数理解析研究所講究録
906 巻 1995 年 204-211



# On Breadth First Construction of OBDDs Representing Maximal Independent Sets

Kazuyoshi HAYASE
早瀬 千善
Department of Information Science University of Tokyo
東京大学理学部情報科学科
e-mail : hayase@is.s.u-tokyo.ac.jp

**Abstract**

Breadth first construction of OBDDs has been proposed as an output-size sensitive algorithm to cope with a defect of conventional method. In this paper, we investigate applications of this method to the problems of independent, dominating, and maximal independent sets of a graph. A maximal independent set corresponds to a prime implicant of the function of independent sets and also of dominating sets. We show the difficulty in constructing the OBDD of maximal independent sets by the breadth first construction. But OBDDs of the two other functions can be obtained in an output-size sensitive manner by the breadth first algorithm and the OBDD of maximal independent sets can be constructed with these two OBDDs, although not in a purely output-size sensitive sense.

## 1 Introduction

OBDDs (Ordered Binary Decision Diagrams) are useful representations for Boolean functions, which can be used for efficient manipulation [1]. Because of their good properties, OBDDs have been investigated and applied to various fields such as design and formal verification of digital systems, combinatorics, and so on. In combinatorial problems, generating and counting the number of all objects satisfying some condition are sometimes needed, and these computation can be efficiently done with OBDDs.

In almost all the conventional approaches using OBDDs, we have to represent problems in Boolean expressions and apply Boolean operations repeatedly on OBDDs. By these methods, some intermediate OBDDs may become much larger than the target OBDD and it is desired to construct OBDDs in some output-size sensitive manner. In [7], a framework is proposed to resolve this difficulty by constructing OBDDs directly in a top-down fashion from scratch with a breadth first algorithm. It has been shown that this algorithm can construct OBDDs with output-size sensitive complexity and be applied to some problems in graph theory.

The maximal independent sets problem is an interesting and important problem in graph theory and therefore several efficient algorithms for generating maximal independent sets have been proposed [8, 5, 4]. One of merits to represent maximal independent sets by an OBDD is that we can generate and count the number of all the maximal independent sets with it and its size is sometimes much smaller than the number of the sets.

Independent sets and dominating sets have some interesting relationship to maximal independent sets. The class of functions of independent sets is that of negative 2CNFs and that of dominating sets is a subclass of positive CNFs of polynomial size. And we can identify a maximal independent set with a prime implicant of the function representing independent sets or dominating ones respectively. In other words, the sets of the prime implicants of these two functions can be seen as a dual pair. Therefore generating maximal independent sets can be seen as generating prime implicants of these two functions [2, 6]. Furthermore, since a set is maximal independent if and only if it is independent and dominating, we can construct an OBDD of maximal independent sets from the two OBDDs of independent sets and dominating ones.

In this paper, we investigate the complexity to construct the OBDD representing maximal independent sets of a graph by the breadth first algorithm. The breadth first algorithm of [7] requires a good strategy of making subproblems and testing equivalence between two subproblems. However the test of equivalence between two problems or functions is intractable in general; for example, the problem of equivalence test between two Boolean expressions is known to be *co-NP* complete [3]. To support the equivalence test, we should investigate problems to choose a good representation. Unfortunately, it will be shown that the equivalence test between two subproblems of the problem of maximal independent sets is also *co-NP* complete. This fact states that the OBDD representing maximal independent sets can not be constructed efficiently by the algorithm as long as we encode subproblems in a simple and direct way. On the other hand, the two problems of independent sets and dominating sets have much simpler structures than that of maximal independent sets. We show how to apply the breadth first algorithm to these problems. Using the resulting OBDDs, as mentioned above, we can construct an OBDD representing maximal independent sets, although it may not always be done in output-size sensitive manner.

## 2 Preliminaries

In this section, we summarize basic notations of OBDDs and introduce the breadth first algorithm.

### 2.1 OBDDs[1]

An OBDD is a labelled directed acyclic graph representing a Boolean function. Non-terminal nodes of an OBDD are called variable nodes and terminal ones are constant ones. A variable node is labelled as a Boolean variable and a constant one a constant function (0 or 1). There is a total order on variables and the order of the labels of a path from a variable node to a constant node does not contradict it. In the following, this total order is assumed as $x_1 < x_2 < \ldots < x_n$ and we define the level of a variable node to be the subscript of its label. It is assumed that there is a root node and that its level is the smallest in the OBDD. The Boolean value labelled to a constant node $v$ is denoted as $value[v] \in \{0, 1\}$. The Boolean variable labelled to a variable node $v$ is denoted as $index[v] \in \{x_1, x_2, \ldots, x_n\}$. From each variable node $v$, there are two outgoing edges labelled as 0 and 1 and the two nodes directed by them are denoted as $edge(v, 0)$ and $edge(v, 1)$ respectively. For each node $v$, the Boolean function $F[v]$ represented by the subgraph rooted by $v$ is defined as the following equations

$$\begin{cases} F[v] = value[v] \in \{0,1\} & \text{if } v \text{ is a constant node} \\ F[v] = \overline{x_i} \cdot F[edge(v,0)] + x_i \cdot F[edge(v,1)] & \text{if } index[v] = x_i \end{cases} \tag{1}$$

It has been known that there exist two canonical forms for a Boolean function using OBDDs and this is one of the most delightful merits of OBDDs. These forms are the smallest in some senses, so we define two kinds of "unnecessary" nodes as follows.

**A redundant node** $v$ : $edge(v, 0) = edge(v, 1)$

**Equivalent nodes** $u, v$ : the two subgraphs rooted by $u$ and $v$ are isomorphic.

An OBDD without redundant nodes and equivalent nodes is called as an ROBDD (Reduced OBDD). An OBDD without equivalent nodes and in which every path from the root node to a terminal one consists of $n + 1$ nodes is called as a QOBDD (Quasi-reduced OBDD). As previously mentioned, it has been proved that for any Boolean function there are an ROBDD and a QOBDD representing it and that they are uniquely determined up to isomorphism. It is trivial that the QOBDD is not smaller than the ROBDD for any Boolean function, but the size (= number of nodes) of the QOBDD is of the order of the number of variables times the size of the ROBDD. We will consider mainly QOBDDs rather than ROBDDs for the rest of this paper.

### 2.2 The Breadth First Algorithm for Constructing QOBDDs

A breadth first algorithm for constructing a QOBDD based on (1) is proposed in [7], and Figure 1 shows an outline of it. This algorithm uses an idea that resembles the branching operation of the branch-and-bound method.

Each node in an OBDD represents a Boolean function, and each subproblem in the branch-and-bound can be seen as representing a Boolean function because there exists a one-to-one correspondence between the feasible solutions of the subproblem and the satisfying assignments of the Boolean function. (A subproblem $P|_{x_i:=b_i,x_j:=b_j,\dots}$ of a problem $P$ is defined by adding the constraints $x_i = b_i, x_j = b_j, \dots$ to $P$.) We can identify a subproblem with the corresponding function and consider a node $v$ in an OBDD as representing a subproblem $P[v]$. The level of $P[v]$ is defined to be the same as that of $v$.

An algorithm which constructs QOBDDs (for a specified problem) is called output-size sensitive if it has a time complexity of the order of a polynomial in the size of them and the number of variables. If we can find good strategies for representing subproblems and testing equivalence among them, we can construct QOBDDs in output-size sensitive manner using this algorithm.

**procedure** CONST($P$):<br>
  **var** $u, v$ : node;<br>
**begin**<br>
  create a root node $v$;<br>
  $P[v]$ :=INIT($P$);<br>
  **for** $i := 1$ **to** $n$ **do begin**<br>
    $table := \phi$;<br>
    **for_all** $u$ s.t. $index[u] = x_i$ **do**<br>
      **for** $b := 0$ **to** $1$ **do begin**<br>
        $P[u]|_{x_i:=b}$ := RESTRICT($P[u], x_i, b$);<br>
        **if** $\exists P[u'] \in table$ s.t. $(P[u]|_{x_i:=b} = P[u'])$ **then** $edge(u, b) := u'$;<br>
        **else begin**<br>
          create a variable node $u'$;<br>
          $edge(u, b) := u'$;<br>
          $table := table \cup \{P[u]|_{x_i:=b}\}$;<br>
        **end**;<br>
      **end**;<br>
  **end**;<br>
**end**;

Figure 1: The Breadth First Algorithm for Constructing the QOBDD for $P$

## 3 OBDDs Representing Maximal Independent Sets

In this section, we investigate the functions that represent the independent, maximal independent, and dominating sets of a graph. Then, we examine the breadth first algorithm for these three problems.

### 3.1 Maximal Independent Sets and Boolean Functions

Let $G = (V, E)$ be a simple undirected graph where $V = \{1, 2, \dots, n\}$ is the set of vertices and $E$ is the set of edges. $\Gamma(v)$ denotes the set of adjacent vertices of $v$.

A set $S \subseteq V$ is called independent if any pair of vertices in $S$ are not adjacent to each other. A set $S \subseteq V$ is called maximal independent if $S$ is independent and there is no independent set that includes properly $S$. A set $S \subseteq V$ is called dominating if any vertex is in $S$ or has an adjacent vertex in $S$. Let $IS(G), DS(G)$, and $MIS(G)$ denote the classes of all the subsets of vertices that are independent, maximal independent, or dominating, respectively. The next fact is well known in graph theory among these three classes.

**Fact 3.1** *A set $S \subseteq V$ is maximal independent iff $S$ is independent and dominating.*

Here we introduce how we treat a class of subsets by means of a Boolean function. Let $U$ be a finite set $\{1, 2, \ldots, |U|\}$. The characteristic vector of $S$ on $U$ ($S \subseteq$U) $\chi^U(S)$ is a $|U|$-dimensional Boolean vector in $\{0,1\}^{|U|}$ s.t.

$$\chi_i^U(S) = \begin{cases} 0 & (i \notin S) \\ 1 & (i \in S) \end{cases}$$

We identify an element of $U$ and the coordinate of $\mathbf{x}$ to which it corresponds, as long as there is no ambiguity. The Boolean function $f_{\mathcal{C}}$ representing the class $\mathcal{C}$ of subsets of $U$ is defined as:

$$f_{\mathcal{C}}(\mathbf{x}) = 1 \Leftrightarrow \exists S \in \mathcal{C} \text{ s.t. } \mathbf{x} = \chi^U(S)$$

## 3.2 A Boolean Expression for Maximal Independent Sets

We can form Boolean expressions of the functions $f_{IS(G)}, f_{DS(G)}$, and $f_{MIS(G)}$ as follows:

**Proposition 3.1**

$$f_{IS(G)} = \bigwedge_{(u,v)\in E} (\neg x_u \vee \neg x_v) \tag{2}$$

$$f_{DS(G)} = \bigwedge_{v\in V} (x_v \vee \bigvee_{u\in\Gamma(v)} x_u) \tag{3}$$

$$f_{MIS(G)} = f_{IS(G)} \wedge f_{DS(G)} \tag{4}$$

The next proposition can be proved where $PI(f)$ denotes the set of the prime implicants of $f$.

**Proposition 3.2** *$f_{IS(G)}$ is negative and $f_{DS(G)}$ is positive. strategies for There exist two one-to-one correspondences $R_{IS(G)}$ between $MIS(G)$ and $PI(f_{IS(G)})$, and $R_{DS(G)}$ between $MIS(G)$ and $PI(f_{DS(G)})$ as follows.*

$$X = \bigwedge_{v\in V-S} \neg x_v \;\Leftrightarrow\; (S, X) \in R_{IS(G)} \subseteq MIS(G) \times PI(f_{IS(G)})$$

$$X = \bigwedge_{v\in S} x_v \;\Leftrightarrow\; (S, X) \in R_{DS(G)} \subseteq MIS(G) \times PI(f_{DS(G)})$$

## 3.3 The QOBDD Representing Maximal Independent Sets

As stated in section 2.2, the breadth first algorithm requires good strategies for INIT and RESTRICT. We concentrate on how these can be done efficiently for the rest of this paper. The essential problem of the algorithm is that it requires equivalence test among subproblems. In the case of maximal independent sets problem, its complexity is intractable because the following problem EQ, which would have to be solved if trivial representation for subproblems is used in the breadth first algorithm, is *co-NP* complete.

**Definition 3.1** *Problem EQ: Given a graph $G$, an integer $j$ and two subsets of vertices $V_1$ and $V_2 \subseteq V$. Let $U = \{j+1, j+2, \ldots, n\} \subseteq V$. Two Boolean functions $F_1$ and $F_2$ on the characteristic vector of a subset $X \subseteq U$ are defined as:*

$$F_i(\chi^U(X)) = 1 \Leftrightarrow (V_i \cup X) \in MIS(G) \quad (i = 1, 2)$$

*Decide whether $F_1 = F_2$ or not.*

**Theorem 3.1** *The problem EQ is co-NP complete.*

**Proof:** It is easy to see that EQ $\in$ *co-NP*. We will show a polynomial time reduction from 3SAT like in [4]. An instance of 3SAT is given as a set of clauses $\{c_1, c_2, \ldots, c_m\}$ and a set of variables $\{x_1, x_2, \ldots, x_n\}$ where a clause $c_i = \{l_{i1}, l_{i2}, l_{i3}\}$. We will construct a graph $G$ indicated as follows. $G$ has a vertex for a clause, a vertex for a literal and an additional vertex $a$. The vertex $a$ is adjacent to all the others, and $x_i$ is adjacent to $\neg x_i$. For a clause $c_i$, there are three edges: $(c_i, l_{i1}), (c_i, l_{i2}), (c_i, l_{i3})$. Set $j = m + 1$, $V_1 = \{a, c_1\}$, and $V_2 = \emptyset$.

Let the order of the vertices be as follows:

$$a < c_1 < c_2 < \ldots < c_m < x_1 < \neg x_1 < x_2 < \neg x_2 < \ldots < x_n < \neg x_n$$

$F_1 = \perp$ is obvious by the definition of $MIS(G)$. We can consider that an truth assignment $A$ of 3SAT gives an independent set $X_A = \{(\neg)x_i \in A | 1 \leq i \leq n\}$ of $G$ and only such a set can be maximal under $V_2 = \emptyset$. If $F_1 = F_2$, on the other hand, we can conclude that any $A$ does not satisfy all the clauses because $X_A$ is not maximal and there must be a vertex $c_i$ that is not dominated by $X_A$. □

Note that we can not conclude that there is no possibility to construct the QOBDD representing $f_{MIS(G)}$ with an output-size sensitive complexity even with this fact and $NP \neq P$. We might be able to find another approach to construct the QOBDD or a good representation of subproblems and strategy to solve this hard problem.

## 3.4 QOBDDs for Independent Sets and Dominating Sets

Here we show two strategies for the problems of independent sets and dominating sets. As mentioned above, an OBDD that represents $f_{MIS(G)}$ can be constructed with a strategy that uses these two strategies although it may not always be a QOBDD because of its equivalent nodes.

In the following, $UNFIX[i] = \{x_j | j \geq i\}$ denotes the set of the variables that are not fixed in subproblems of level $i$. If a subproblem is found to be inconsistency (0) or tautology (1), it is represented as $\perp$ or $\top$, respectively, and exceptionally processed, but that will not be mentioned explicitly.

### 3.4.1 The QOBDD Representing Independent Sets

Here we show a strategy for the independent set problem. If we fix a variable $x_i$ to be 1 in a subproblem $P_a$ where some vertex $v$ in $\Gamma(x_i)$ has been fixed to be 1, the subproblem $P_a|_{x_i:=1}$ can be decided to be $\perp$. So a subproblem $P_a$ can be represented by the set that consists of unfixed variables that have adjacent vertices that have been fixed to be 1. Figure 2 shows the strategies for $f_{IS(G)}$ based on this consideration.

**Theorem 3.2** *The OBDD constructed by the strategies indicated in figure 2 is the QOBDD representing $f_{IS(G)}$.*

**function** INIT($G$): problem;
  **return** $\emptyset$;

**function** RESTRICT($P[u], x_i, b$) : problem;
**begin**
  **if** $b = 0$ **then return** $P[u] - \{x_i\}$;
  **else if** $x_i \in P[u]$ **then return** $\perp$;
    **else return** $P[u] \cup (\Gamma(x_i) \cap \mathit{UNFIX}[i+1])$;
**end**;

Figure 2: The Strategies for Independent Sets

### 3.4.2 The QOBDD Representing Dominating Sets

In this section, we show a strategy for the dominating sets problem. We investigate difference between the Boolean expressions of independent sets and dominating sets. Equation (2) gives a positive 2CNF of $f_{IS(G)}$ and the strategy described in the previous section can be recognized to compute a canonical form of each subproblem in a sense. So we expect that we could get a similar strategy for $f_{DS(G)}$ by computing canonical forms. Here we describe a strategy for $f_{DS(G)}$ only, but it can be applied to a little wider class of expressions.

We introduce an equivalence relation $\equiv_i$ on $V$ based on the set of unfixed adjacent vertices of each node. Define a partial order $\prec_i$ on the quotient set $V/\!\equiv_i$ of the relation $\equiv_i$. Note that we define only a partial order on $V/\!\equiv_i$, but we can define a total order on it and we use it to indicate subscript of a characteristic vector on $V/\!\equiv_i$.

**Definition 3.2**

$$\begin{array}{rcll} \Pi_i(x_p) & := & \mathit{UNFIX}[i] \cap (\Gamma(x_p) \cup \{x_p\}) & (\forall x_p \in V) \\ x_p \equiv_i x_q & \Leftrightarrow & \Pi_i(x_p) = \Pi_i(x_q) & (\forall x_p, x_q \in V) \\ \Pi_i(S) & := & \Pi_i(x) & (\forall x \in V,\ S \in V/\!\equiv_i \text{ s.t. } x \in S) \\ S \prec_i T & \Leftrightarrow & \Pi_i(S) \subset \Pi_i(T) & (\forall S, T \in V/\!\equiv_i \text{ s.t. } S \neq T) \end{array}$$

The strategy (Figure 3) represents a subproblem of level $i$ with a characteristic vector on $V/\!\equiv_i$. We prove that the strategy constructs the QOBDD representing $f_{DS(G)}$.

**Proposition 3.3** *$V/\!\equiv_i$ is a refinement of $V/\!\equiv_j$ if $i \leq j$.*

**Lemma 3.1** *For an equivalence class $S$ and a subproblem $P$ of level $i$ that is not decided to be $\perp$, the following is true where $POS(P)$ denotes the set of variables that are fixed to be 1 in the subproblem $P$.*

$$P_S = 1 \quad \Leftrightarrow \quad \text{(a)“}\exists x \in S,\quad \Pi_1(x) \cap POS(P) = \emptyset\text{”} \quad \textit{and} \quad \text{(b)“}\forall T \prec_i S,\quad P_T = 0\text{”}$$

**Lemma 3.2** *Assume that two subproblems $P$ and $Q$ in level $i$ are given and $P \not\leq Q$, i.e. there exists an equivalence class $S$ in level $i$ such that $P_S = 1$ and $Q_S = 0$, then the following holds:*

1. *if there exists $T$ such that $T \prec_i S$ and $Q_T = 1$, there exists a feasible solution in $P$ but is not in $Q$. That is the next equation holds:*

$$P(\xi) = 1 \textit{ and } Q(\xi) = 0 \quad \textit{for } \xi \in \{0,1\}^{n-i+1} \textit{ s.t. } \xi_k = \begin{cases} 0 & x_{k+i-1} \in \Pi_i(T) \\ 1 & \textit{otherwise} \end{cases}$$

```
function INIT(G): problem;
  var init: problem;
begin
  init := 1^{|V/≡_1|};
  return CANON(init, 1);
end;

function RESTRICT(P[u], x_i, b) : problem
  var Q: problem
begin
  if b = 0 then begin
    if {x_i} ∈ V/≡_i and P[u]_{x_i} = 1 then return ⊥;
    for_all S ∈ V/≡_i do
      Q_S := (∃T ∈ V/≡_{i-1} s.t. Π_i(S) = Π_{i-1}(T) − {x_i} and P[u]_T = 1);
  end;
  else for_all S ∈ V/≡_i do
    Q_S := (∃T ∈ V/≡_{i-1} s.t. Π_i(S) = Π_{i-1}(T) and P[u]_T = 1);
  return CANON(Q, i + 1);
end;

function CANON(P, i) : problem;
begin
  for j := |V| − i + 1 downto 2 do
    for_all S ∈ V/≡_i s.t. |Π_i(S)| = j and P_S = 1 do
      if ∃T ∈ V/≡_i s.t. T ≺_i S and P_T = 1 then P_S = 0;
  return P;
end;
```

Figure 3: The Strategies for Dominating Sets

2. *otherwise, there exists a feasible solution of $Q$ that is not feasible in $P$, that is,*

$$P(\xi) = 0 \text{ and } Q(\xi) = 1 \quad \text{for } \xi \in \{0,1\}^{n-i+1} \text{ s.t. } \xi_k = \begin{cases} 0 & x_{k+i-1} \in \Pi_i(S) \\ 1 & \text{otherwise} \end{cases}$$

**Theorem 3.3** *The OBDD constructed by the strategies indicated in Figure 3 is the QOBDD representing $f_{DS(G)}$.*

**Proof:** A subproblem $P$ is decided to be $\perp$ iff $P = Q|_{x_i:=0}$ and $Q_{\{x_i\}} = 1$. Then any $x$ in $\Pi_1(x_i)$ is fixed to be 0 in $P$ by lemma 3.1. It proves that $P$ has no feasible solution and that the OBDD represents $f_{DS(G)}$. Furthermore we can confirm the OBDD is a QOBDD by lemma 3.2. □

We can see that a characteristic vector $P$ indicates which clauses of equation (3) are left in the subproblem. Therefore we can also devise a similar strategy for positive CNFs because they have unique smallest forms and we can compute them as canonical forms.

Furthermore, as noticed above, we can obtain an OBDD representing $MIS(G)$ using the strategies for $IS(G)$ and $DS(G)$ in top-down fashion, although this OBDD may not be a QOBDD because we could make equivalent nodes and therefore it may not be in a purely output-size sensitive manner.

## 4 Conclusion

We have investigated an application of the breadth first algorithm for constructing QOBDDs to the problems of the independent sets, dominating sets, and maximal independent sets of a graph. It has been shown that we have to solve a hard problem to construct the QOBDD of the maximal independent sets in breadth first manner, although we have shown strategies to apply the algorithm to the problems of independent sets and dominating sets using their monotonicity.

As future works, we will study possibility to construct the QOBDD of maximal independent sets in output-size sensitive manner and to apply the breadth first algorithm when the input is limited to a certain class. We also investigate relationships between the size of the QOBDD of maximal independent sets and that of independent or dominating sets.

## Acknowledgement

The author would like to thank members of Imai laboratory for their helpful discussions and comments to this work.

## References

[1] R. E. Bryant. Graph-based algorithms for Boolean function manipulation. *IEEE Trans. on Computers*, C-35(8):677–691, 1986.

[2] O. Coudert and J. C. Madre. Implicit and incremental computation of primes and essential primes of Boolean functions. In *Proc. 29th ACM/IEEE DAC*, pages 36–39, 1992.

[3] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness*. W.H.Freeman, 1979.

[4] D. S. Johnson, M. Yannakakis, and H. Papadimitriou. On generating all maximal independent sets. *Information Processing Letters*, 27:119–123, 1988.

[5] E. L. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan. Generating all maximal independent sets: *NP*-hardness and polynomial-time algorithms. *SIAM J. Comput.*, 9(3):558–565, 1980.

[6] S. Minato. Fast generation of prime-irredundant covers from binary decision diagrams. *IEICE Trans. Fundamentals*, E76-A:967–973, 1993.

[7] S. Tani and H. Imai. A reordering operation for an ordered binary decision diagram and an extended framework for combinatorics of graphs. In *ISAAC'94, Lecture Notes in Computer Science*, volume 834, pages 575–583, 1994.

[8] S. Tsukiyama, M. Ide, H. Ariyoshi, and I. Shirakawa. A new algorithm for generating all the maximal independent sets. *SIAM J. Comput.*, 6(3):505–517, 1977.