How to build a Reduced Ordered BDD

• Done by recursion on the structure of the formula
• Order propositional variables
• For each variable, create a three node tree corresponding to (if Var then true else false)
• For formula not P, build ROBDD from ROBDD for P by flipping values in the leaves

How to build a Reduced Ordered BDD

• For P <op> Q
  – Fill in each branch of BDDs for P and Q with all nodes appearing in either BDD, not already on branch
  – For each ordered sequence of values (done in order of variables, greatest to least, false before true) build the branch in partial BDD for P <op>
  – Value in end leaf is v1 <op> v2 where v1 is value at end of branch in BDD for P, v2 for branch in Q

Example

• Find ROBDD for (A ∧ B) ∨ (not C)
(A \land B) \lor (\neg C) \quad \text{Variables: C > B > A}
(A \land B) \lor (\neg C) \quad \text{Variables: } C > B > A
(A ∧ B) ∨ (not C)  Variables: C > B > A

(A ∧ B) ∨ (not C)  Variables: C > B > A

(A ∧ B) ∨ (not C)  Variables: C > B > A

(A ∧ B) ∨ (not C)  Variables: C > B > A

(A ∧ B) ∨ (not C)  Variables: C > B > A

(A ∧ B) ∨ (not C)  Variables: C > B > A
Example
• Find ROBDD for \((A \land B) \lor (\neg C)\) \lor \neg (A \land B)\)
((A ∧ B) ∨ (not C)) ∨ (not A ∧ B)  Variables: C > B > A
\(((A \land B) \lor (\neg C)) \lor (\neg (A \land B))\)

Variables: C > B > A
\((A \land B) \lor \neg C) \lor \neg(A \land B)\)  
Variables: C > B > A
Uses of ROBDDs

- Reduced Order BDD for a proposition is unique for a fixed variable ordering
- Proposition is valid iff its ROBDD is just True
- Proposition is satisfiable its ROBDD is not just False
- Can check if a given valuation satisfies proposition in time linear to number of variables by walking the corresponding branch