IIT KHARAGPUR

NPTEL ONLINE CERTIFICATION COURSES

NPTEL

Lecture 27: Binary Decision Diagrams (Part 1)

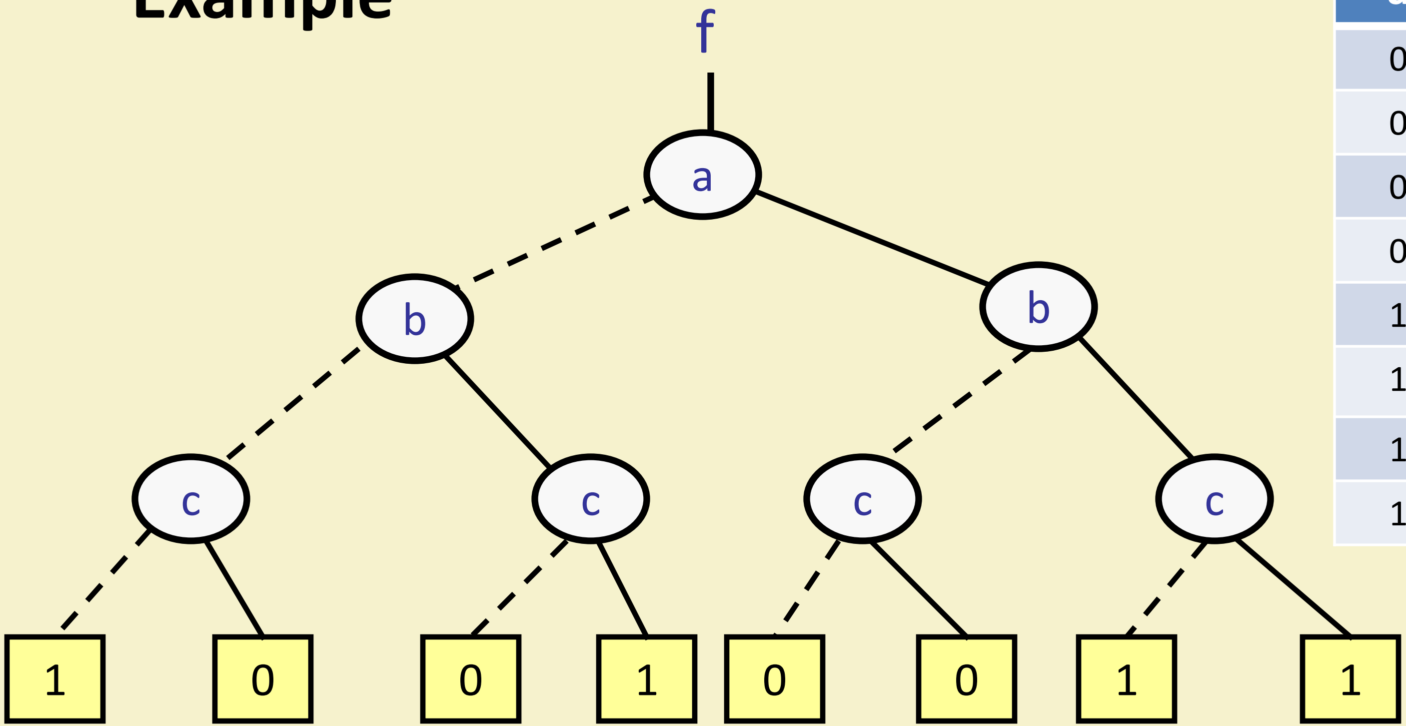
PROF. INDRANIL SENGUPTA

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Binary Decision Diagrams (BDD)

- A data structure used to represent a Boolean function.
- Represented as a rooted, directed, acyclic graph, which consists of several *decision nodes* and two *terminal nodes, called* *0-terminal* and *1-terminal*.
  - Each decision node is labeled by a Boolean variable and has two child nodes called *low child*, and *high child*.
  - The edge from a node to a low (high) child represents an assignment of the variable to 0 (1).
- A BDD is said to be *ordered* if different variables appear in the same order on all paths from the root.

Example



a	b	c	f
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

- Construction of a BDD is based on the *Shannon expansion* of a function.

Shannon Expansion

- Given a Boolean function $f\,(x_1, x_2, \ldots, x_i, \ldots, x_n)$
- Positive cofactor

  $$f_i^1 = f\,(x_1, x_2, \ldots, 1, \ldots, x_n)$$

- Negative cofactor

  $$f_i^0 = f\,(x_1, x_2, \ldots, 0, \ldots, x_n)$$

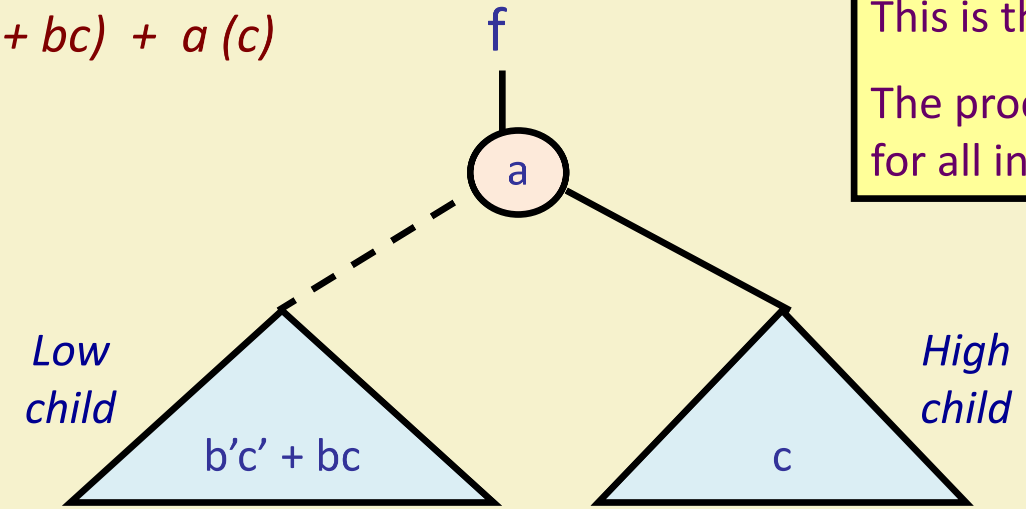
- Shannon's expansion theorem states that

$$f = x_i' f_i^0 + x_i f_i^1$$

$$f = (x_i + f_i^0)(x_i' + f_i^1)$$

How to construct BDD?

$$f = ac + bc + a'b'c$$
$$= a'(b'c' + bc) + a(c + bc)$$
$$= a'(b'c' + bc) + a(c)$$



This is the first step.

The process is continued for all input variables.

$f = ac + bc + a'b'c'$

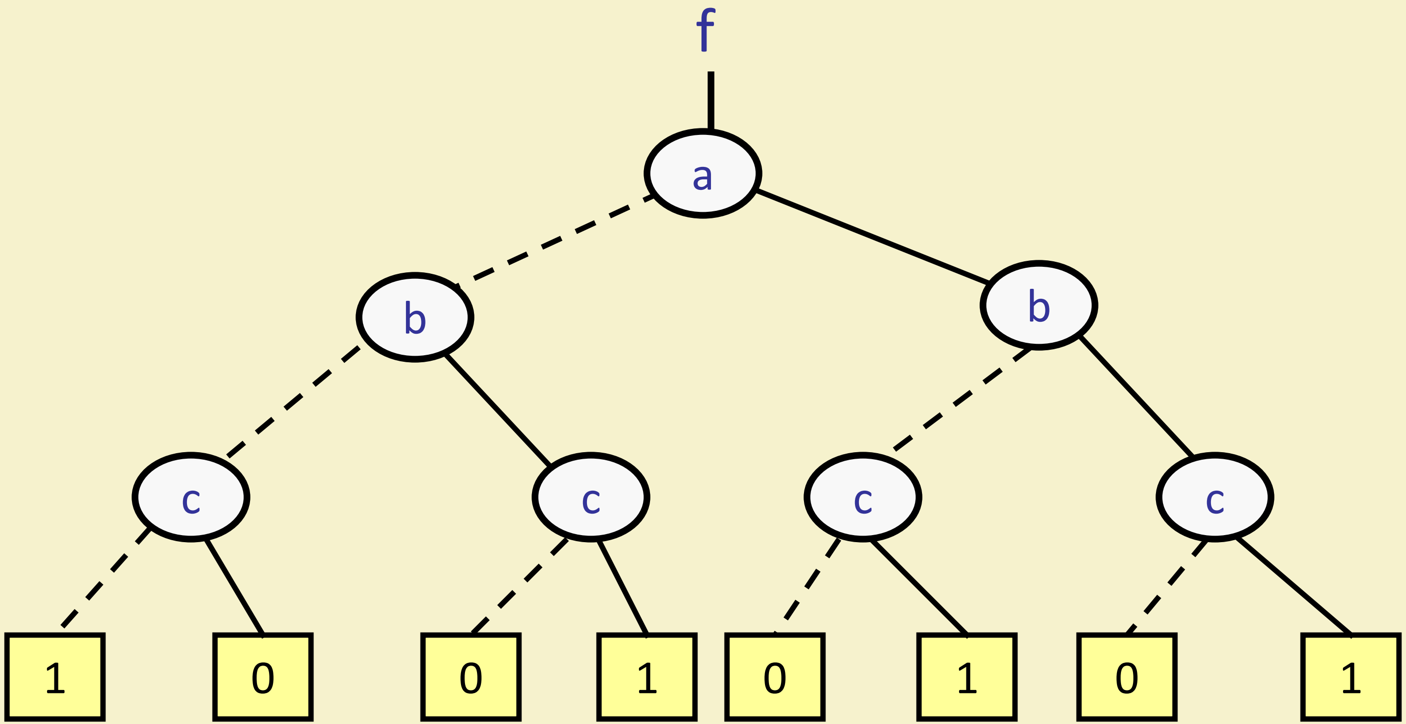
$= a'(b'c' + bc) + a(c + bc)$ *Expand by a*

$= a'(b'c' + bc) + a(c)$

$b'(c') + b(c)$ $b'(c) + b(c)$ *Expand by b*

$c'(1) + c(0)$ $c'(0) + c(1)$ $c'(0) + c(1)$ $c'(0) + c(1)$ *Expand by c*

Variable ordering: a, b, c


f
a
b
b
c
c
c
c
1
0
0
1
0
1
0
1

Another Example: $f = a'bc + b'c' + ac'$

Variable Ordering (OBDD)

- The size of a BDD is determined both by the function being represented and the chosen ordering of the variables.
  - For some functions, the size of a BDD may vary between a *linear* to an *exponential* range depending upon the ordering of the variables.
- An example:

$$f(x_1,\ldots,x_{2n}) = x_1x_2 + x_3x_4 + \ldots + x_{2n-1}x_{2n}$$

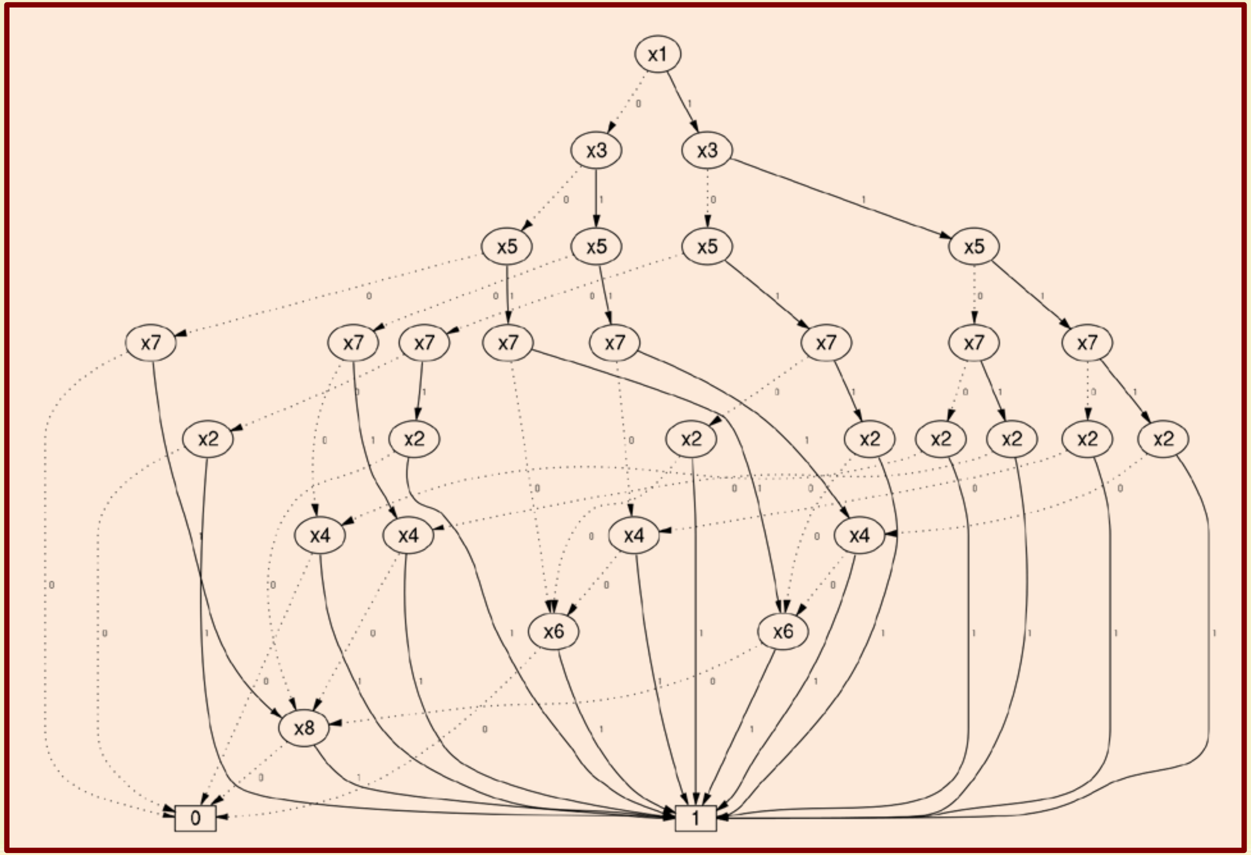
Variable ordering: $x_1 < x_3 < \ldots < x_{2n-1} < x_2 < x_4 < \ldots < x_{2n}$

BDD requires 2^{n+1} *nodes* to represent the function.

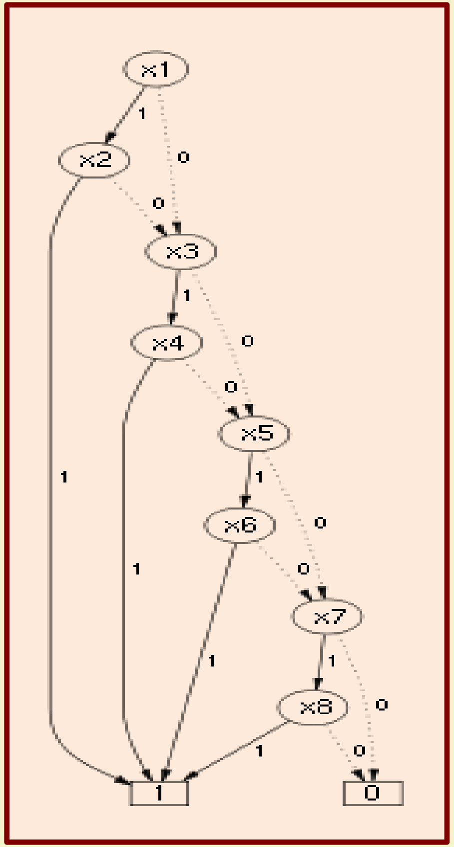
Variable ordering: $x_1 < x_2 < x_3 < x_4 < \ldots < x_{2n-1} < x_{2n}$

BDD requires $2n$ *nodes* to represent the function.

BDD for the function $f(x_1, \ldots, x_8) = x_1x_2 + x_3x_4 + x_5x_6 + x_7x_8$ using bad variable ordering



Same function using good variable ordering



- Important point to note:
  - It is essential to find a good variable ordering when using the OBDD data structure in practice.
  - The problem of finding the best variable ordering is NP-hard.
  - Several heuristics for variable ordering have been proposed.

END OF LECTURE 27