of a system of functions, *J. Assoc. Comput. Mach.*, **5** (1958), pp.331-334.

[8] Tanaka, K., and Nishino, T., On the complexity of negation-limited Boolean networks, In *Proceedings of the 26th Annual ACM Symposium on the Theory of Computing* (May 1994), pp.38-47.

[9] Valiant, L. G., Negation is powerless for Boolean slice functions, *SIAM J. Comput.*, **15** (1986), pp.531-535.