ショートノート

論理式の飛び越し型評価に関する一定理†

佐々政孝†† 中田育男††

論理式を飛び越し型（ショート・サーキット方式）で評価（計算）する際の最適化に関していくつかの公式と定理を示す．(1)まず，論理式に対し，各論理一次子の評価時間と真偽確率が与えられたとき，論理式全体の飛び越し型評価の時間期待値を見積もる公式を導く．(2)次に，and, or 演算の可換律と結合律を利用して，部分論理式の評価順序を入れ換えることによって評価の時間期待値を最小化するための定理を，(1)で得られた公式を用いて示す．

1. はじめに

論理式とは，論理一次子（関係式や論理変数など）を論理演算子（ここでは，and, or, not）で結合したものである．論理式を評価（計算）する際，途中までの評価で全体の真偽値が決まったら，残りの評価を飛び越すようにした，いわゆる飛び越し型（ショート・サーキット方式）評価法は，コンパイラやデータベース検索における重要な（時間）最適化技術の一つである[1)~4)]．

ところが，従来の研究では，and, or 演算の可換律や結合律を考慮にいれていなかったため，更なる最適化の機会をのがしていた．本稿では，この性質を利用して部分論理式の評価順序を入れ換えることにより，論理式の飛び越し型評価を最適化する一つの方法を示す．そのために，まず

（1） 論理式を飛び越し方式で評価する場合は，一般にその値が真か偽かで評価時間が異なる．そこで，各論理一次子の（真偽それぞれの）評価時間期待値と真偽確率が与えられたとき，その論理式全体の（真偽それぞれの）評価時間期待値と真偽確率を見積もる公式を導く．

（2） 次に，(1)で得られた時間期待値が最小になるようにするには，部分論理式の評価順序をどのように入れ換えればよいかを示す定理を与える．

2. 飛び越し型評価の時間期待値

以下で導出する時間期待値の公式が適用できるために，次の前提を仮定する．

前提 1 与えられた論理式に対し，部分論理式の真偽確率とその評価時間は，それより左にある他の部分論理式には依存しない．

与えられた論理式は構文解析され，構文木の形になっているとする．構文木の葉は論理一次子である．同じ演算子が続けて現れた場合は，結合律を利用して，構文木では複数の子供をもった一つの演算子節（n 項演算子節）で表す．たとえば，b_1, b_2, b_3, b_4 が論理一次子のとき，

$(b_1$ **or** b_2 **or** $b_3)$ **and** b_4

は，図1のような構文木で表す．

構文木中の節 b_i に対して，p_i, T_i^t, T_i^f を次のように定義する（以下，説明の便宜のため，構文木の節と，それに対応する部分木，または対応する部分論理式とを同一視することがある）．

● p_i は，節 b_i の評価（正しくは，節 b_i に対応する部分式の評価）が真になる確率（$1-p_i$ は，節 b_i の評価が偽になる確率であるが，これを $\bar{p}_i$ とも表す）．$p_i=0$ や 1 は特殊な場合であるので，以後は，$0<p_i<1$ と仮定する．

● T_i^t, T_i^f は，節 b_i の評価の結果がそれぞれ真になるとき，偽になるときの評価の時間期待値．

節 b_i（に対応する部分式）の評価の時間期待値を T_i とすると，これは，値の真偽を平均すれば，

$$T_i=p_iT_i^t+\bar{p}_iT_i^f$$

で決まる．

and 節に対する公式（図2(**a**)）

図2(a)の and 節 b について，子供 b_i の p_i, T_i^t, T_i^f あるいは T_i $(i=1,\cdots,n)$ が与えられているとき，節 b（に対応する部分式）の p, T^t, T^f, T は次の公式で計算できる（導出方法を付録1に示した）．

$$p=p_1p_2\cdots p_n=\prod_{i=1}^{n}p_i \qquad (1)$$

† A Theorem on Short-Circuit Evaluation of Boolean Expressions by MASATAKA SASSA and IKUO NAKATA (Institute of Information Sciences and Electronics, University of Tsukuba).

†† 筑波大学電子情報工学系

$$T^t=T_1^t+T_2^t+\cdots+T_n^t=\sum_{i=1}^{n}T_i^t \qquad (2)$$

$$T^f=\left[\sum_{i=1}^{n}\left[\left(\prod_{j=1}^{i-1}p_j\right)\bar{p}_i\left\{\left(\sum_{j=1}^{i-1}T_j^t\right)+T_i^f\right\}\right]\right]\Big/\bar{p}$$
$$(=(T-pT^t)/\bar{p}) \qquad (3)$$

$$T=pT^t+\bar{p}T^f \quad \text{または}$$
$$T=T_1+p_1T_2+p_1p_2T_3+\cdots+p_1p_2\cdots p_{n-1}T_n$$
$$=\sum_{i=1}^{n}\left(\prod_{j=1}^{i-1}p_j\right)T_i \qquad (4)$$

T についての式(4)は，子供の p_i と T_i だけで決まり，T_i^t や T_i^f に陽にはよらないことが注目される．

or 節に対する公式（図2(b)）

or 節は，and 節と双対 (dual) であるので，or 節についての公式は，and 節についての公式において，真と偽とを入れ換えて

$$p\leftrightarrow\bar{p},\quad p_i\leftrightarrow\bar{p}_i,$$
$$T^t\leftrightarrow T^f,\quad T_i^t\leftrightarrow T_i^f$$

とすれば容易に得られる：

$$p=1-\bar{p}_1\bar{p}_2\cdots\bar{p}_n=1-\prod_{i=1}^{n}\bar{p}_i$$

$$T^t=\left[\sum_{i=1}^{n}\left[\left(\prod_{j=1}^{i-1}\bar{p}_j\right)p_i\times\left\{\left(\sum_{j=1}^{i-1}T_j^f\right)+T_i^t\right\}\right]\right]\Big/p$$
$$(=(T-\bar{p}T^f)/p)$$

$$T^f=T_1^f+T_2^f+\cdots+T_n^f=\sum_{i=1}^{n}T_i^f$$

$$T=T_1+\bar{p}_1T_2+\bar{p}_1\bar{p}_2T_3+\cdots+\bar{p}_1\bar{p}_2\cdots\bar{p}_{n-1}T_n$$
$$=\sum_{i=1}^{n}\left(\prod_{j=1}^{i-1}\bar{p}_j\right)T_i$$

not 節に対する公式（図2(c)）

飛び越し型評価法では，not 演算に対する評価は行わず，真と偽の値が反転するだけである：

$$p=\bar{p}_1,\quad T^t=T_1^f,\quad T^f=T_1^t$$
$$T=pT^t+\bar{p}T^f=\bar{p}_1T_1^f+p_1T_1^t=T_1$$

論理式全体の真偽確率と評価時間期待値は，上述の公式を構文木上で葉から根に向かって上向きに適用すれば得られる．

例 1 次のような論理式を考える．

(eoln **or** ch=tab **or** ch=' ') **and** flag

各論理一次子の p_i, T_i^t, T_i^f が次のように与えられているとしよう*.

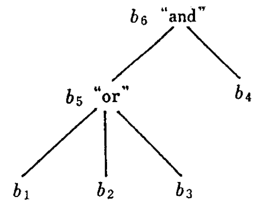
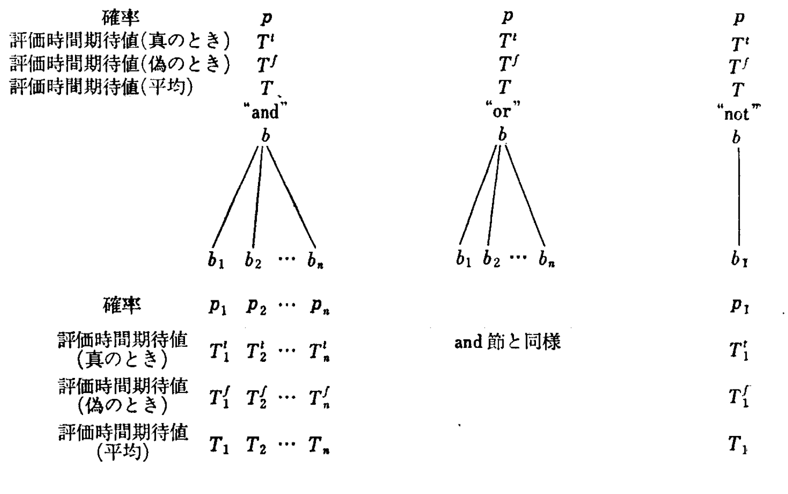


図 1 論理式「(b_1 **or** b_2 **or** b_3) **and** b_4」の構文木
Fig. 1 Syntax tree of an example boolean expression "(b_1 **or** b_2 **or** b_3) **and** b_4".



(a) and 節 (b) or 節 (c) not 節

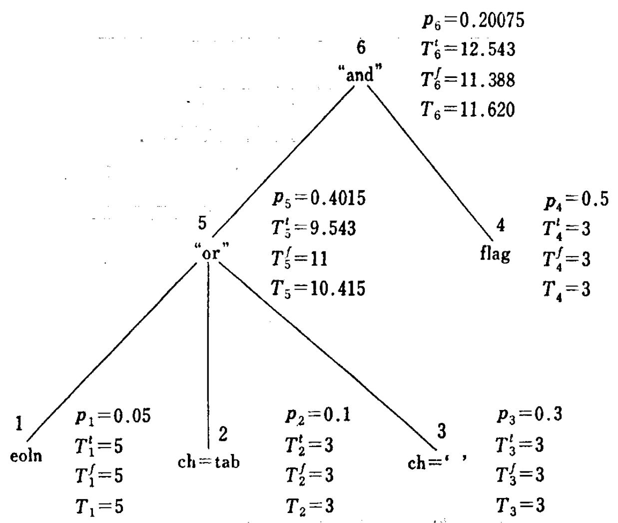
図 2 and, or, not 節の確率と評価時間期待値
Fig. 2 Probabilities and expected evaluation times for "and", "or" and "not" nodes.

節	論理一次子	p_i	T_i^t	T_i^f	T_i
1	eoln	0.05	5	5	5
2	ch=tab	0.1	3	3	3
3	ch=' '	0.3	3	3	3
4	flag	0.5	3	3	3

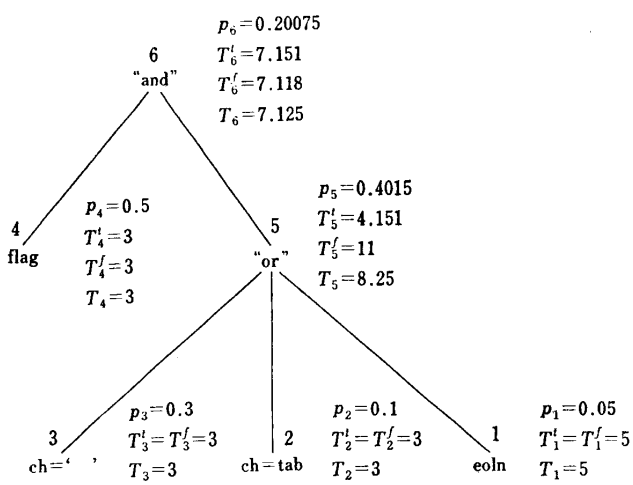
すると，その構文木の各節での p_i や T_i などの値は図3(a)のようになる．たとえば，節5 (or 節) については，次のようにして得られる．

$$p_5=1-\bar{p}_1\bar{p}_2\bar{p}_3=1-0.95\times0.9\times0.7=0.4015$$
$$T_5^f=T_1^f+T_2^f+T_3^f=5+3+3=11$$
$$T_5=T_1+\bar{p}_1T_2+\bar{p}_1\bar{p}_2T_3$$
$$=5+0.95\times3+0.95\times0.9\times3=10.415$$
$$T_5^t=(T_5-\bar{p}_5T_5^f)/p_5$$
$$=(10.415-0.5985\times11)/0.4015=9.543$$

* $T_i^t=T_i^f=T_i$ となるか否かについては文献5)に議論がある．



(a) もとの構文木とその確率，評価時間期待値
(a) Original syntax tree with probabilities and expected evaluation times attached.



(b) 評価順序入れ換え後の最適な構文木
(b) Optimal syntax tree after reordering.

図 3 「(eoln **or** ch=tab **or** ch='　') **and** flag」の構文木と評価順序の入れ換え
Fig. 3 An example reordering of the syntax tree for "(eoln **or** ch=tab **or** ch='　') **and** flag".

3. 評価時間期待値についての最小化定理

and, or 演算の可換律と結合律を利用して，部分論理式の評価順序を入れ換えることにより，飛び越し型評価の時間期待値を最小にすることができる．

この入れ換えが可能であるために，次の前提を仮定する．これらについては，文献 5) に議論がある．

前提 2 飛び越し型評価の順序を入れ換えてもプログラムの意味は変わらない．

前提 3 構文木中の and あるいは or 節の子供の p_i や T_i は，子供の間で評価順序を入れ換えても不変である（これは前提 1 よりやや強い前提である[5]）．

このとき，次の定理が成り立つ．

最小化定理

前提 2，3 が満たされているとき，論理式の構文木について次が成り立つ．

（1） n 個の子供 $(1, \cdots, n)$ を持つ and 節について，その節に対応する部分式の飛び越し型評価の時間期待値 T は，子供の評価順序を

$$\frac{T_{i_1}}{1-p_{i_1}} \leqq \frac{T_{i_2}}{1-p_{i_2}} \leqq \cdots \leqq \frac{T_{i_n}}{1-p_{i_n}}$$

が成り立つような順番

$i_1, i_2, \cdots, i_n$

に入れ換えることにより，最小になる．

（2） 同様に，or 節についての評価の時間期待値 T は，子供の評価順序を

$$\frac{T_{i_1}}{p_{i_1}} \leqq \frac{T_{i_2}}{p_{i_2}} \leqq \cdots \leqq \frac{T_{i_n}}{p_{i_n}}$$

が成り立つような順番

$i_1, i_2, \cdots, i_n$

に入れ換えることにより，最小になる．

(not 節については，子供が一人であるので入れ換えは無意味である．)

証明 証明を付録 2 に示した．なお，この証明には動的計画法の考え方を用いている．

上の定理の意味を，and 節について述べると，『値が偽になる確率が高く，評価時間が短い子供の節から先に評価すれば，早めに残りの子供の評価をスキップできることが多いので，評価時間を少なくできる』というものである．

最小化戦略

構文木上で，上の最小化定理に基づいた評価順序の入れ換えを，葉から根に向かって上向きに適用すれば，全体の評価の時間期待値を最小化できる．

例 2 例 1 の論理式（図 3 (a)）について，評価時間が最小になるように入れ換えを行おう．節 5 (or 節) の子供の最良の評価順序は，

$$\frac{T_3}{p_3}=\frac{3}{0.3}<\frac{T_2}{p_2}=\frac{3}{0.1}<\frac{T_1}{p_1}=\frac{5}{0.05}$$

なので，(節 3，節 2，節 1) の順とする．これにより T_5 は 8.25 に変化する．

次に，節 6 (and 節) の子供の最良の評価順序は，

$$\frac{T_4}{1-p_4}=\frac{3}{0.5}<\frac{T_5}{1-p_5}=\frac{8.25}{0.5985}$$

なので，(節 4，節 5) の順とする．このようにして評価順序を入れ換えたものを図 3 (b) に示す．全体として，この式の評価の時間期待値は，$T_5=11.620$ (図 3 (a)) から $T_6=7.125$ (図 3 (b)) に改善された．

4. おわりに

論理式を飛び越し型で評価するときの時間期待値を見積もる式と，評価の順序を入れ換えることにより評価時間を最小化するための定理を述べた．

本方式を実際に適用する場合は，評価時間や確率に対して推定値を用いればよい．たとえば，コンパイラでは，

変数の評価時間 ＜ 関数呼び出しの評価時間

とか，システム関数 endofline や endoffile が真になる確率は小さい，などとして適用することができる．

関連する研究として，Breitbart[3] や Gudes[4] の仕事がある．前者は，最適な評価順序の近似解を与えるものだが，not 演算を扱えない．後者は，実行時に論理一次子の評価を飛び越す方法を与えているが，評価順序は固定されている．両者とも，真偽確率や評価の時間期待値を考慮に入れていない．

本稿では述べられなかったが，前提 1-3 に関する議論，公式の詳しい導出，構文木の正準形，コンパイラへの適用については文献 5) を参照されたい．

謝辞 ご討論いただいた山本芳嗣氏 (筑波大学社会工学系) に感謝する．

参考文献

1) 中田育男：論理式のコンパイル方式，情報処理，Vol. 16, No. 3, pp. 186-194 (1975).
2) Logothetis, G. and Mishra, P.: Compiling Short-circuit Boolean Expressions in One Pass, *Softw. Pract. Exper.*, Vol. 11, pp. 1197-1214 (1981).
3) Breitbart, Y. and Reiter, A.: Algorithms for Fast Evaluation of Boolean Expressions, *Acta Inf.*, Vol. 4, pp. 107-116 (1975).
4) Gudes, E. and Reiter, A.: On Evaluating Boolean Expressions, *Softw. Pract. Exper.*, Vol. 3, pp. 345-350 (1973).
5) Sassa, M.: Time-Optimal Short-Circuit Evaluation of Boolean Expressions, Technical Report, ISE-TR-84-43, Inst. of Inf. Sciences and Electronics, Univ. of Tsukuba (1984).

付録 1 式(1)-(4)の公式の導出

- 式(1)，(2)はほぼ自明である．
- 式(4)は次のようにして導出できる．

T は，次の各場合の評価の時間期待値をそれが起こる確率で重みづけして合計したものである．

	評価時間		確率
(i) 第 1 子は必ず評価される	T_1	×	1
(ii) 第 2 子は，第 1 子が真のときのみ評価される	T_2	×	p_1
(iii) 第 3 子は，第 1，2 子が真のときのみ評価される	T_3	×	$p_1\times p_2$
…			
(iv) 第 n 子は，第 $1\cdots(n-1)$ 子が真のときのみ評価される	T_n	×	$p_1\cdots p_{n-1}$

(4)式は，これらの合計として得られる．

- 式(3)は，直接に，または式(4)と，

$$T^f=(T-pT^t)/\bar{p}$$

から導出できる．(詳しくは文献 5) 参照)

付録 2 最小化定理の証明

or は and と双対なので，and の場合のみ示す．

$$T(i_1, i_2, \cdots, i_n)=T_{i_1}+p_{i_1}T_{i_2}+\cdots+p_{i_1}\cdots p_{i_{n-1}}T_{i_n}$$
$$=\sum_{k=1}^{n}\left(\prod_{j=1}^{k-1}p_{i_j}\right)T_{i_k}$$

と置く．ここで，$(i_1, i_2, \cdots, i_n)$ は $(1, 2, \cdots, n)$ の置換だとする．

一般性を失うことなく

$$\frac{T_1}{1-p_1}\leqq\frac{T_2}{1-p_2}\leqq\cdots\leqq\frac{T_n}{1-p_n} \qquad \text{(a)}$$

と仮定してよい．このとき，いろいろな $T(\cdots)$ のうちで $T(1, 2, \cdots, n)$ が最小であることが言えればよい．

まず，(a)が成り立つとき，

$$T(1, \cdots, k-1, k, k+1, k+2, \cdots, n)$$
$$\leqq T(1, \cdots, k-1, k+1, k, k+2, \cdots, n) \qquad \text{(b)}$$

であることを示す．これは

左辺－右辺

$$=(p_1\cdots p_{k-1}T_k+p_1\cdots p_{k-1}p_kT_{k+1})$$
$$-(p_1\cdots p_{k-1}T_{k+1}+p_1\cdots p_{k-1}p_{k+1}T_k)$$
$$=p_1\cdots p_{k-1}\{T_k(1-p_{k+1})-T_{k+1}(1-p_k)\}$$
$$\leqq 0$$ ((a) と任意の i について $0<p_i<1$，より)

からわかる．よって(b)が言えた．

次に，もし $T(i_1, \cdots, i_k, i_{k+1}, \cdots, i_n)$ の中に，$i_k>i_{k+1}$ であるような逆順の対 (i_k, i_{k+1}) があるとすると，(b)

から

$$T(i_1, \cdots, i_k, i_{k+1}, i_{k+2}, \cdots, i_n)$$
$$\geq T(i_1, \cdots, i_{k+1}, i_k, i_{k+2}, \cdots, i_n)$$

がわかる．このステップを逆順の対がある限り繰り返すことにより，$T(1, 2, \cdots, n)$ が最小であることが言える．実は，$T(\cdots)$ は束 (lattice) を成し，$T(1, 2, \cdots, n)$ はこの束の最小元である．（証明終り）

（昭和60年6月5日受付）
（昭和60年10月17日採録）

佐々 政孝（第27巻第1号参照）
中田 育男（第27巻第1号参照）