CS477 Formal Software Development Methods

Elsa L Gunter
2112 SC, UIUC
egunter@illinois.edu
http://courses.engr.illinois.edu/cs477

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

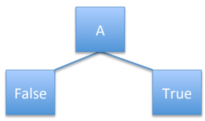
How to build a Reduced Ordered BDD

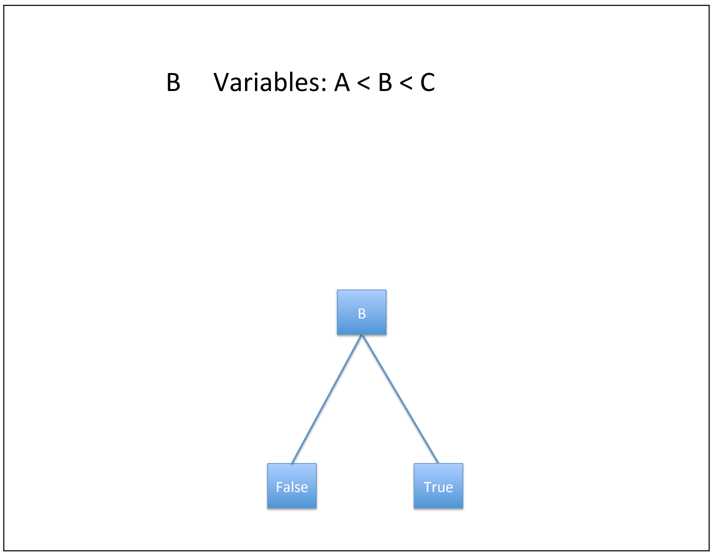
- When you add a new node
  - If the outedges point to same node, move all edges pointing to it to point to what it points to, and remove / ghost it.
  - If there exists another node with the same label whose edges point the same nodes, remove the new node and move all edges pointing to it to the other node

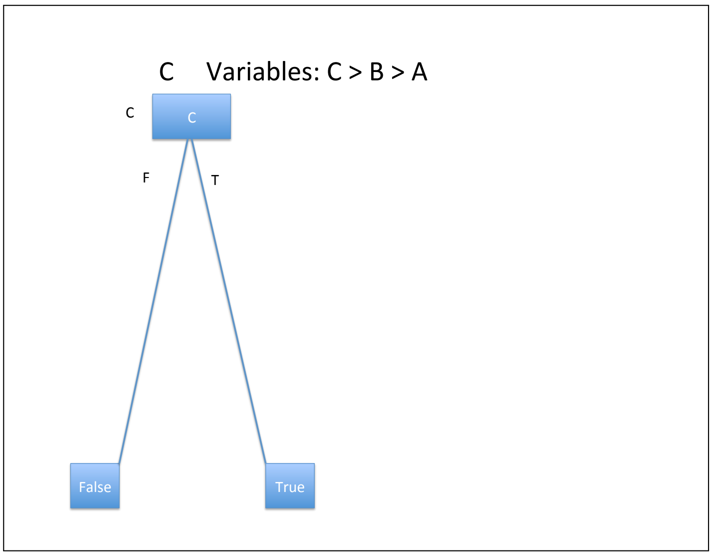
Example

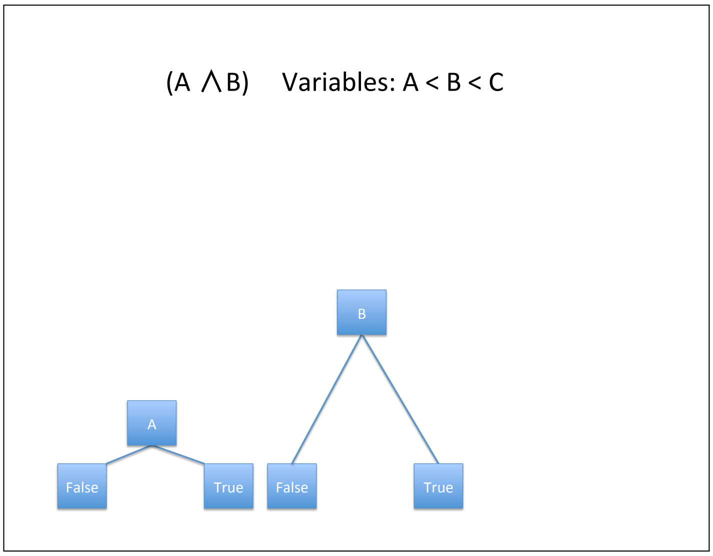
- Find ROBDD for $(A \wedge B) \vee (\text{not } C)$

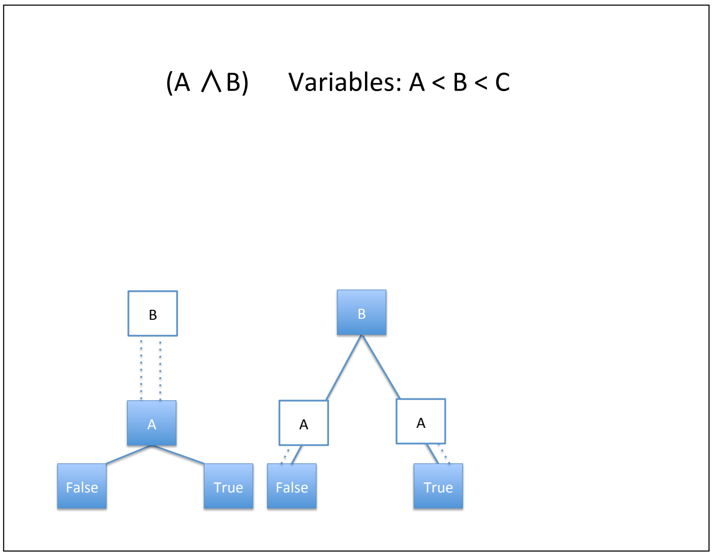
A Variables: A < B < C

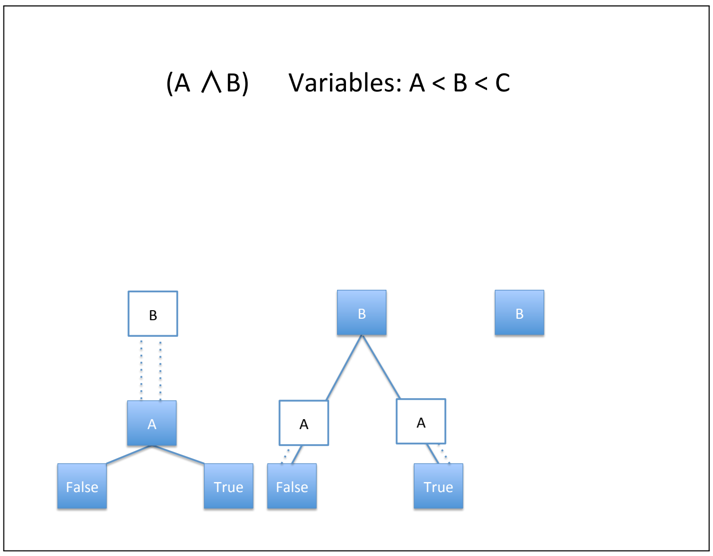


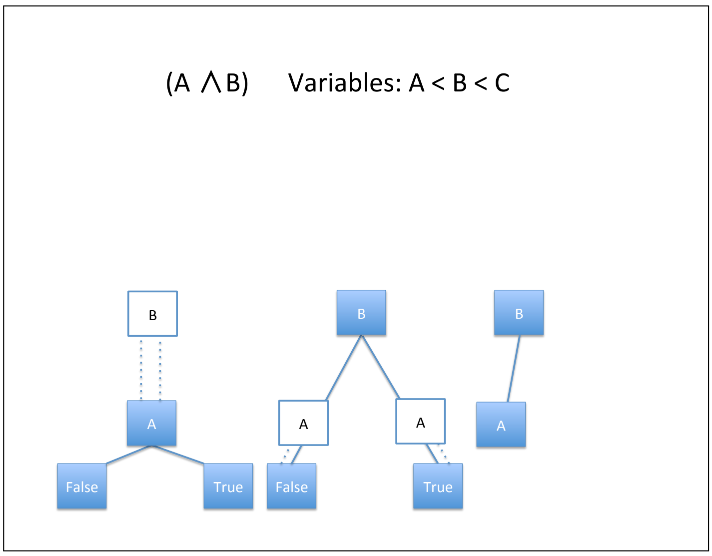

B Variables: A < B < C
B
False
True


C Variables: C > B > A
C
C
F
T
False
True

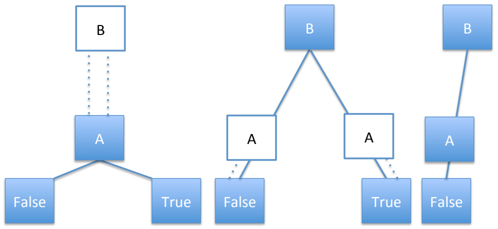

(A ∧B) Variables: A < B < C
A
B
False
True
False
True


(A ∧B) Variables: A < B < C
B
B
A
A
A
False
True
False
True

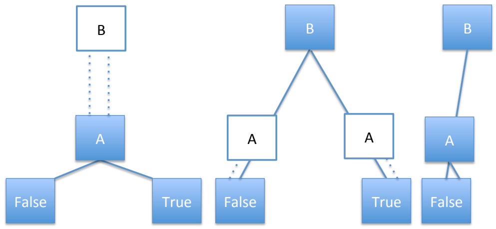

(A ∧B) Variables: A < B < C
B
B
B
A
A
A
False
True
False
True


(A ∧B) Variables: A < B < C
B
B
B
A
A
A
A
False
True
False
True

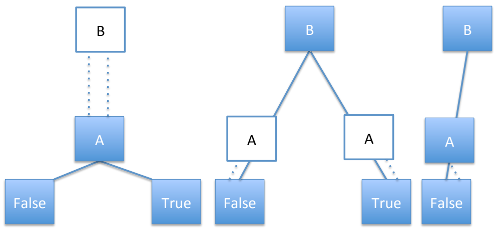
(A ∧B) Variables: A < B < C



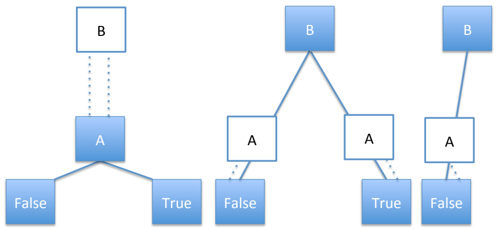
(A ∧B) Variables: A < B < C



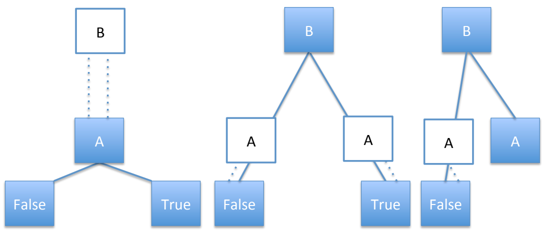
(A ∧B) Variables: A < B < C



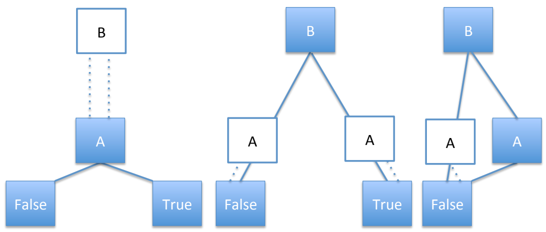
(A ∧B) Variables: A < B < C

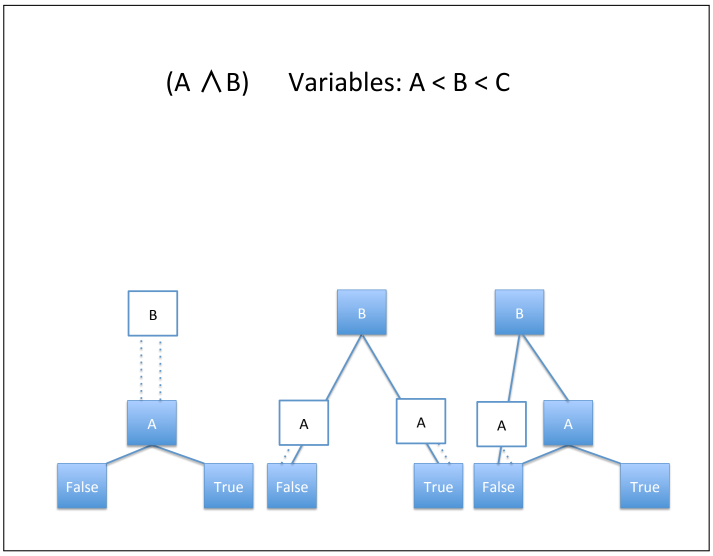


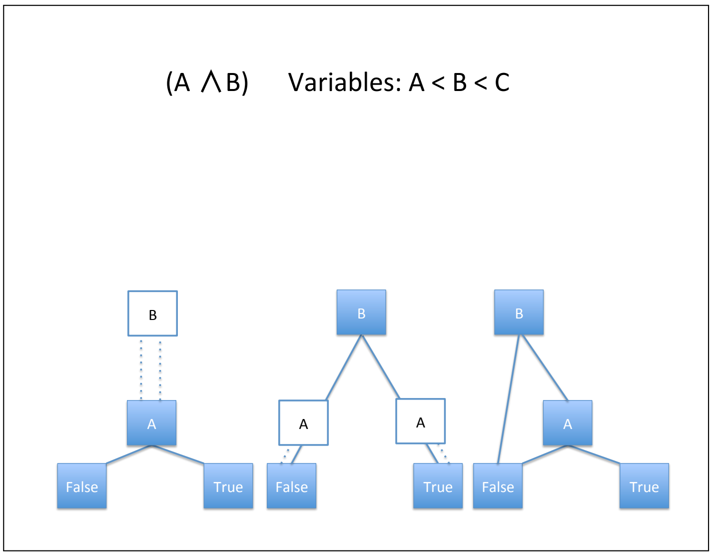
(A ∧B) Variables: A < B < C

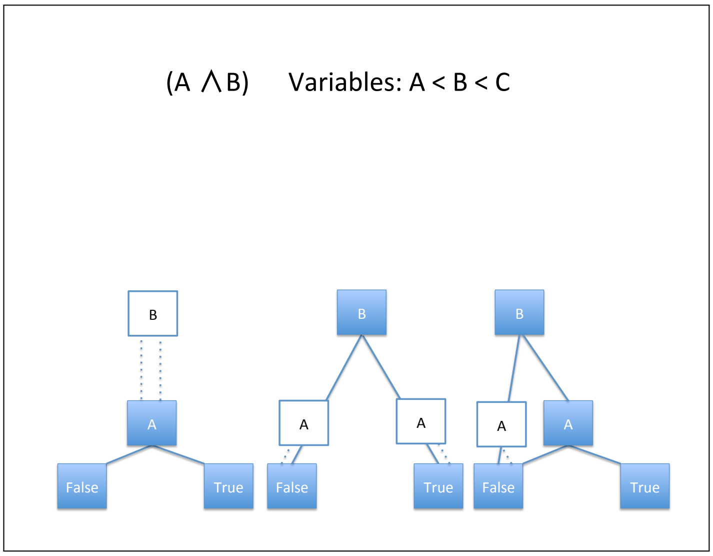


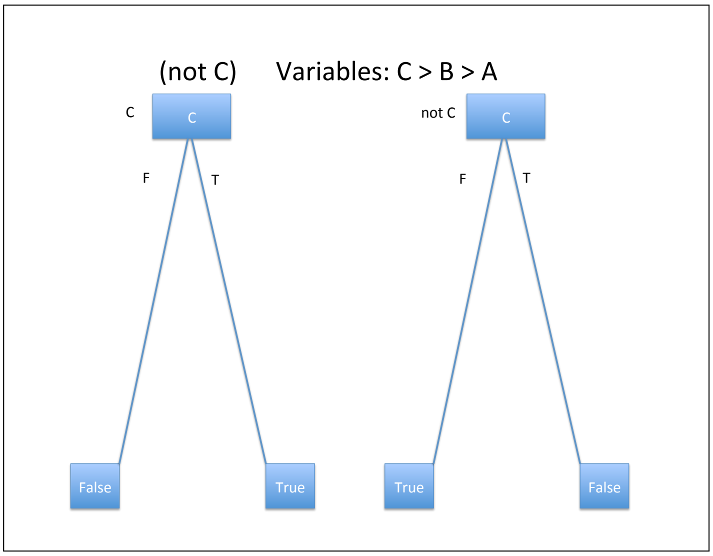
(A ∧B) Variables: A < B < C

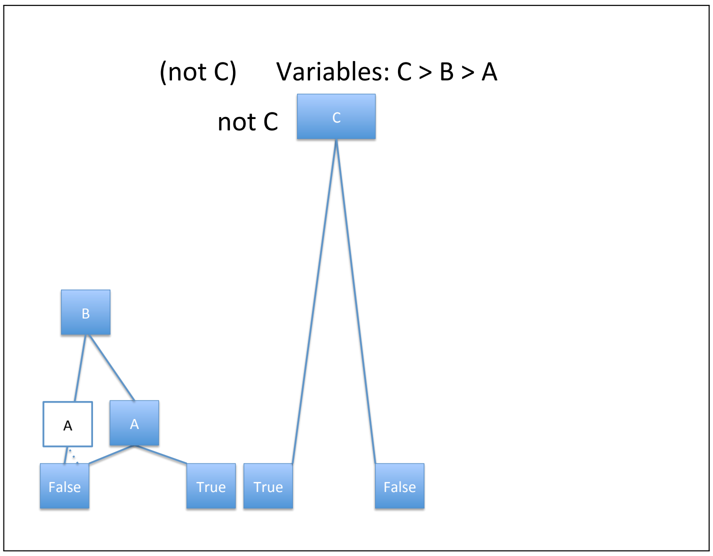


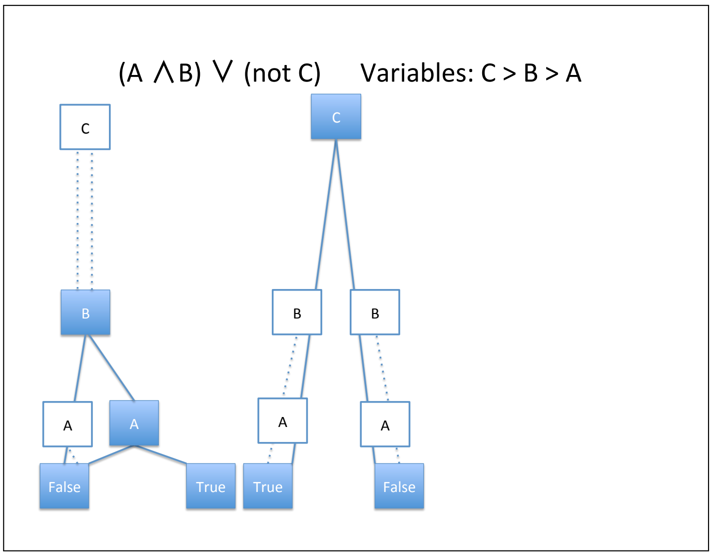

(A ∧B) Variables: A < B < C
B
A
False
True
B
A
A
False
True
B
A
A
False
True

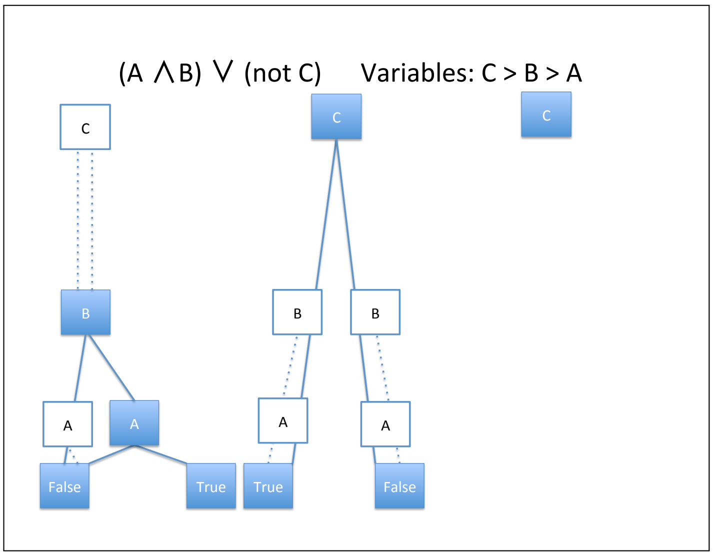

(A ∧B) Variables: A < B < C
B
A
False
True
B
A
A
False
True
B
A
False
True

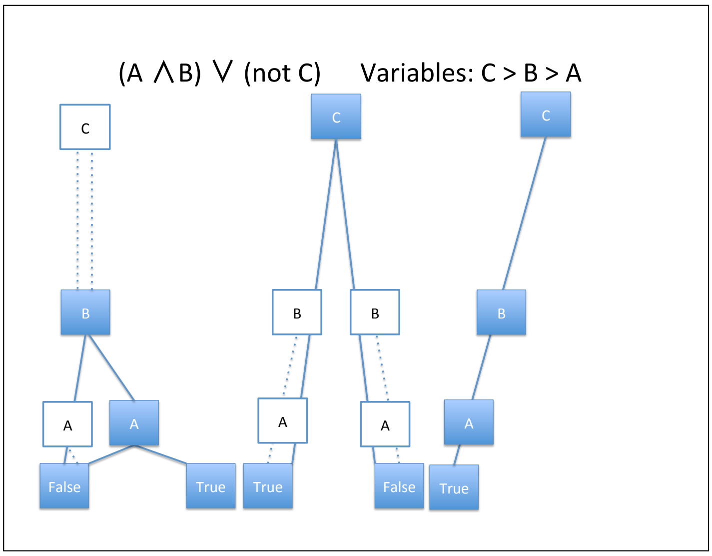

(A ∧B) Variables: A < B < C
B
A
False
True
B
A
A
False
True
B
A
A
False
True

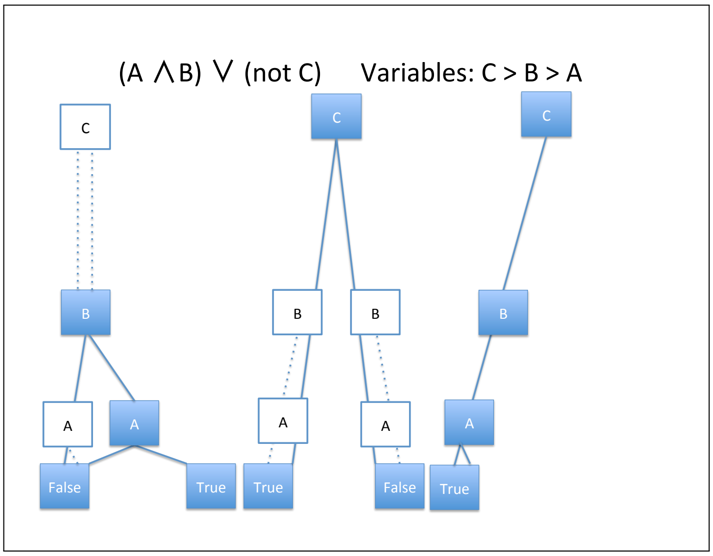

(not C) Variables: C > B > A
C
C
F
T
False
True
not C
C
F
T
True
False

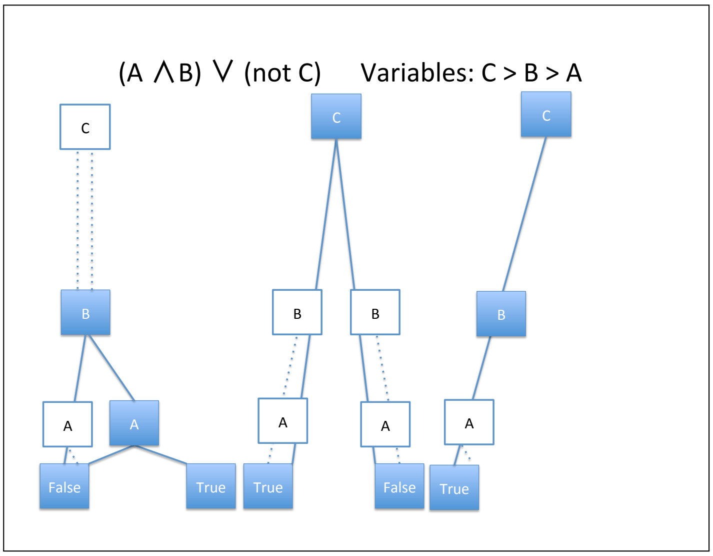

(not C) Variables: C > B > A
not C
C
B
A
A
False
True
True
False

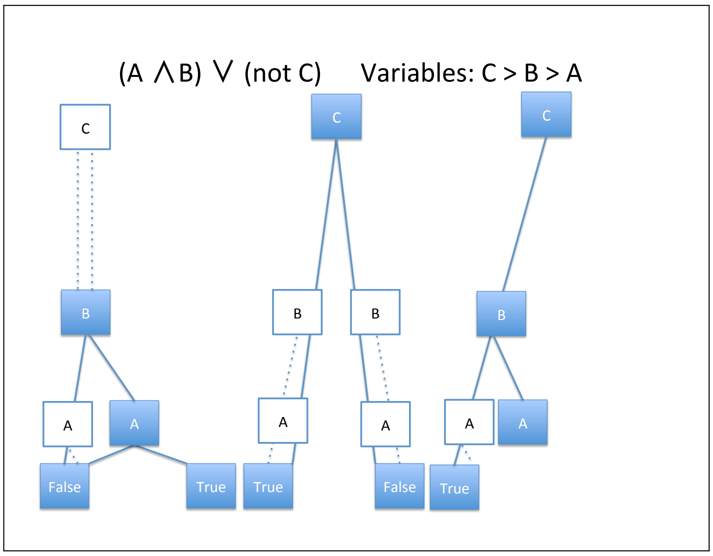

(A ∧B) ∨ (not C) Variables: C > B > A
C
C
B
B
B
A
A
A
A
False
True
True
False

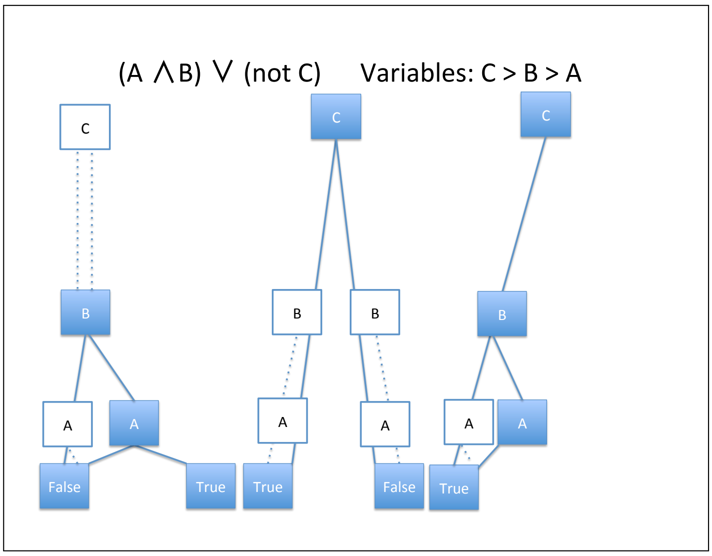

(A ⋀B) ⋁ (not C)
Variables: C > B > A

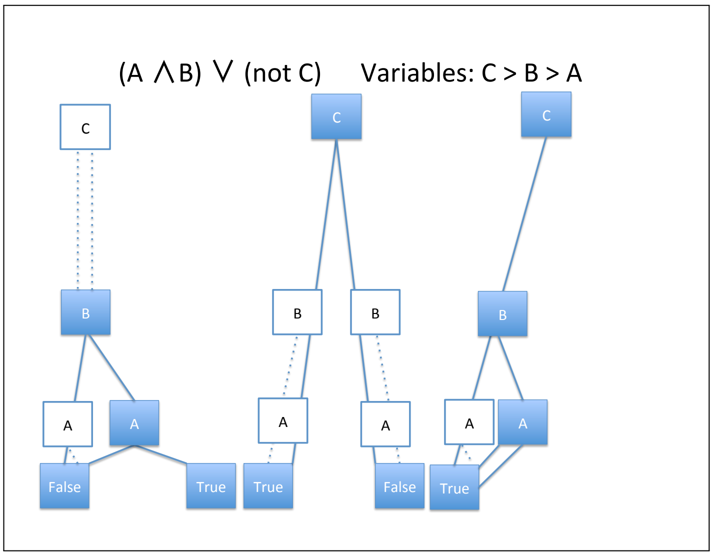
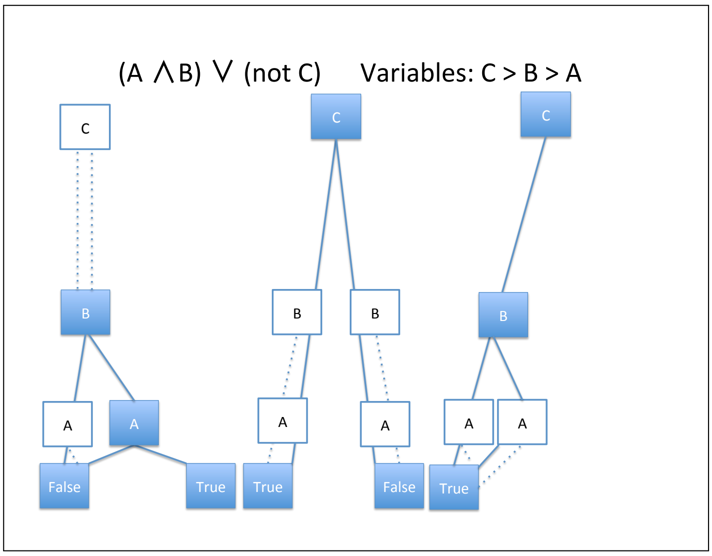
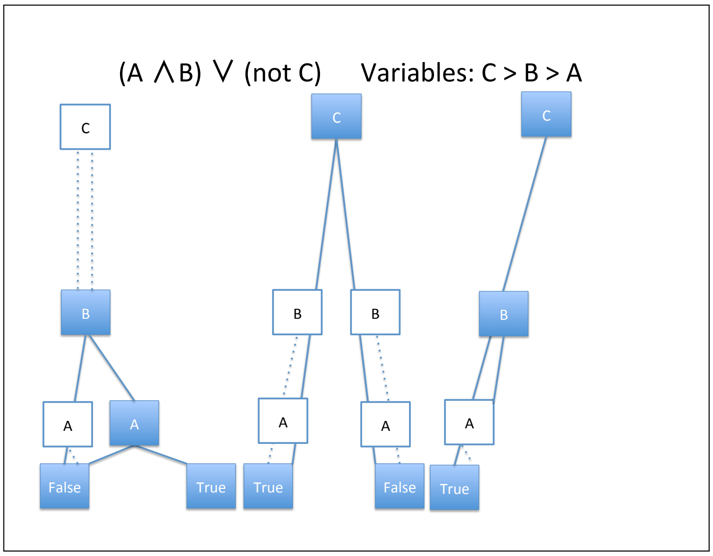
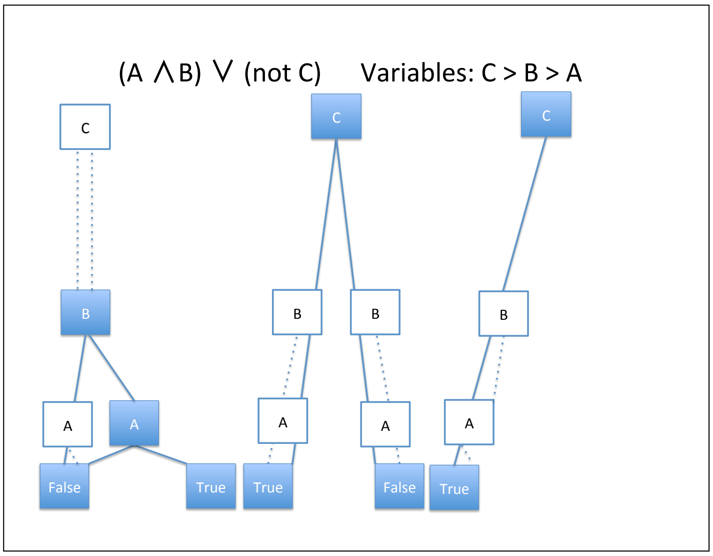
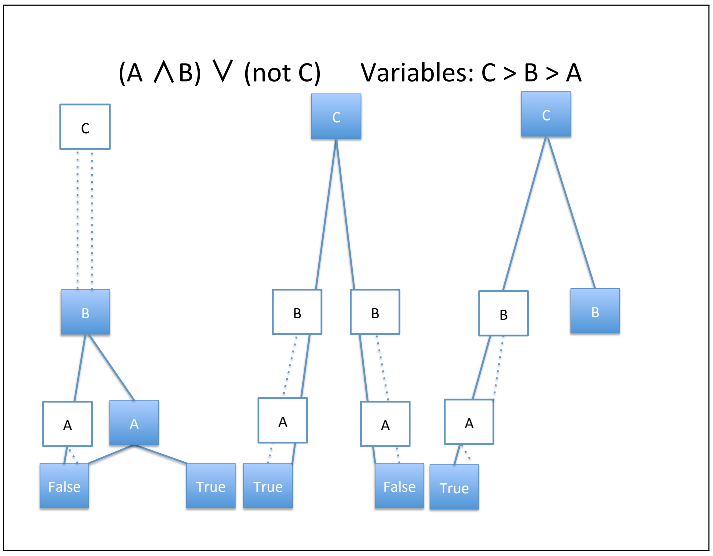
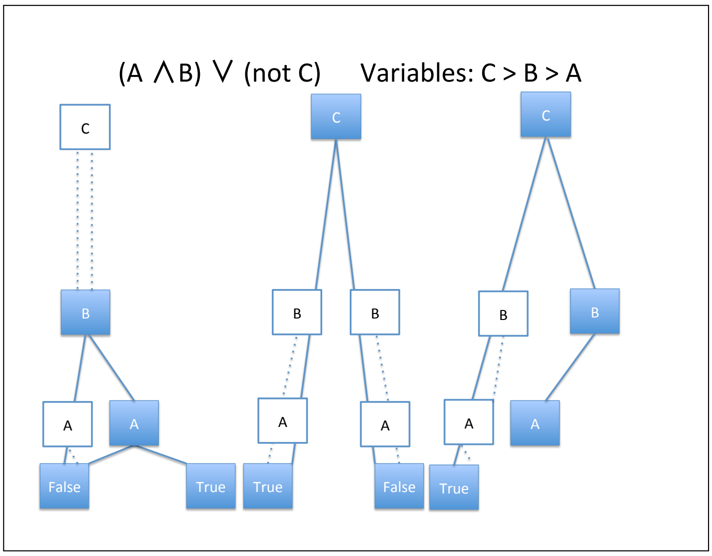

(A ⋀B) ⋁ (not C)
Variables: C > B > A

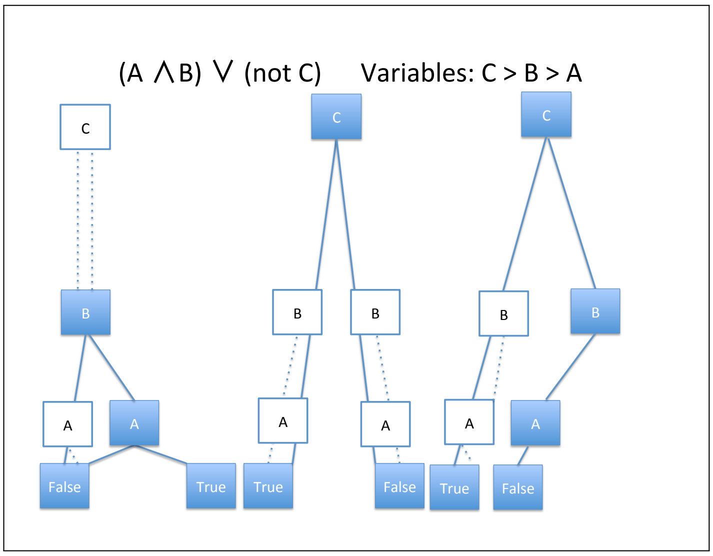

(A ⋀B) ⋁ (not C)
Variables: C > B > A

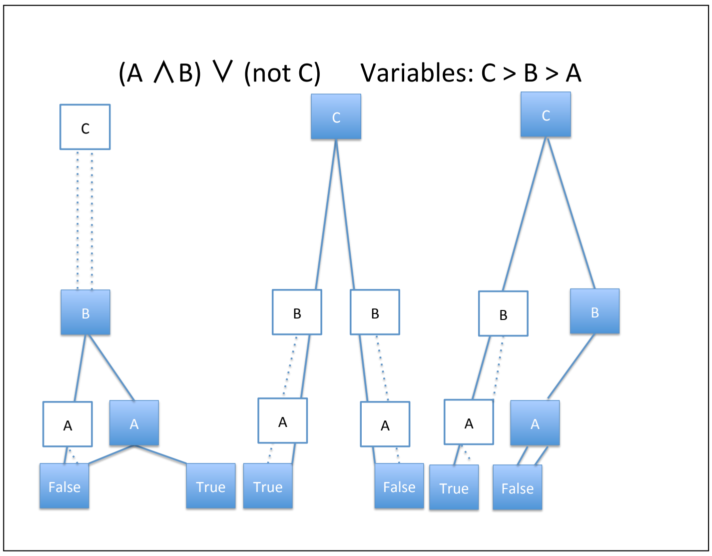

(A ⋀B) ⋁ (not C)
Variables: C > B > A

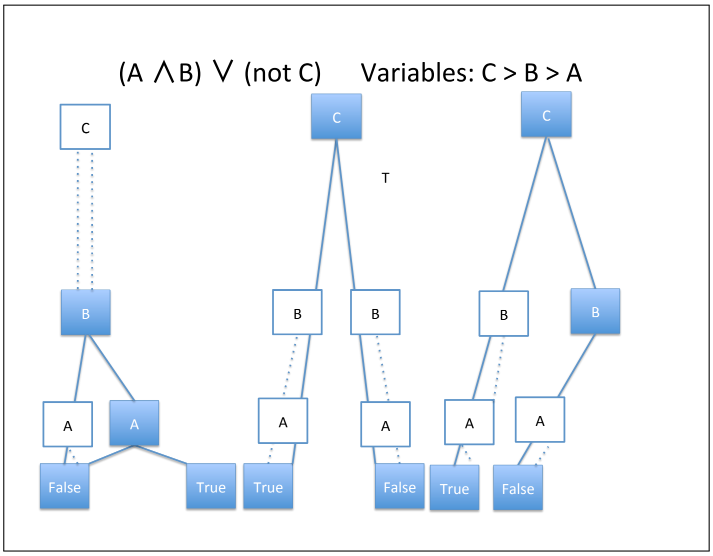

(A ⋀B) ⋁ (not C)
Variables: C > B > A

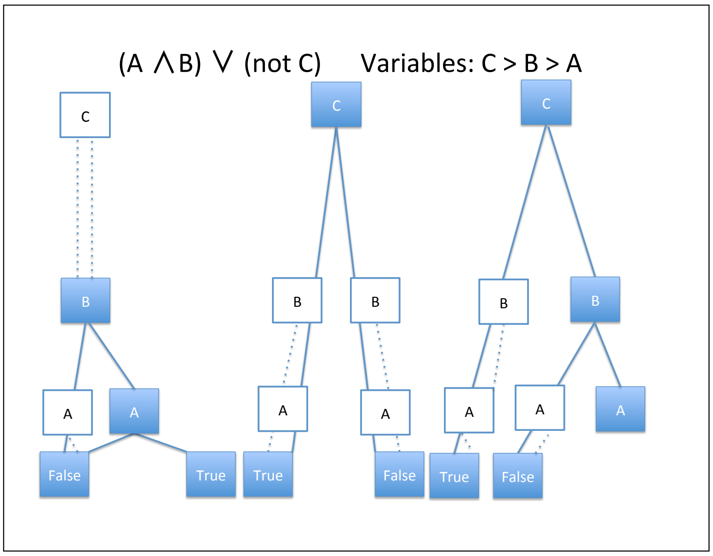

(A ⋀B) ⋁ (not C)
Variables: C > B > A

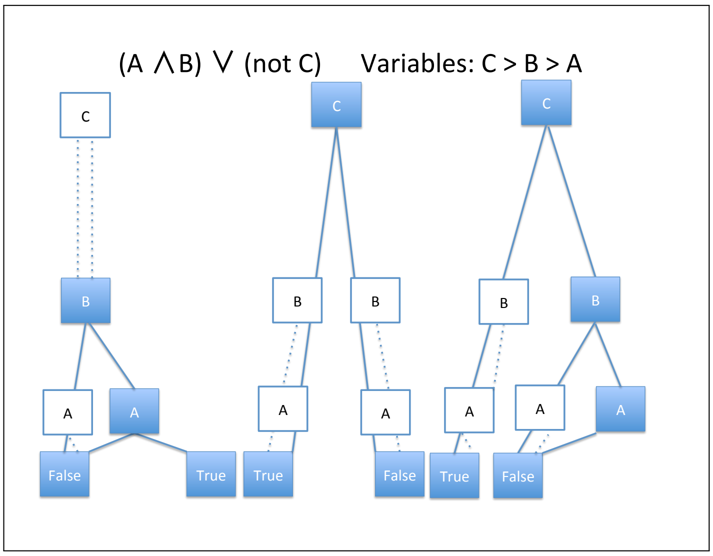

(A ⋀B) ⋁ (not C)   Variables: C > B > A
C
B
A
False
True

(A ⋀B) ⋁ (not C)   Variables: C > B > A
C
B
A
False
True

(A ⋀B) ⋁ (not C)   Variables: C > B > A
C
B
A
False
True

(A ⋀B) ⋁ (not C)   Variables: C > B > A
C
B
A
False
True

(A ⋀B) ⋁ (not C)   Variables: C > B > A
C
B
A
False
True

(A ⋀B) ⋁ (not C)   Variables: C > B > A
C
B
A
False
True

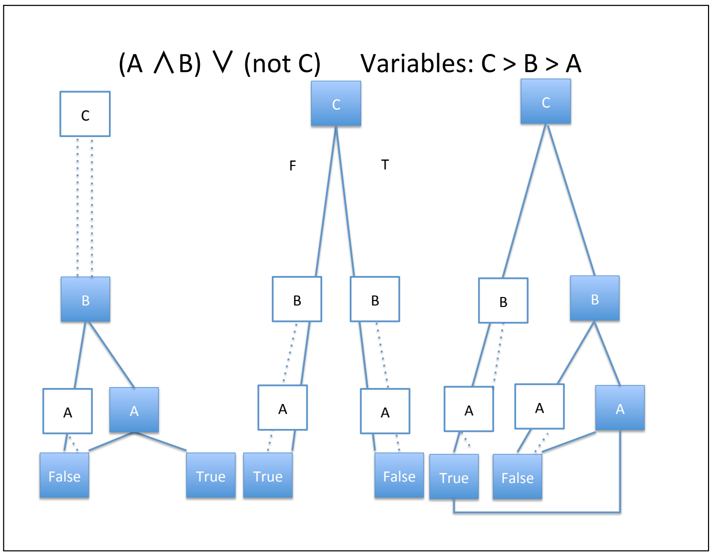

(A ∧B) ∨ (not C) Variables: C > B > A


(A ∧B) ∨ (not C) Variables: C > B > A

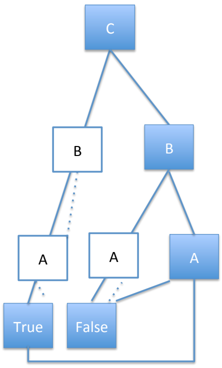

(A ∧B) ∨ (not C) Variables: C > B > A
T


(A ∧B) ∨ (not C) Variables: C > B > A

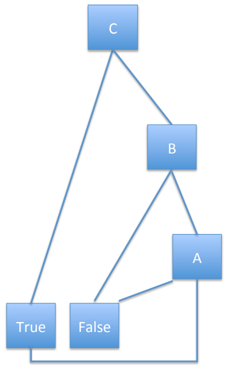

(A ∧B) ∨ (not C) Variables: C > B > A


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

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



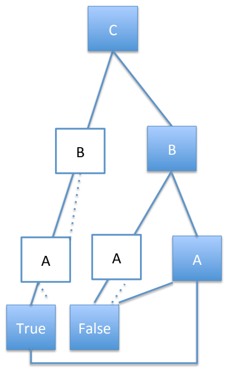
(A ∧B) ∨ (not C)   Variables: C > B > A



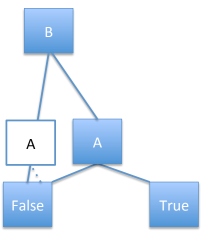
Example

- Find ROBDD for ((A ∧B) ∨ (not C)) ∨ not (A ∧B)

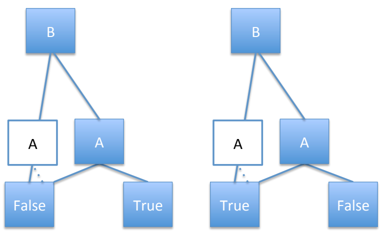
((A ∧B) ∨ (not C))   Variables: C > B > A

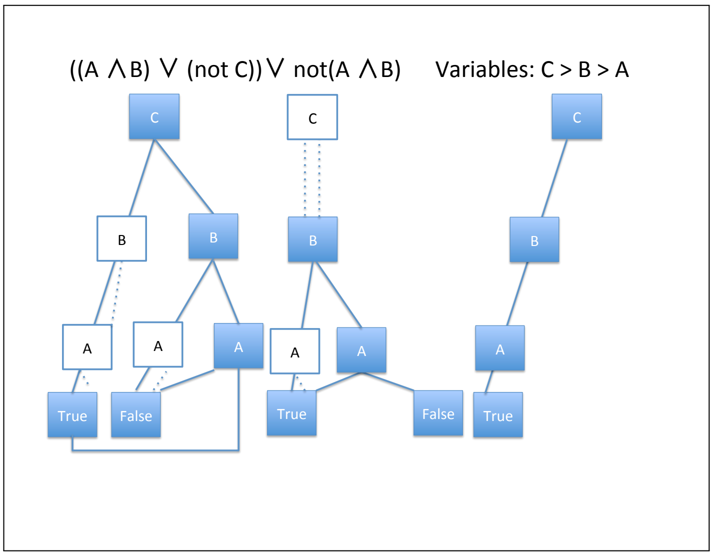


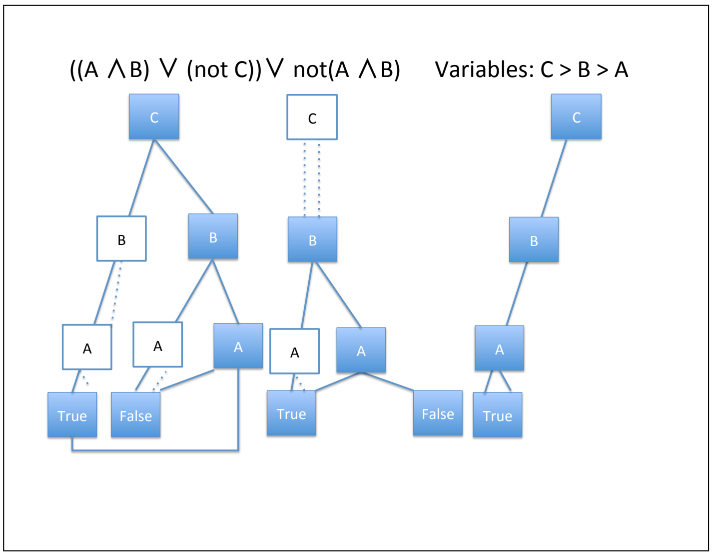
(A ∧B)   Variables: C > B > A

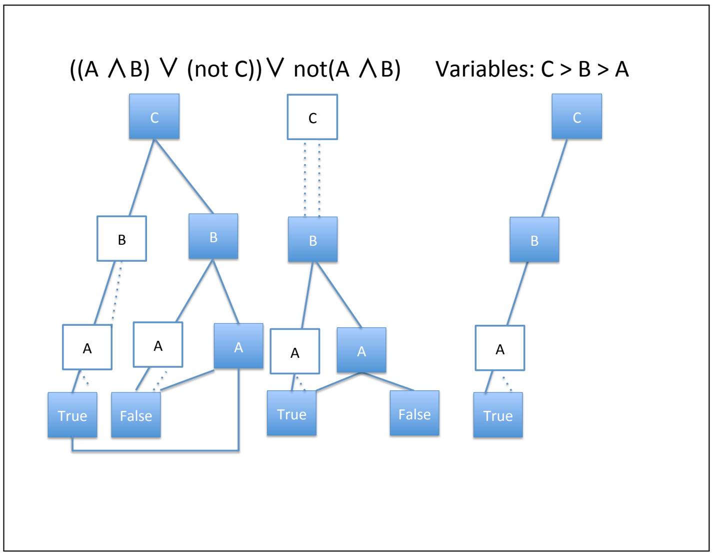


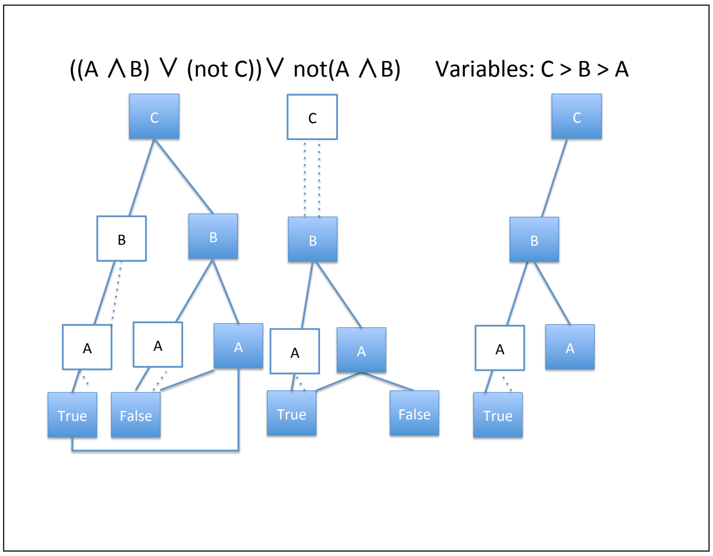
not (A ∧B)   Variables: C > B > A

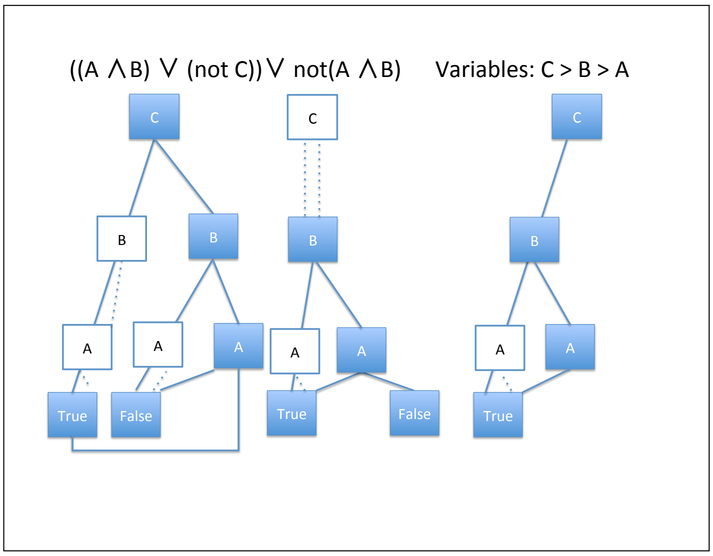


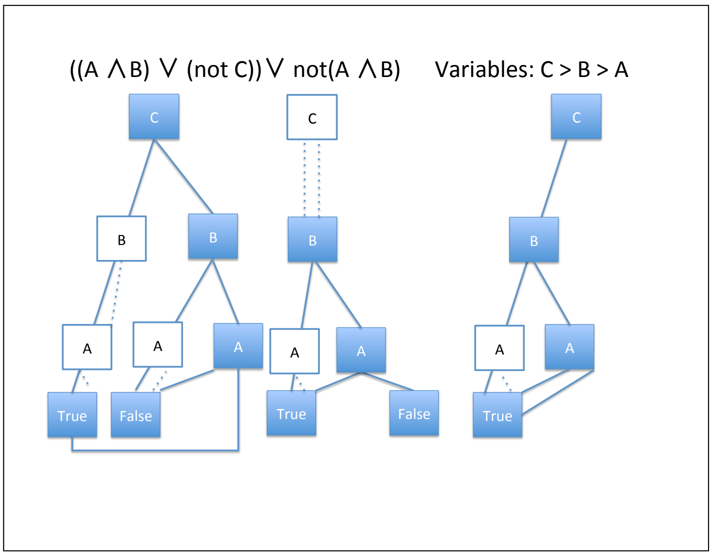

((A ∧B) ∨ (not C))∨ not(A ∧B)
Variables: C > B > A

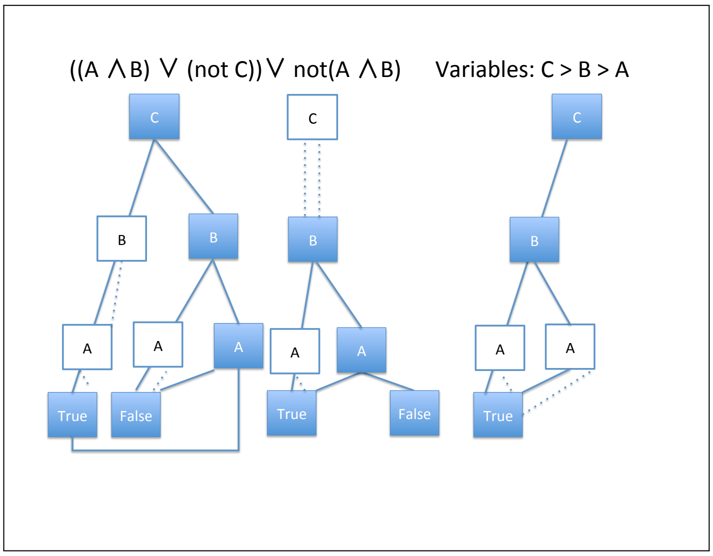

((A ∧B) ∨ (not C))∨ not(A ∧B)
Variables: C > B > A

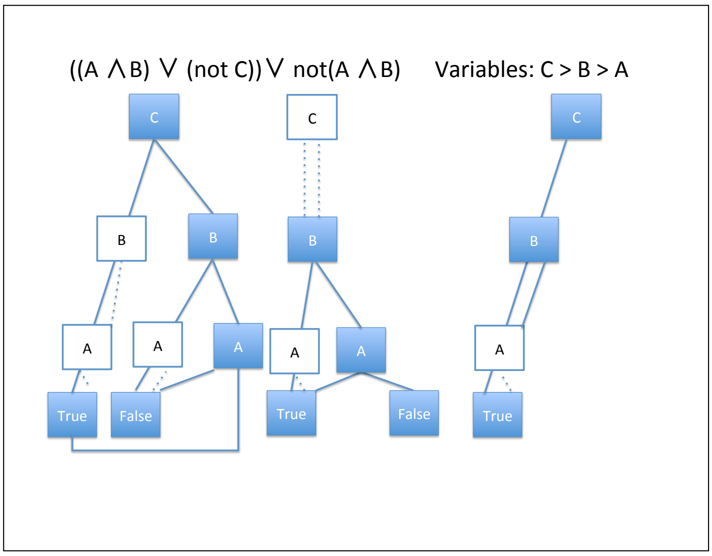

((A ∧B) ∨ (not C))∨ not(A ∧B)
Variables: C > B > A

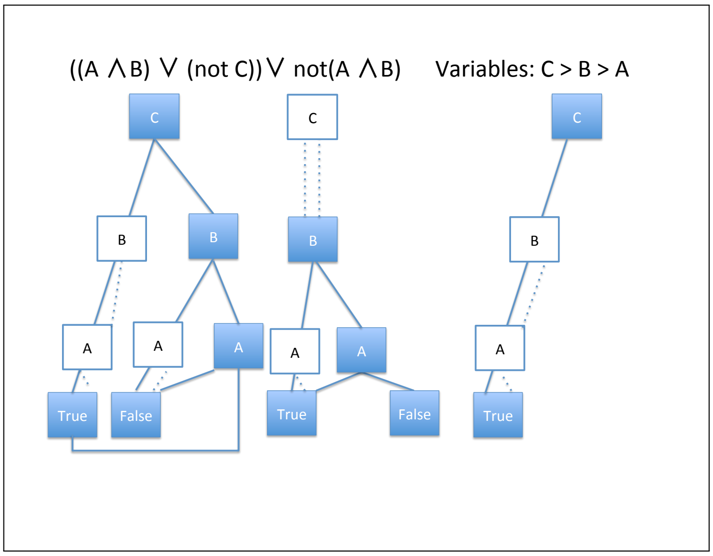

((A ∧B) ∨ (not C))∨ not(A ∧B)
Variables: C > B > A

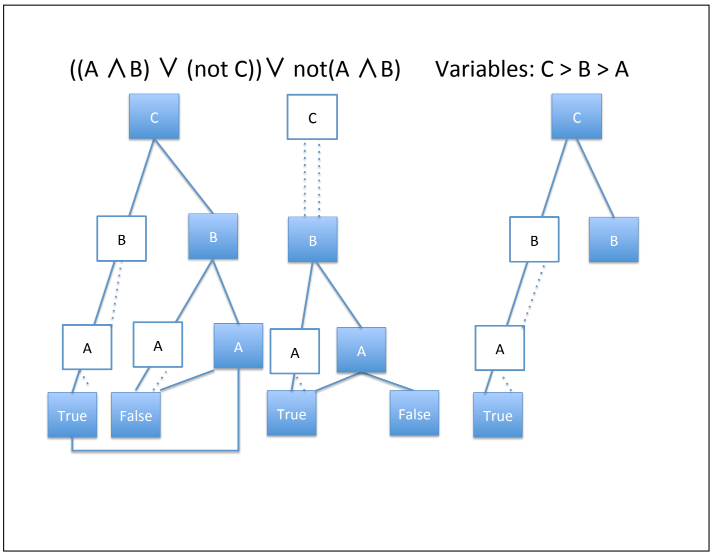

((A ∧B) ∨ (not C))∨ not(A ∧B)
Variables: C > B > A

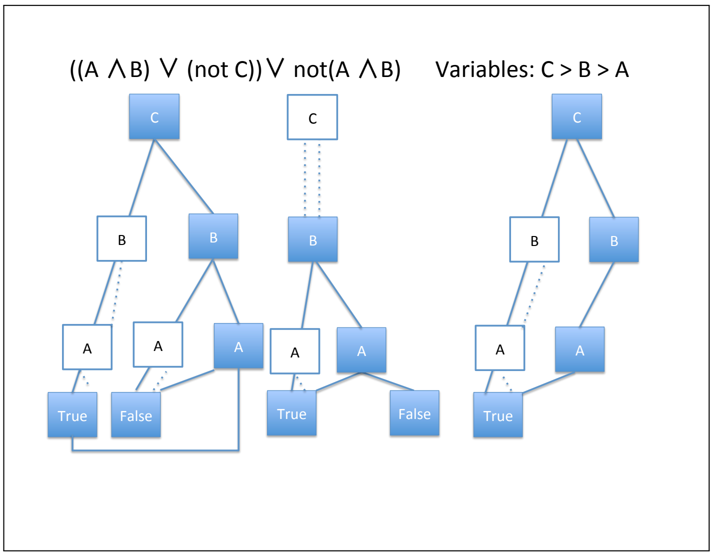

((A ∧B) ∨ (not C))∨ not(A ∧B)
Variables: C > B > A

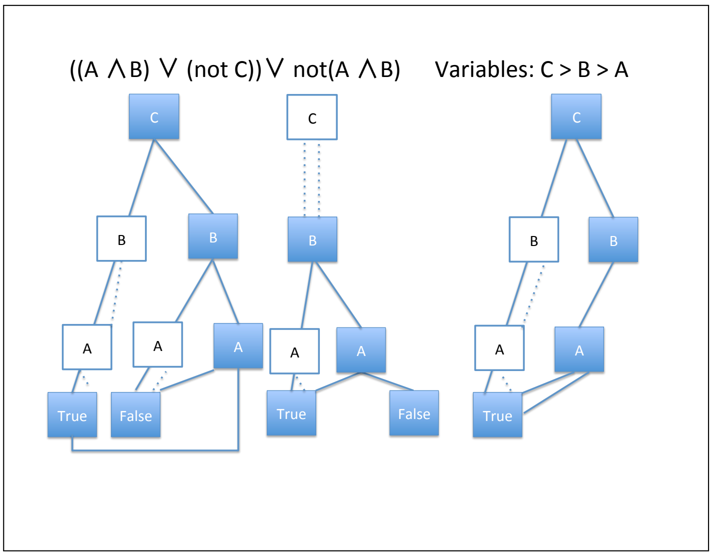

((A ∧B) ∨ (not C))∨ not(A ∧B)
Variables: C > B > A


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

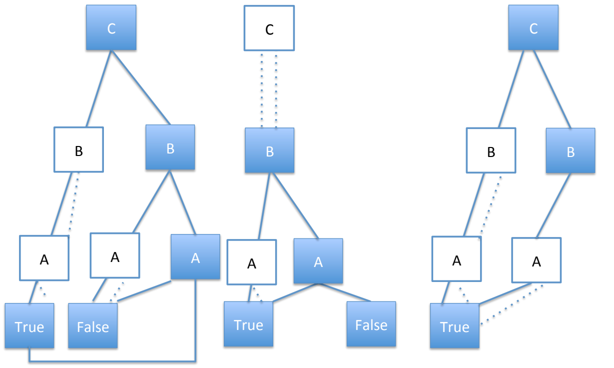

((A ∧B) ∨ (not C))∨ not(A ∧B)
Variables: C > B > A

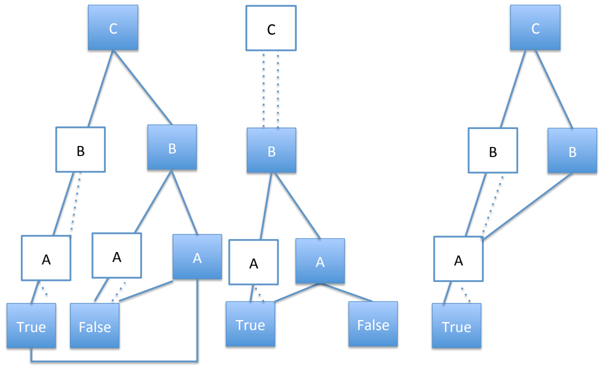

((A ∧B) ∨ (not C))∨ not(A ∧B)
Variables: C > B > A

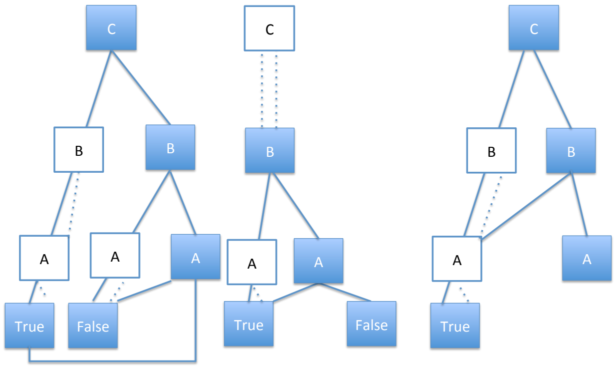

((A ∧B) ∨ (not C))∨ not(A ∧B)
Variables: C > B > A

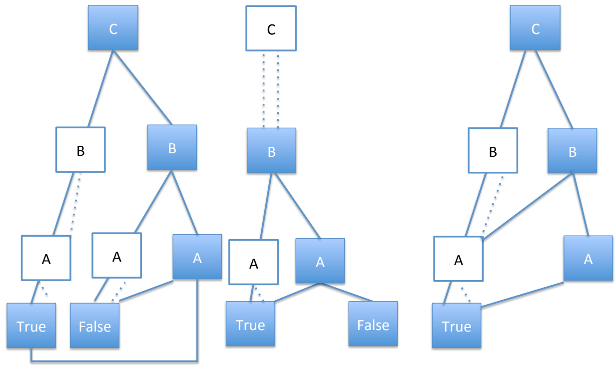

((A ∧B) ∨ (not C))∨ not(A ∧B)
Variables: C > B > A

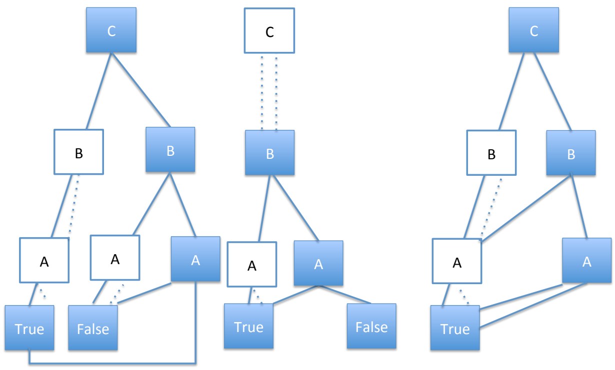
$((A \wedge B) \vee (\text{not } C)) \vee \text{not}(A \wedge B)$ Variables: C > B > A

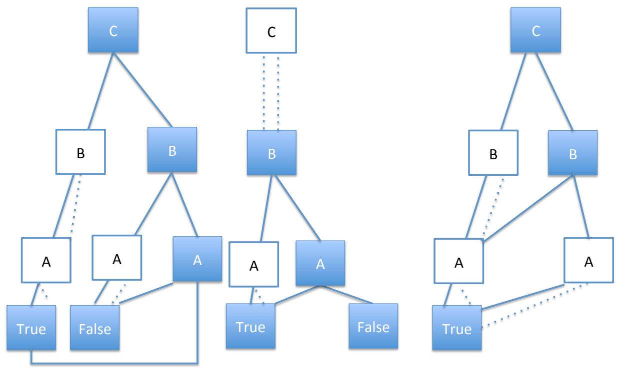




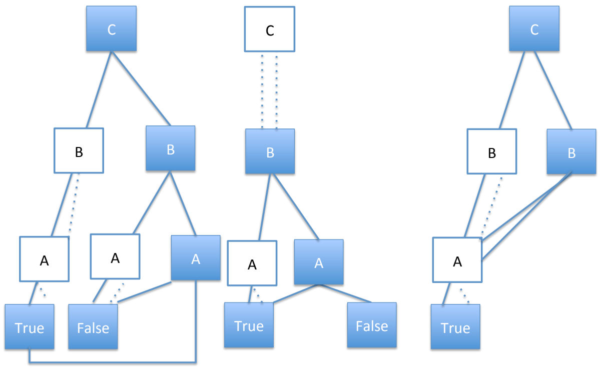




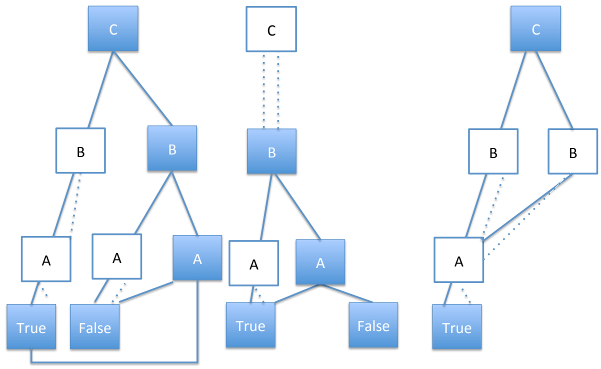




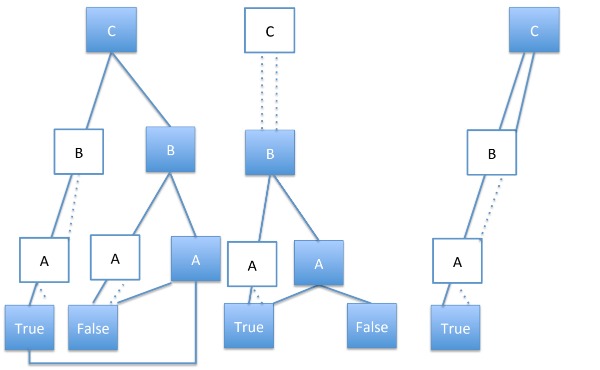
((A ∧B) ∨ (not C)) ∨ not(A ∧B)   Variables: C > B > A



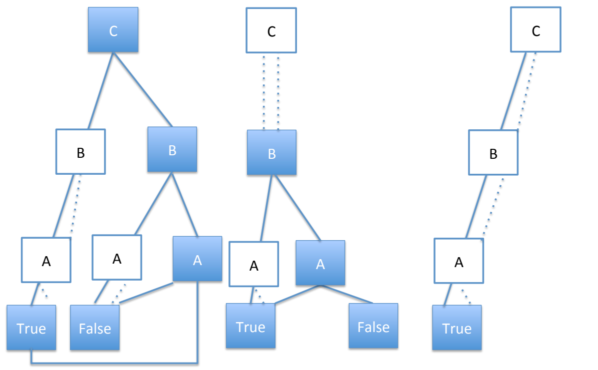
((A ∧B) ∨ (not C)) ∨ not(A ∧B)   Variables: C > B > A



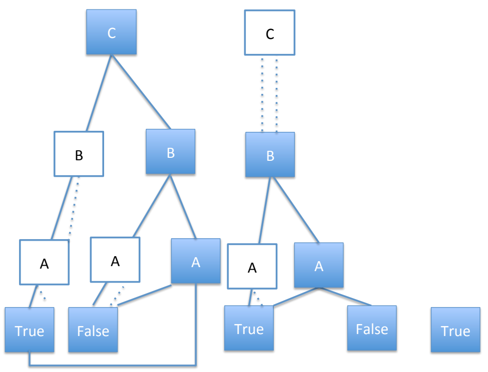
((A ∧B) ∨ (not C)) ∨ not(A ∧B)   Variables: C > B > A



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



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



Uses of ROBDDs

- Reduced Order BDD for a proposition is unque 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