ECE 510 OCE
BDDs and Their Applications

Lecture 7.
(1) BDDs for Symmetric Functions
(2) Constraint Satisfaction Problems

April 18, 2000
Alan Mishchenko

Overview

- BDDs for symmetric functions
- Characteristic functions of sets and sets of sets
- Building BDD for Tuples "k out of n"
- Constraint satisfaction problems
- Example: N-queen Problem
- Binate Covering Problem
- Solution of BCP using Shortest Path on BDDs

BDDs for Symmetric Functions

- The value of a symmetric function depends only on the number of 1's in the input vector
- For n-var functions, there are n+1 possible numbers of 1's and so 2^{n+1} different functions
- Cofactors of symmetric functions are symmetric functions
- If the function is cofactored k times, there are no more than n+1-k different cofactors
- The BDD size is limited by n*(n+1-k) ~ $O(n^2)$

Characteristic Function of a Set

- Function F: $B^n \rightarrow B$, B = {0,1}, defines a subset of minterms of B^n, on which it is 1.
- Given a binary encoding of a set of elements, characteristic function of a *subset* of this set is a boolean function, which is 1 for minterms encoding the subset and 0 for other minterms.

April 18, 2000 ECE 510 OCE: BDDs and Their Applications 4

Example

- Problem: Given the set {p_0, p_1, p_2, p_3, p_4, p_5 } and its encoding:

  $p_0 - \bar{x}_2\bar{x}_1\bar{x}_0$ $p_2 - \bar{x}_2 x_1\bar{x}_0$ $p_4 - x_2\bar{x}_1\bar{x}_0$

  $p_1 - \bar{x}_2\bar{x}_1 x_0$ $p_3 - \bar{x}_2 x_1 x_0$ $p_5 - x_2\bar{x}_1 x_0$

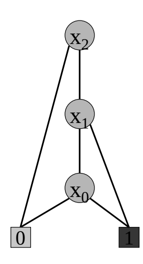
  find characteristic function of subset {p_0, p_2, p_3} and represent the subset using BDD
- Solution: Define a function over the encoding variables (x_0,x_1,x_2) such that it is equal to 1 for minterms representing subset {p_0, p_2, p_3}.

$$\varphi_{\{p0,\ p2,\ p3\}}(x_0,x_1,x_2) = \bar{x}_2\bar{x}_1\bar{x}_0 + \bar{x}_2 x_1\bar{x}_0 + \bar{x}_2 x_1 x_0$$

April 18, 2000 ECE 510 OCE: BDDs and Their Applications 5

BDD Representation of the Characteristic Function

$$\mathbf{F} = \bar{x}_2\bar{x}_1\bar{x}_0$$
$$+ \ \bar{x}_2 x_1\bar{x}_0$$
$$+ \ \bar{x}_2 x_1 x_0$$



April 18, 2000 ECE 510 OCE: BDDs and Their Applications 6

Char Function of Set of Subsets

- Let the set S = {p_1, p_2,..., p_n } contain n elements. Assuming positional notation, the elements of the set are encoded using one variable per element
- Example: Suppose n = 5 and the subset is {p_1, p_3, p_4 }. This subset can be represented by the characteristic function $\varphi_{\{p1, p3, p4\}} = x_1x_2'x_3x_4x_5'$
- Example: For the same set with n = 5, the set of all subsets containing exactly one element is represented by the characteristic function $\varphi_{S1} = x_1x_2'x_3'x_4'x_5' + x_1'x_2x_3'x_4'x_5' + x_1'x_2'x_3x_4'x_5' + x_1'x_2'x_3'x_4x_5' + x_1'x_2'x_3'x_4'x_5$

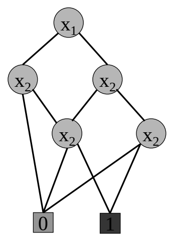
Char Function of Tuples “k out of n”

- Problem: Given the set S of n elements, build a BDD for the characteristic function of the set of subjects containing exactly k out n elements of set S
- The brute force approach results in creating the sum of n!/k!/(n-k)! cubes, each of which represents a characteristic function of a subset
- The intelligent approach uses a recursive BDD procedure to build the char function

Building Char Function of Tuple Set

```
bdd Tuples( int k, int n)
{
    if ( k < 0 || n < k )        return bddfalse;
    if ( k == 0 && n == 0 )  return bddtrue;
    // check cache for results
    bdd F0 = Tuples( k,   n-1 );
    bdd F1 = Tuples( k-1, n-1 );
    bdd Res = bdd_ite( bdd_ithvar(n), F1, F0 );
    // insert into cache
    return Res;
}
```

BDD for Tuple(2, 3)



Constraint Satisfaction Problems

- The parameters of SAT problems are represented using binary variables and the requirements for a solution are expressed as boolean formulas over the binary variables
- When all these formulas are multiplied, we get a BDD whose path lead to constant 1 for those assignments that satisfy the problem
- A SAT problem is solved by analyzing this BDD (for example, by counting the number of paths, or finding the shortest path)

N-Queen Problem

- Is it possible to place n queens on a n x n chess board so that no queen can be captured by another queen? If yes, how many different placements are possible?
- Let us encode the presence/absence of a queen in each cell by a binary variable (altogether we need n x n binary variables)
- Now, the requirements of not capturing can be expressed in terms of these varables

Encoding the Constraints

- There is no more than one queen in each row, each column, and each diagonal

  $X_{ij} => \prod(X_{ik}')$, $1 \le k \le n$, $k \ne j$; $X_{ij} => \prod(X_{mj}')$, $1 \le m \le n$, $m \ne l$, etc.
- There is a queen in each row

  $X_{i1} + X_{i2} + \ldots + X_{iN}$
- After multiplying the constraint, we get the BDD representing all solutions
- Example: for 8 queens, there 2450 nodes, and 92 solutions
- Reference: S. Minato. "Calculation of Unate Cube Set Algebra Using ZBDDs", DAC'94. http://www.sigda.acm.org/Archives/ProceedingArchives/

Binate Covering Problem (BCP)

- Given a covering table and the objective function $\min[\Sigma_j(w_j x_j)]$, BCP is formulated as follows
- Find a subset S of columns of minimum cost (according to the objective function) such that for every row f_j

  either (1) $\exists_j$: $(f_{ij}=1)$ & $(F_j \in S)$

  or (2) $\exists_j$: $(f_{ij}=0)$ & $(F_j \notin S)$
- This problem arises in incompletely-specified FSM state minimization, technology mapping, decomposition of functions with DCs, BDD minimization, etc.
- BCP is reduced to the problem of finding the shortest path on the BDD representing the product of all the constraints

Example of BCP

x_1	x_2	x_3	x_4	x_5	x_6
1	-	1	-	-	-
-	1	-	1	-	1
-	-	0	1	1	-
-	-	-	-	-	0
0	-	-	-	-	0
-	-	1	0	1	-

- Solution 1: $\{x_1,x_2\}$
- Solution 2: $\{x_3,x_4\}$

BCP in State Minimization of ISFSMs

- Covering constraints

$(c_1+c_{11})(c_1+c_2+c_5)(c_2+c_3+c_5+c_6+c_7+c_8)$
$(c_1+c_2+c_4+c_6+c_{10})(c_1+c_4)(c_3+c_7+c_9+c_{12})$
$(c_3+c_8+c_9+c_{11})(c_4+c_{10})$

- Closure constraints

$(c_2'+c_1)(c_2'+c_{11})(c_2'+c_1+c_4)(c_3'+c_2+c_6)$
$(c_3'+c_4)(c_4'+c_1)(c_4'+c_1)(c_6'+c_{11})(c_6'+c_1+c_4)$
$(c_7'+c_2+c_6)(c_8'+c_2+c_6)(c_8'+c_3+c_9)(c_9'+c_4)$

- Solution

$c_1 = c_4 = c_5 = c_9 = 1$