COL:750

Foundations of Automatic Verification

Instructor: Priyanka Golia

Course Webpage 

https://priyanka-golia.github.io/teaching/COL-750/index.html

Interpolants based Model Checking

Interpolants: Introduced by Craig in 1957

Let A and B be two formulas such that : $A \wedge B \vDash \bot$

then, there exists a formula I called Interpolant such that:

1. $A \rightarrow I$
2. $I \wedge B \vDash \bot$
3. $Vars(I) \subseteq Vars(A) \cap Vars(B)$

It acts as a kind of summary or abstraction of A relevant to the contradiction with B.

$A = (p \vee q) \wedge (\neg p \vee r)$ $B = \neg q \wedge \neg r$ $I = (q \vee r)$

How to Compute Interpolants!

1. $A \wedge B$ are unsatisfiable.

SAT solver can return resolution proof!

All the initial nodes have in-degree 0. All internal nodes have in-degree 2. Sink nodes has out-degree 0.

Internal node v, with edges (v_1, v), (v_2, v) implies that v is a resolvent of v_1, v_2

$$A = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \qquad B = (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$

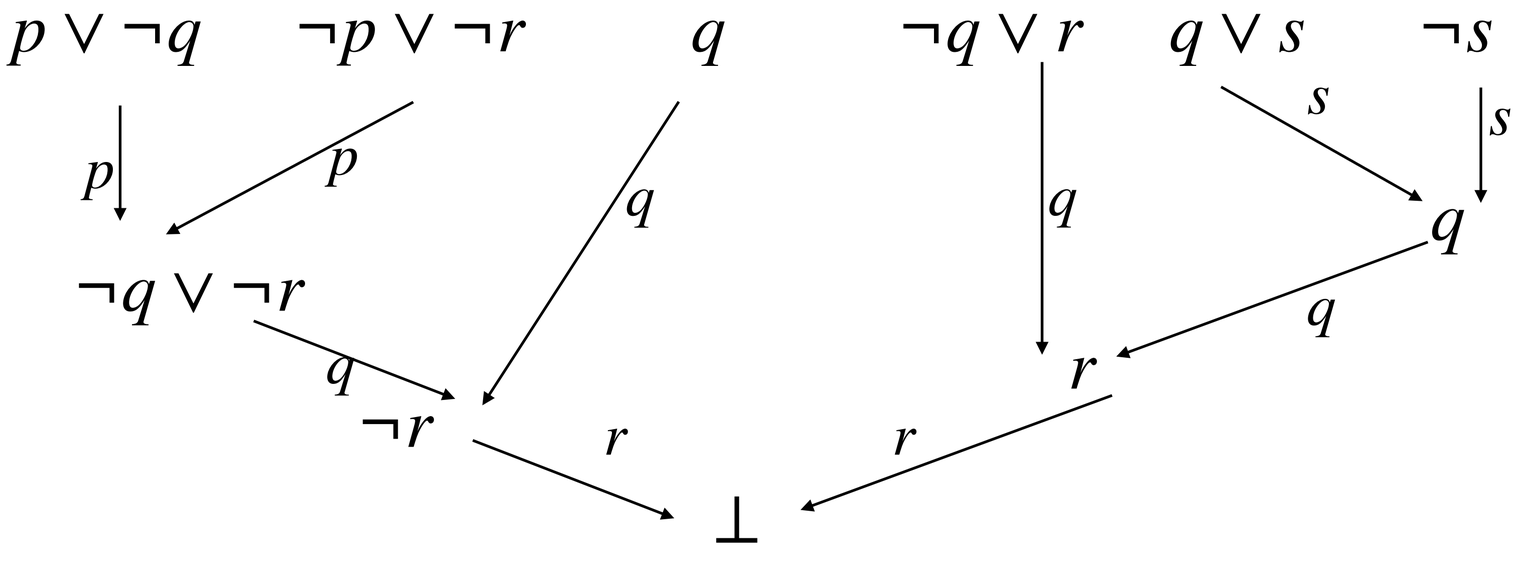
How to Compute Interpolants!

1. $A \wedge B$ are unsatisfiable. SAT solver can return resolution proof!

All the initial nodes have in-degree 0. All internal nodes have in-degree 2. Sink nodes has out-degree 0. Internal node v, with edges (v_1, v), (v_2, v), v is a resolvent of v_1, v_2

$$A = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \quad B = (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$

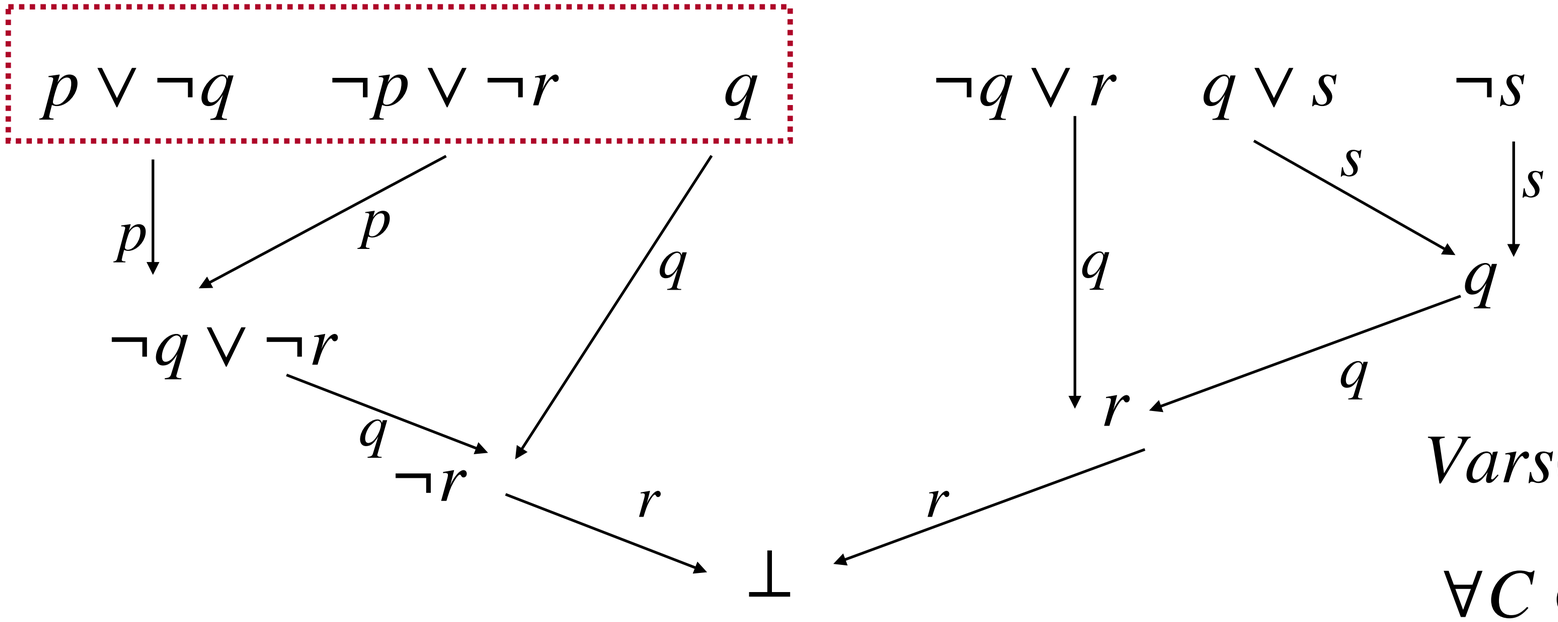
$$A \wedge B = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \wedge (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$



How to Compute Interpolants!

$$A = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \qquad B = (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$

$$A \wedge B = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \wedge (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$



$$Vars(I) \subseteq Vars(A) \wedge Vars(B)$$

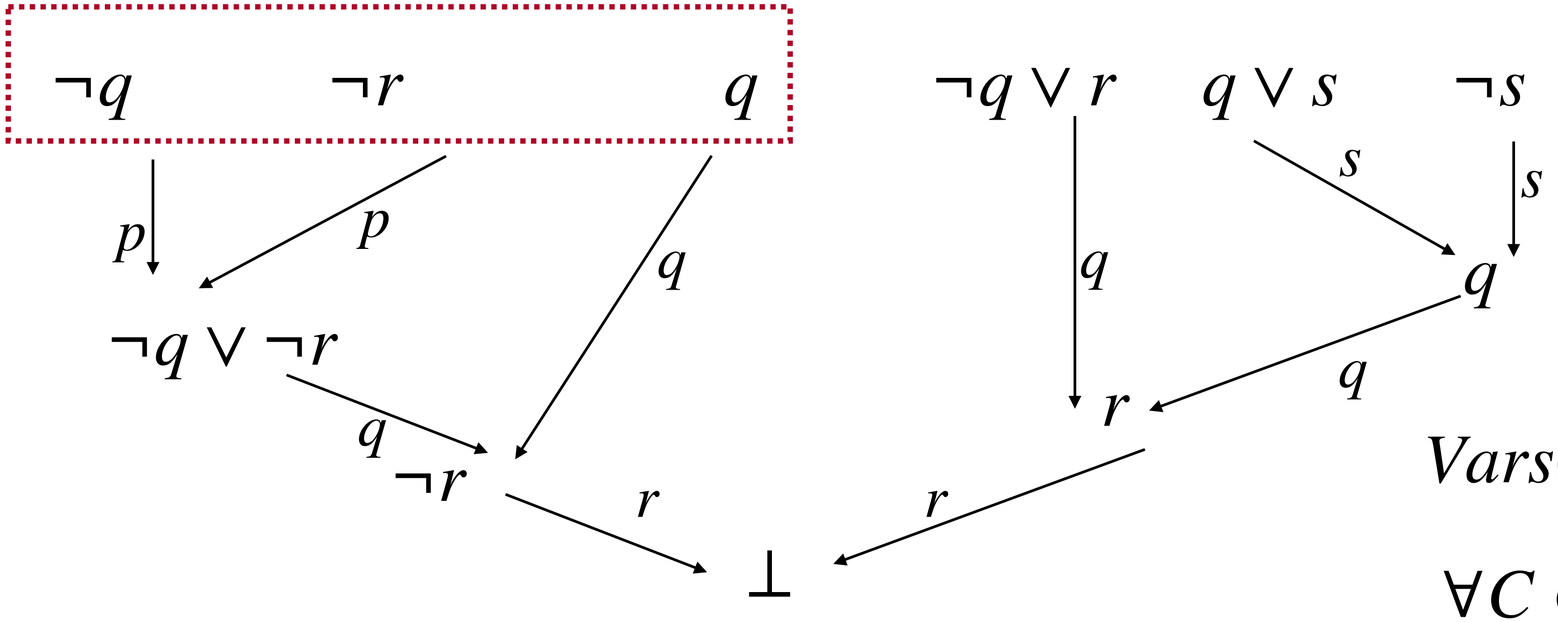
$$\forall C \in Clauses(A), C_{\downarrow(Vars(B))}$$

$$A \rightarrow I$$

How to Compute Interpolants!

$A = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \qquad B = (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$

$A \wedge B = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \wedge (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$



$Vars(I) \subseteq Vars(A) \wedge Vars(B)$

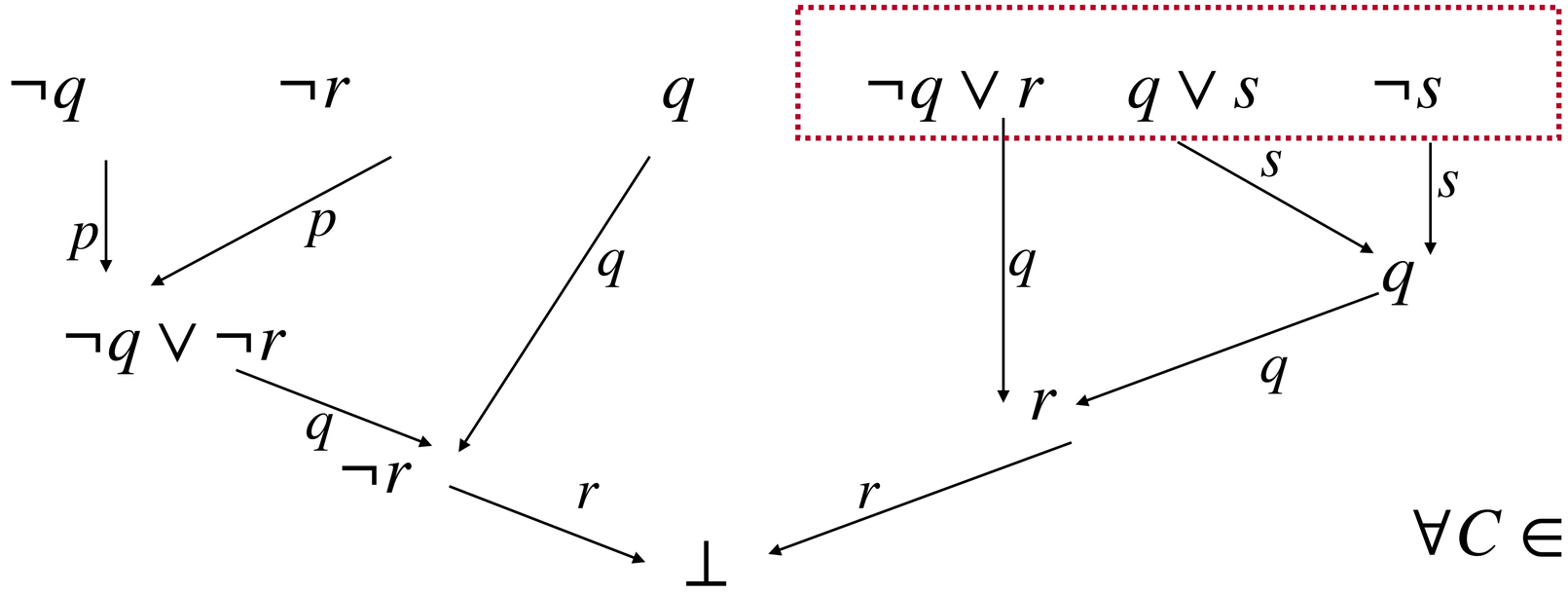
$\forall C \in Clauses(A), C_{\downarrow(Vars(B))}$

$A \rightarrow I$

How to Compute Interpolants!

$A = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \qquad B = (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$

$A \wedge B = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \wedge (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$



$\forall C \in Clauses(B), True$

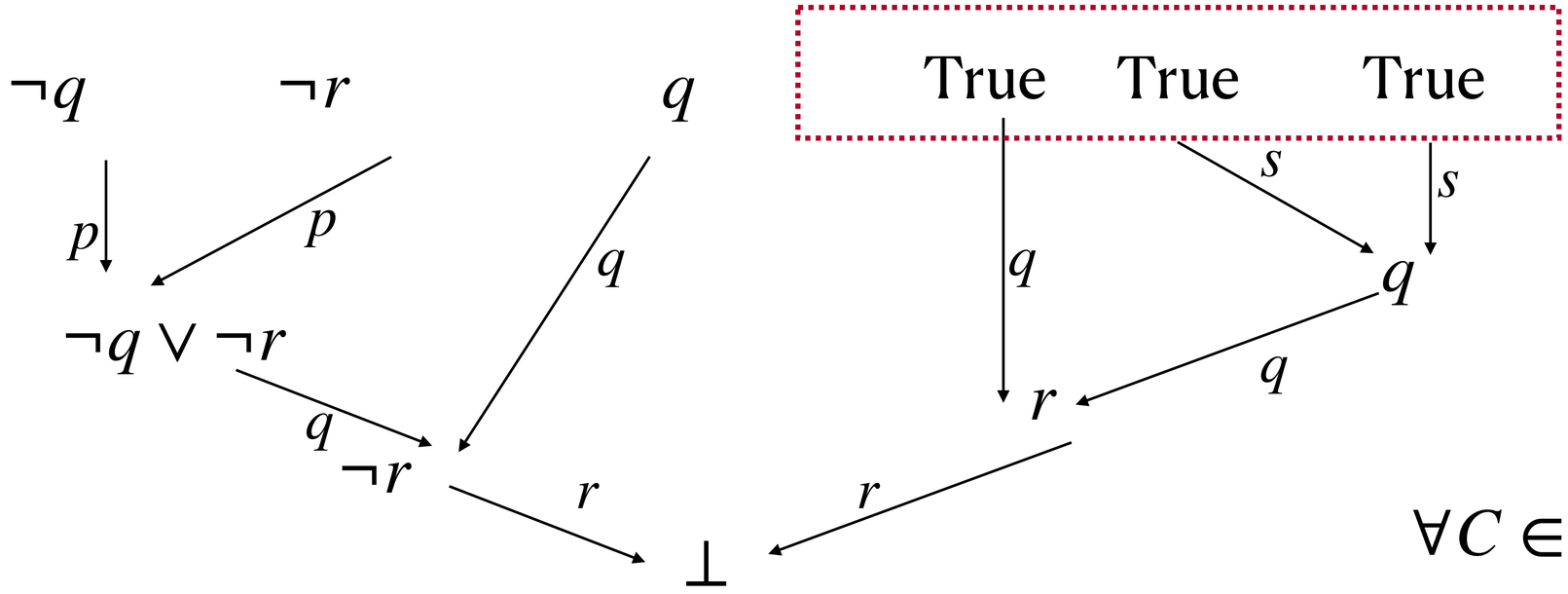
$Clauses(B)$ doesn't contribute to I

$I \wedge B \vDash \bot$ will be taken care by internal nodes.

How to Compute Interpolants!

$A = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \qquad B = (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$

$A \wedge B = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \wedge (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$



$\forall C \in Clauses(B), True$

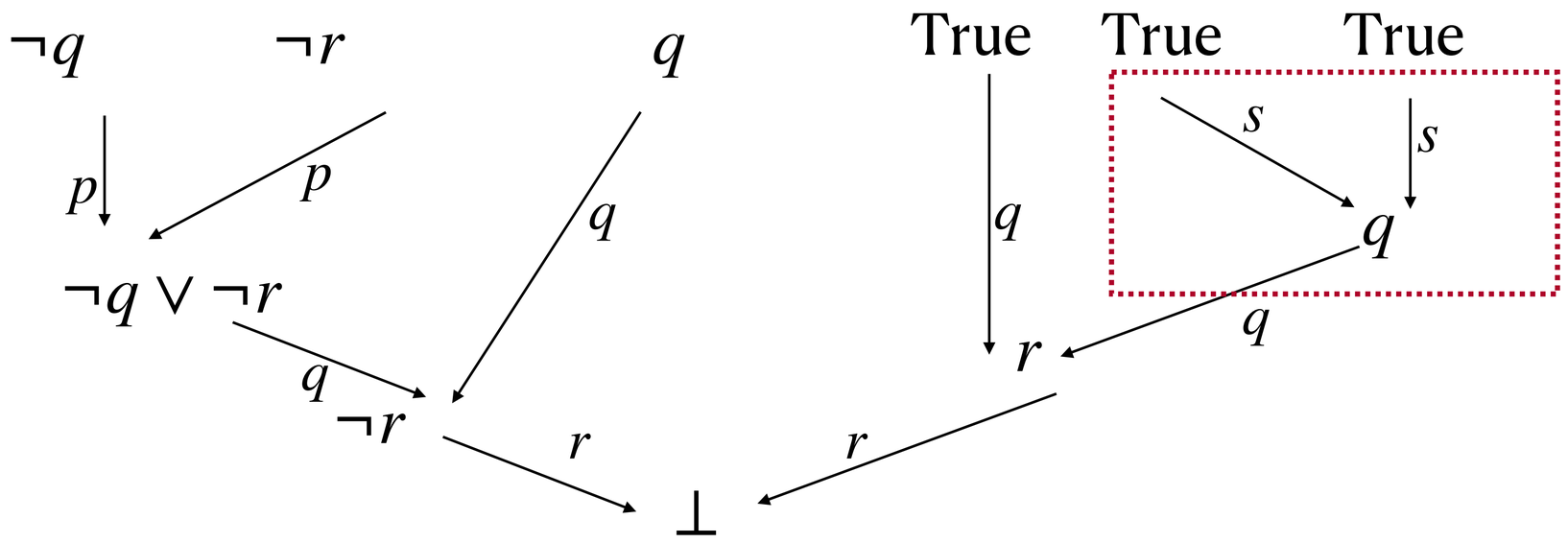
$Clauses(B)$ doesn't contribute to I

$I \wedge B \models \bot$ will be taken care by internal nodes.

How to Compute Interpolants!

$$A = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \qquad B = (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$

$$A \wedge B = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \wedge (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$



When pivot variable is B

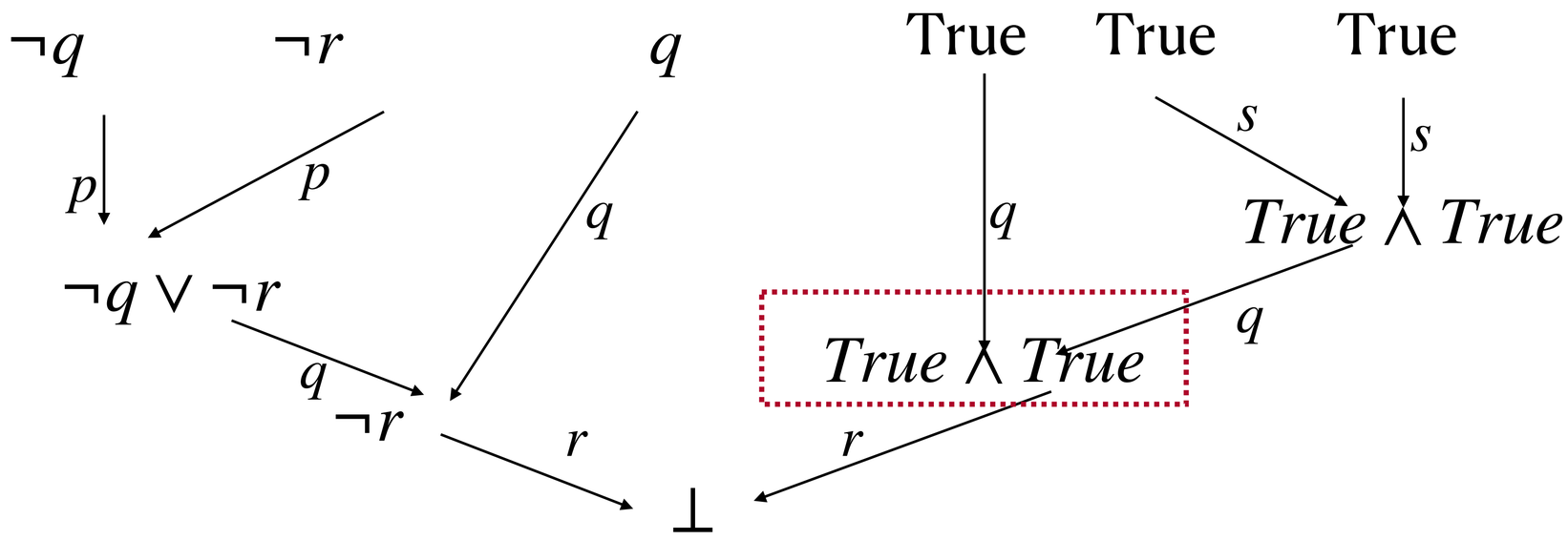
Internal nodes will be “AND” of its both source nodes

To preserve the contradiction with B.
“both source should be considered.

How to Compute Interpolants!

$$A = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \qquad B = (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$

$$A \wedge B = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \wedge (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$



When pivot variable is B

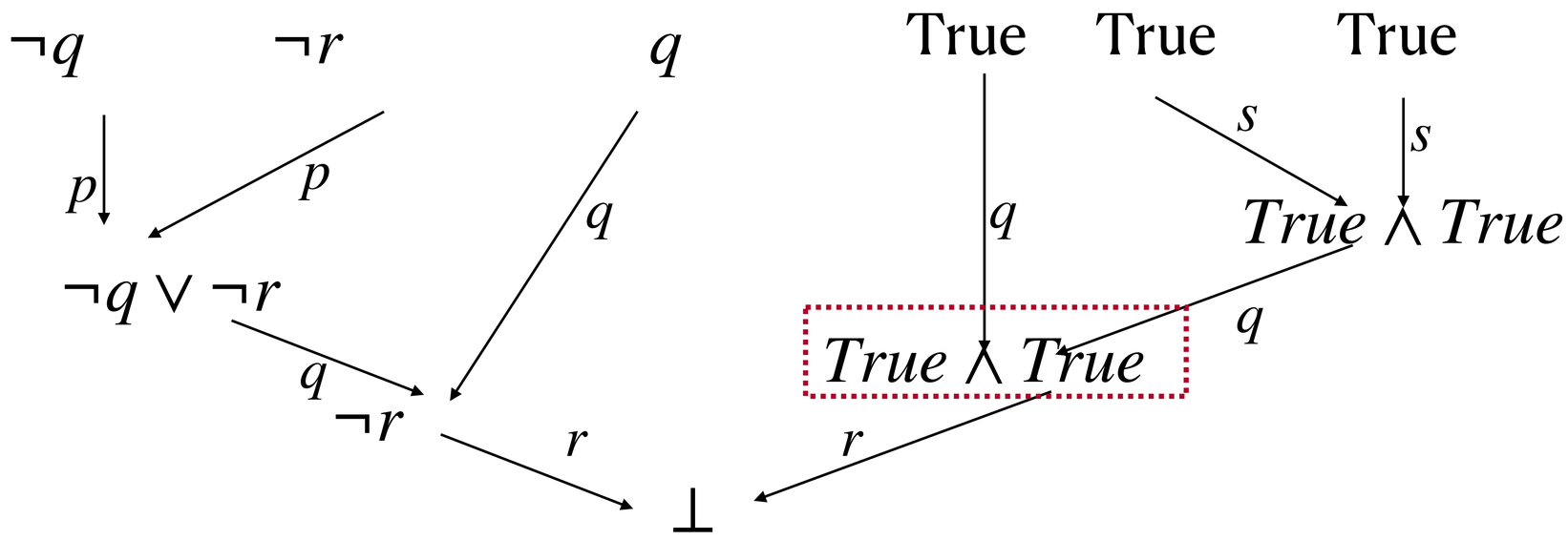
Internal nodes will be “AND” of its both source nodes

To preserve the contradiction with B.
“both source should be considered.

How to Compute Interpolants!

$$A = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \qquad B = (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$

$$A \wedge B = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \wedge (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$



When pivot variable is B

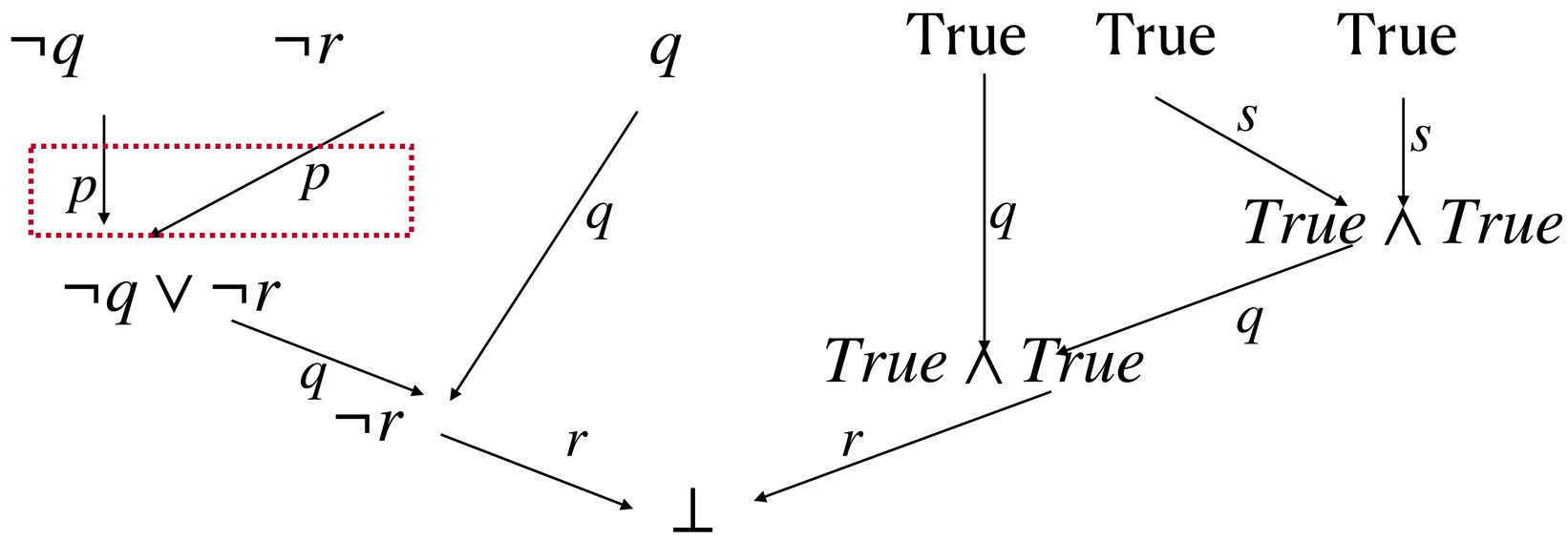
Internal nodes will be “AND” of its both source nodes

To preserve the contradiction with B.
“both” source should be considered.

How to Compute Interpolants!

$$A = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \qquad B = (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$

$$A \wedge B = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \wedge (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$



When pivot variable is A

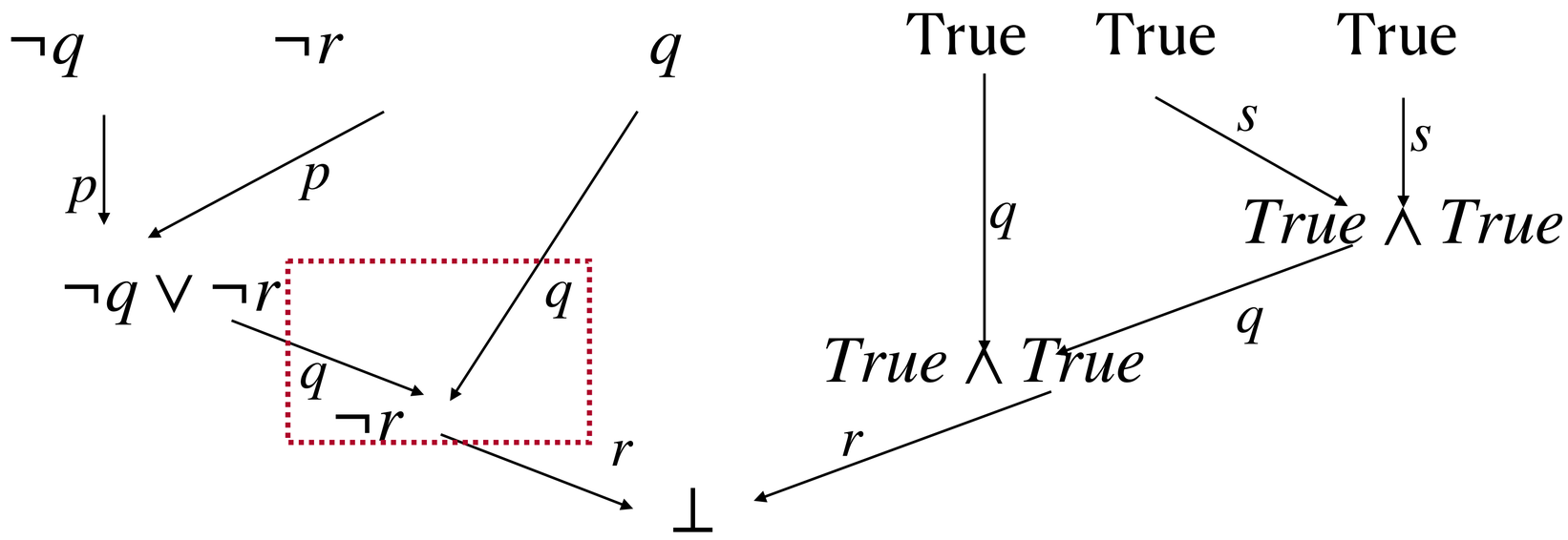
Internal nodes will be “OR” of its both source nodes

To preserve the implication.
Node implies either left clause or right clause

How to Compute Interpolants!

$$A = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \qquad B = (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$

$$A \wedge B = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \wedge (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$



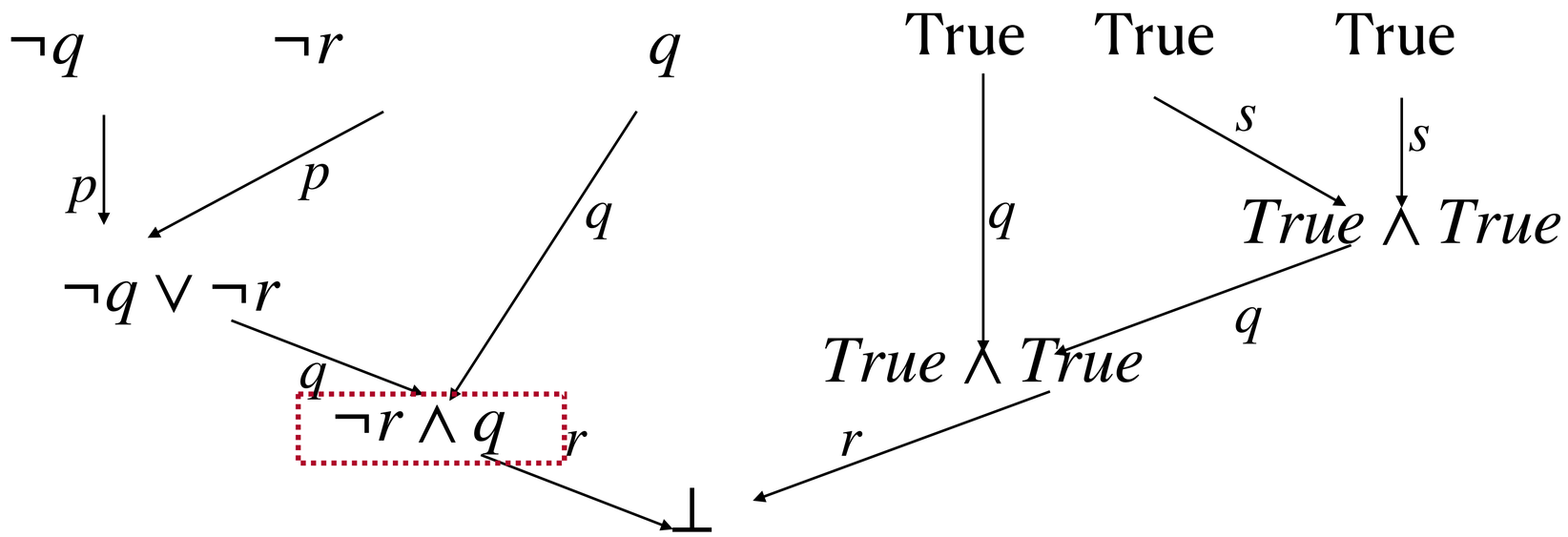
$q \in Vars(B)$

To preserve the contradiction with B.
“both” source should be considered.

How to Compute Interpolants!

$$A = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \qquad B = (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$

$$A \wedge B = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \wedge (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$$



$q \in Vars(B)$

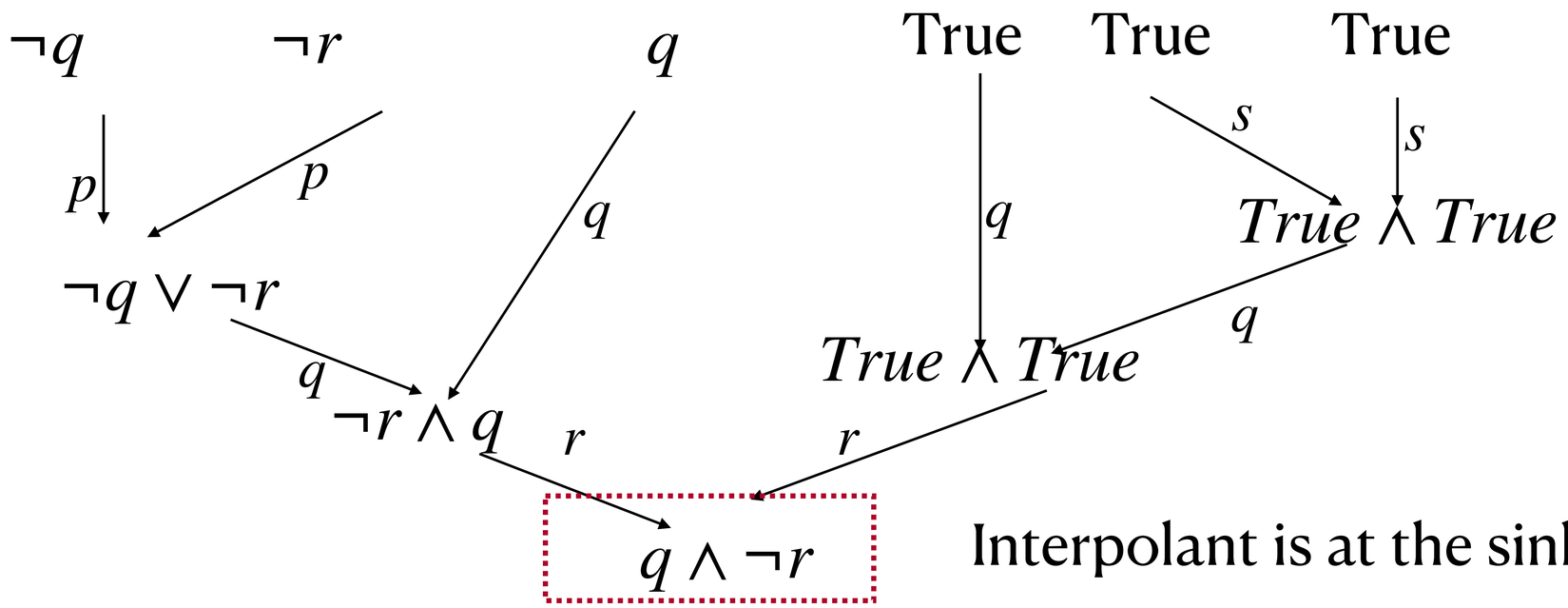
To preserve the contradiction with B.
“both” source should be considered.

How to Compute Interpolants!

$A = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \qquad B = (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$

$I = q \wedge \neg r$

$A \wedge B = (p \vee \neg q) \wedge (\neg p \vee \neg r) \wedge q \wedge (\neg q \vee r) \wedge (q \vee s) \wedge \neg s$



Interpolant is at the sink node

$r \in Vars(B)$

To preserve the contradiction with B.
“both” source should be considered.

How to Compute Interpolants!

McMillan interpolation algorithm (2003)

All the initial nodes have in-degree 0. All internal nodes have in-degree 2. Sink nodes has out-degree 0.

Internal node v, with edges $(v_1, v), (v_2, v)$ implies that v is a resolvent of v_1, v_2

1. Compute Resolution Proof of A and B

2. Base case (input clauses):

If $C \in Clauses(A)$:

$$I_c = C_{\downarrow(Vars(B))}$$

If $C \in Clauses(B)$:

$$I_c = True$$

3. Resolution step: C is Derived by C_1 and C_2 over pivot x.

If $x \in Vars(B)$:

$$I_C = I_{C_1} \wedge I_{C_2}$$

If $x \in Vars(A)$:

$$I_c = I_{c_1} \vee I_{c_2}$$

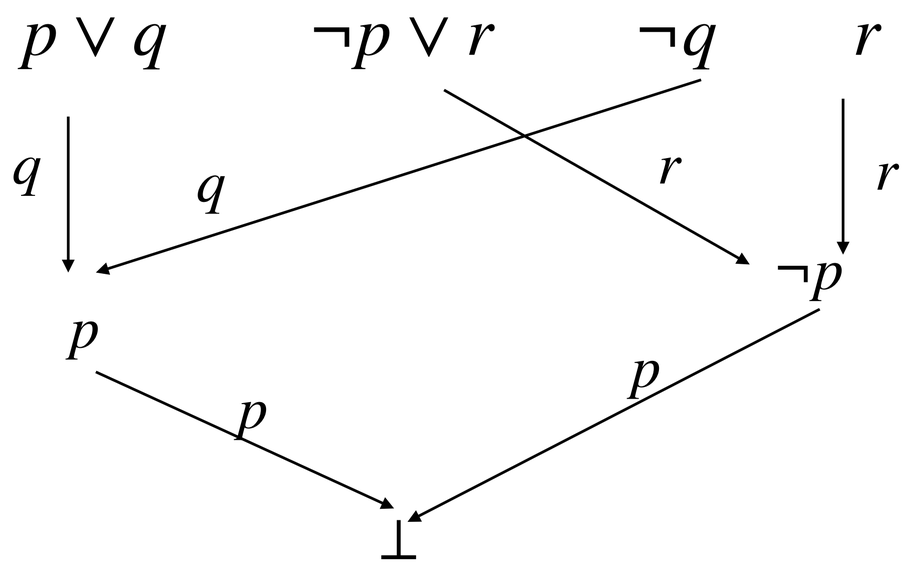
Interpolant of A,B is $I_\perp$

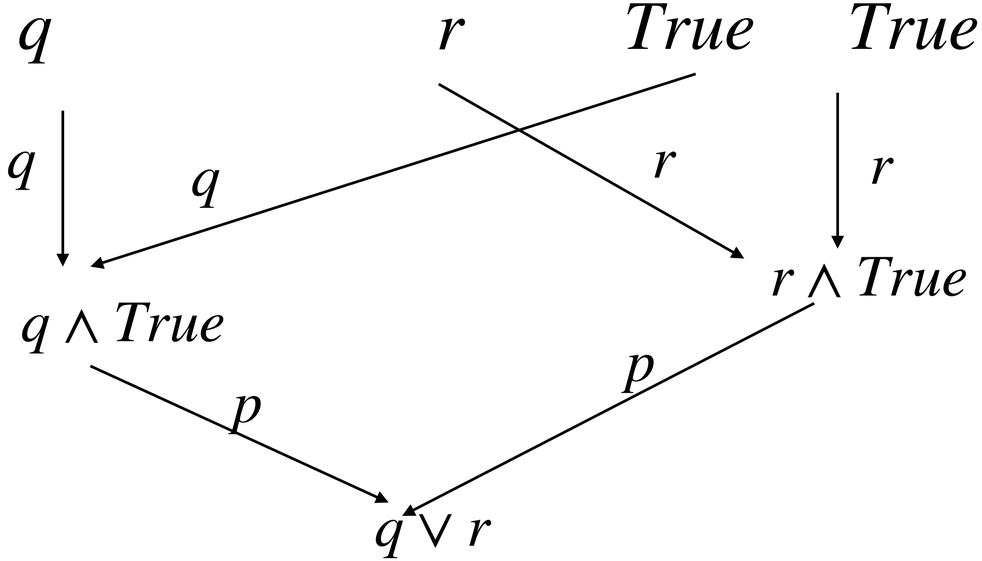
Compute Interpolants

$$A = (p \vee q) \wedge (\neg p \vee r) \qquad B = \neg q \wedge \neg r$$

Compute Interpolants

$$A = (p \vee q) \wedge (\neg p \vee r) \qquad B = \neg q \wedge \neg r$$



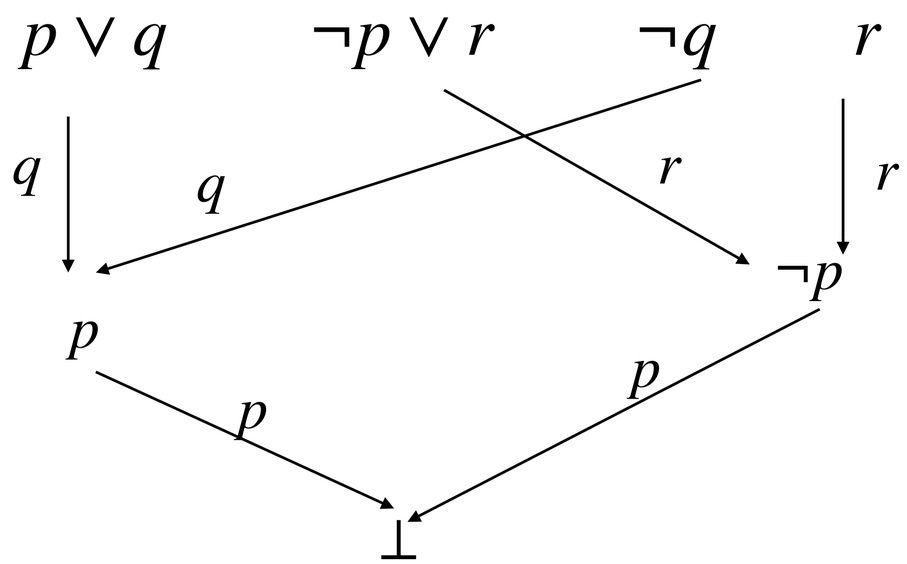


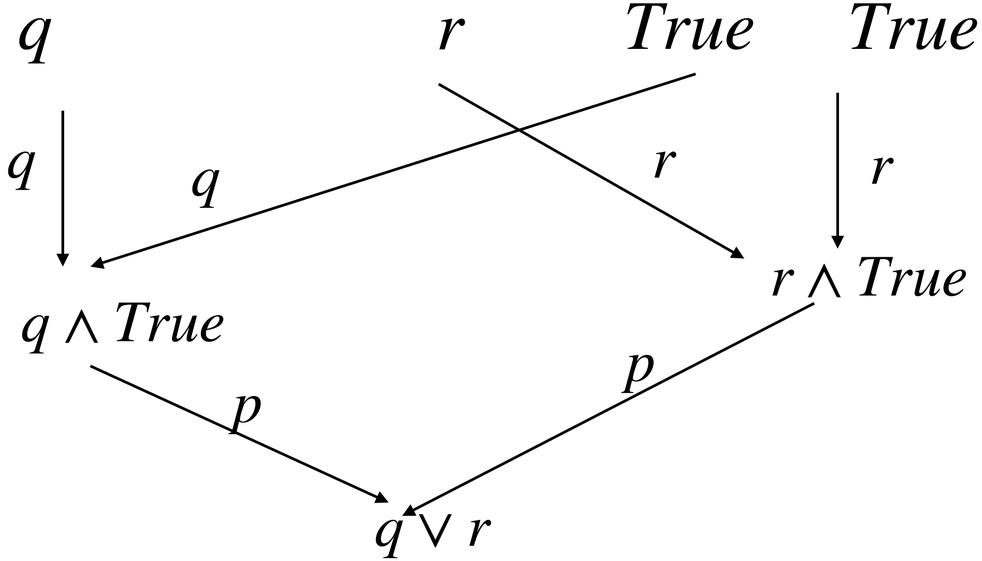
Compute Interpolants

$$A = (p \vee q) \wedge (\neg p \vee r) \qquad B = \neg q \wedge \neg r$$

Compute Interpolants

$$A = (p \vee q) \wedge (\neg p \vee r) \qquad B = \neg q \wedge \neg r$$





Model Checking using Interpolants

Inductive invariant (I_s) for $\forall \square p$

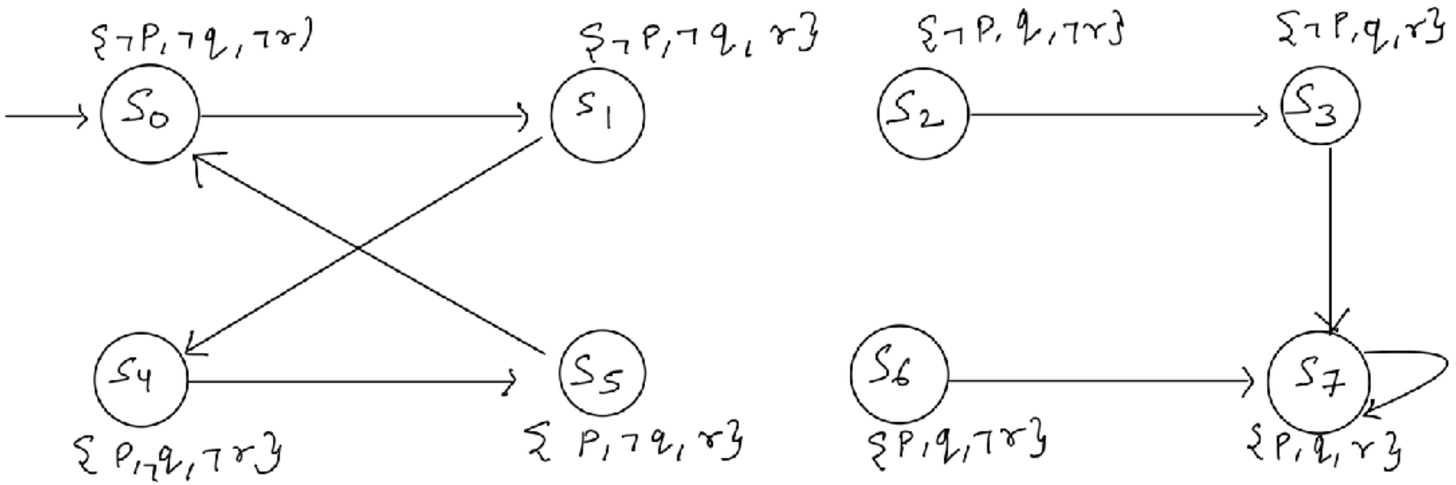
1. I_s must include the set of initial states, $I \subseteq I_s$
2. I_s must not include a state that is labeled with $\neg p$, $\forall s \in I_s, s \vDash p$
3. I_s must be closed under transition relation, post-image(I_s) $\subseteq I_s$ holds.

Post-image (Q) = $\{ s' \mid \exists s \in Q \,.\, T(s, s') \}$

If there exists a inductive invariant for $\forall \square P$, then $M \vDash \forall \square p$

Model Checking using Interpolants

Can you use interplants to compute inductive invariants?



Let us consider the above example: Look carefully at the labelling function.

$F = \forall \square \neg(p \wedge q \wedge r)$.

Only Bad state is S_7

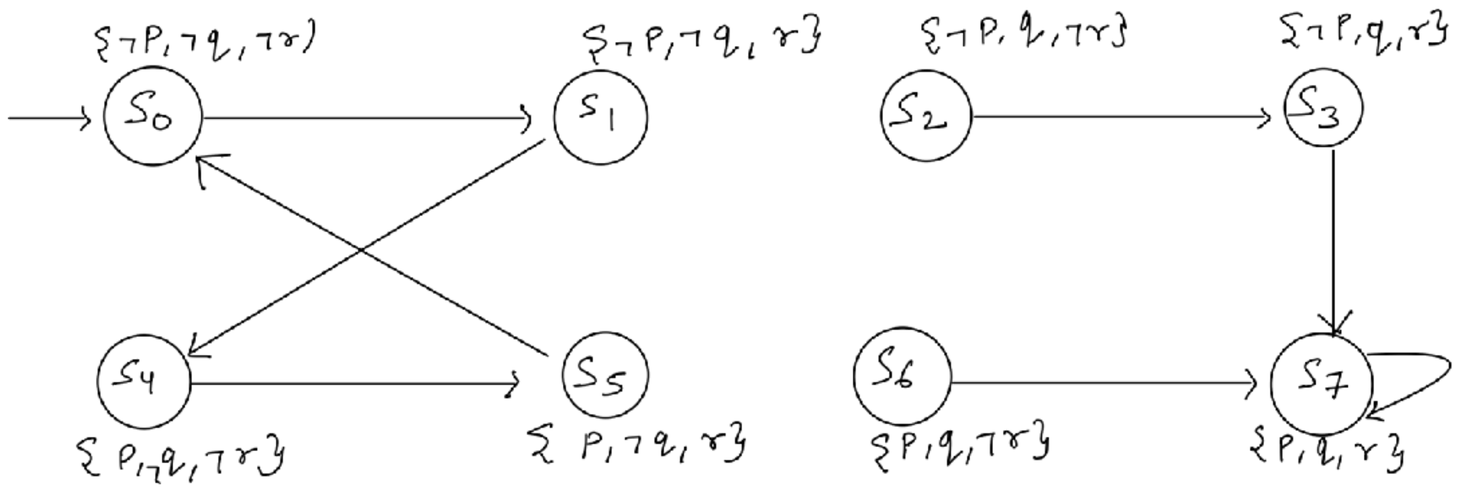
Reachability analysis — can we reach to state where $p \wedge q \wedge r$ holds initial states?

1. Does initial state is a bad state?

$$CheckSAT\{s_o \wedge p_o\}$$

$$(\neg p_o \wedge \neg q_o \wedge \neg r_o) \wedge (p_o \wedge q_o \wedge r_o)$$

UNSAT — good to go!



Let us consider the above example: Look carefully at the labelling function.

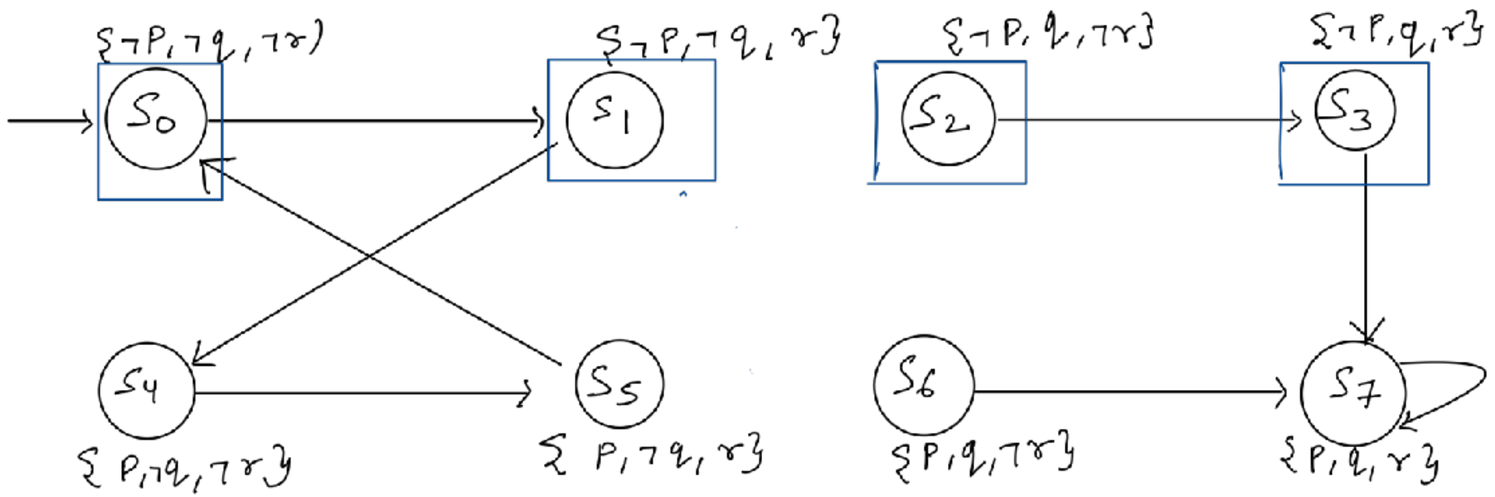
$F = \forall \square \neg(p \wedge q \wedge r)$.

Only Bad state is S_7

Reachability analysis — can we reach to state where $p \wedge q \wedge r$ holds initial states?

$$\underbrace{Q(s_o) \wedge T(s_o, s_1)}_{A} \wedge \underbrace{\bigwedge_{i=1}^{k-1} T(s_i, s_{i+1}) \wedge \bigvee_{i=1}^{k} p(s_i)}_{B} \qquad Q = \{s_o\} \quad \text{K} = 1$$

$$\underbrace{(\neg p_o \wedge \neg q_o \wedge \neg r_o) \wedge (\neg p_1 \wedge \neg q_1 \wedge r_1)}_{A} \; \underbrace{\wedge (p_1 \wedge q_1 \wedge r_1)}_{B} \qquad \text{UNSAT}$$



Let us consider the above example: Look carefully at the labelling function.

$F = \forall \Box \neg (p \wedge q \wedge r)$.

Only Bad state is S_7

Reachability analysis — can we reach to state where $p \wedge q \wedge r$ holds initial states?

$$\underbrace{(\neg p_o \wedge \neg q_o \wedge \neg r_o) \wedge (\neg p_1 \wedge \neg q_1 \wedge r_1)}_{A} \quad \underbrace{\wedge (p_1 \wedge q_1 \wedge r_1)}_{B} \qquad \text{UNSAT}$$

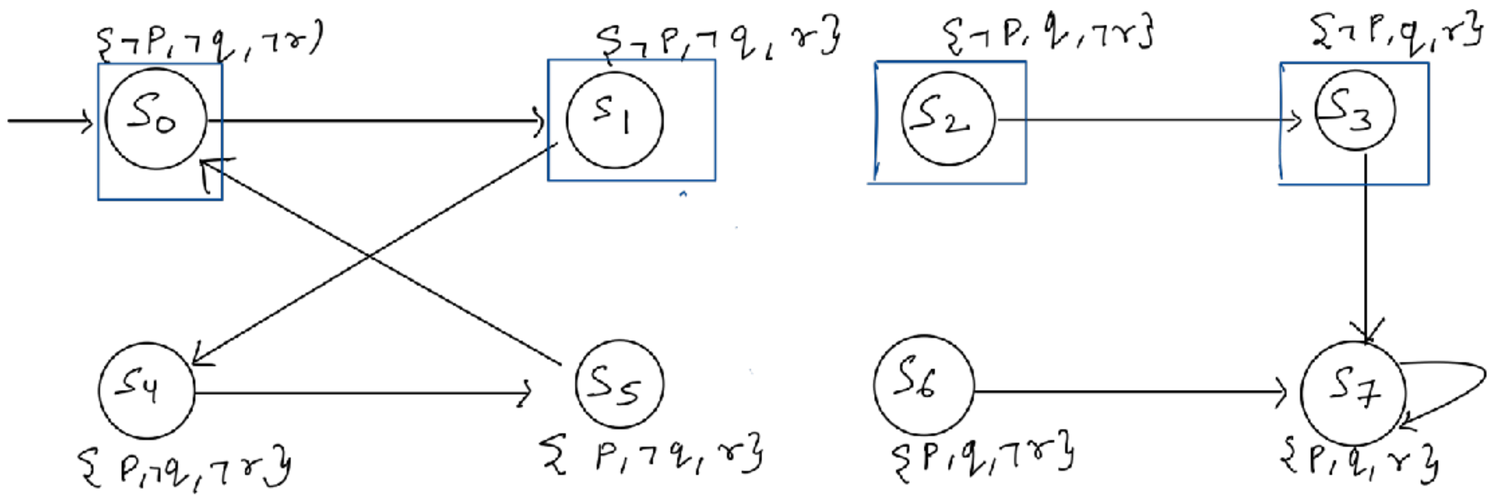
Interpolant := $\neg p_1$

$I_S = \{s_o, s_1, s_2, s_3\}$

$I_s : \{s \mid I \in L(s)\}$

$Q = Q \cup I_s$

Check the reachability with Over-approximate set



Let us consider the above example: Look carefully at the labelling function.

$F = \forall \Box \neg (p \wedge q \wedge r)$.

Only Bad state is S_7

Reachability analysis — can we reach to state where $p \wedge q \wedge r$ holds initial states?

$$\underbrace{(\neg p_o \wedge \neg q_o \wedge \neg r_o) \wedge (\neg p_1 \wedge \neg q_1 \wedge r_1)}_{A} \quad \underbrace{\wedge (p_1 \wedge q_1 \wedge r_1)}_{B} \qquad \text{UNSAT}$$

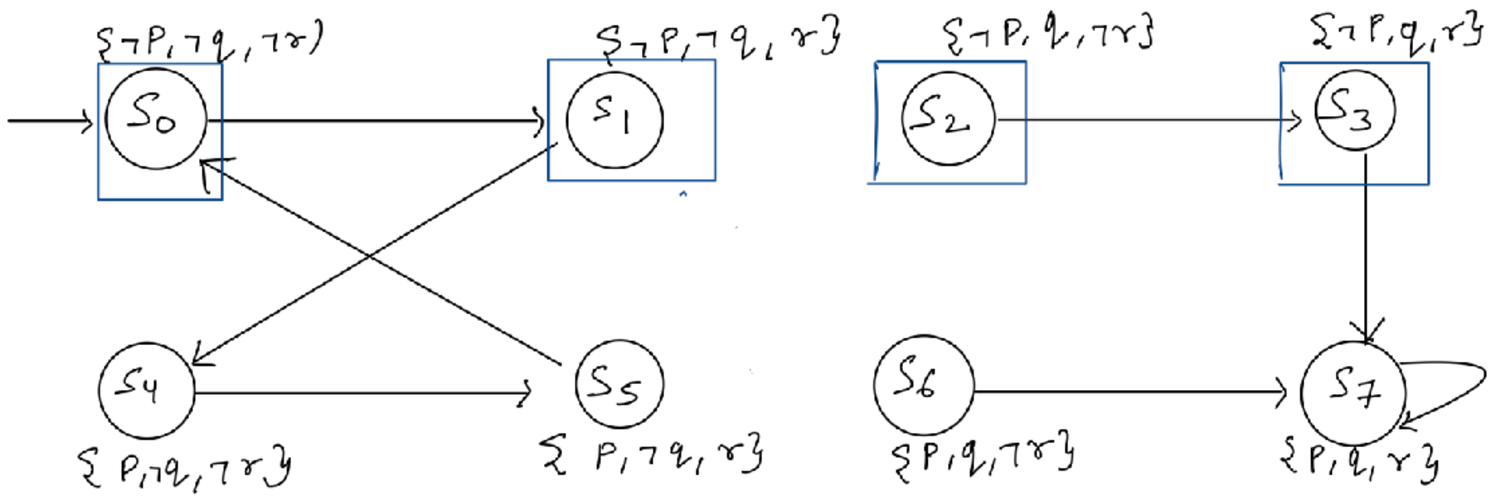
Interpolant := $\neg p_1$

$I_S = \{s_o, s_1, s_2, s_3\}$

$I_s : \{s \mid I \in L(s)\}$

$Q = Q \cup I_s$

Check the reachability with Over-approximate set



Let us consider the above example: Look carefully at the labelling function.

$F = \forall \Box \neg (p \wedge q \wedge r)$.

Only Bad state is S_7

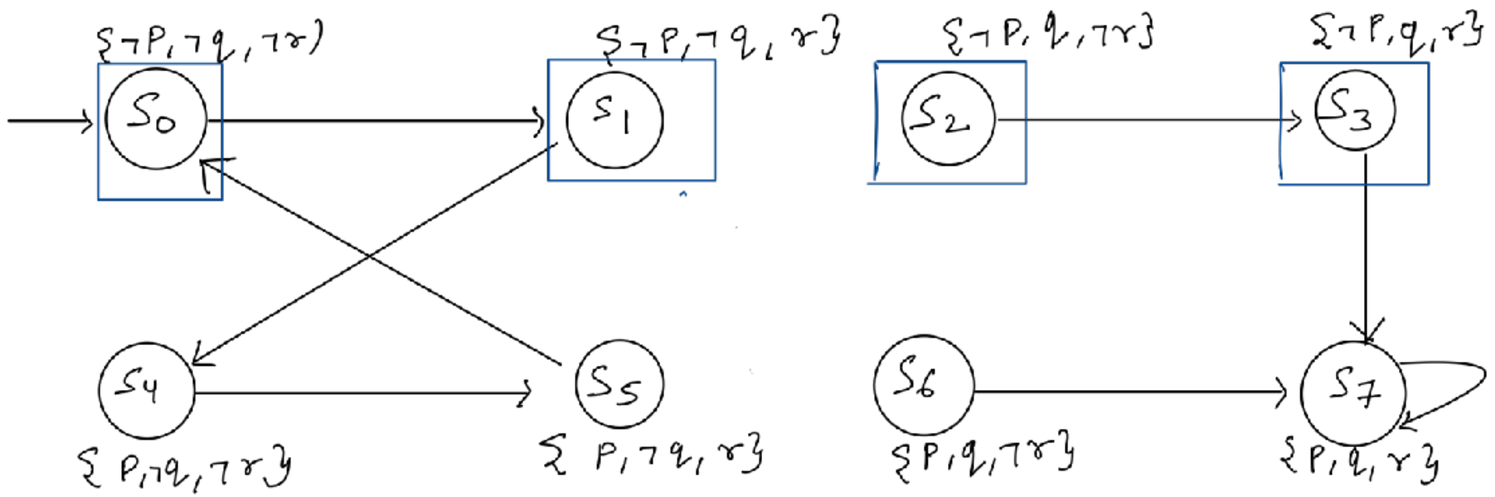
Reachability analysis — can we reach to state where $p \wedge q \wedge r$ holds initial states?

$Q = Q \cup I_s$

Check the reachability with Over-approximate set

$Q = \{s_o, s_1, s_2, s_3\}$

Is Q an inductive invariant ? No! post-image$(s_1) \notin Q$



Let us consider the above example: Look carefully at the labelling function.

$F = \forall \Box \neg (p \wedge q \wedge r)$.

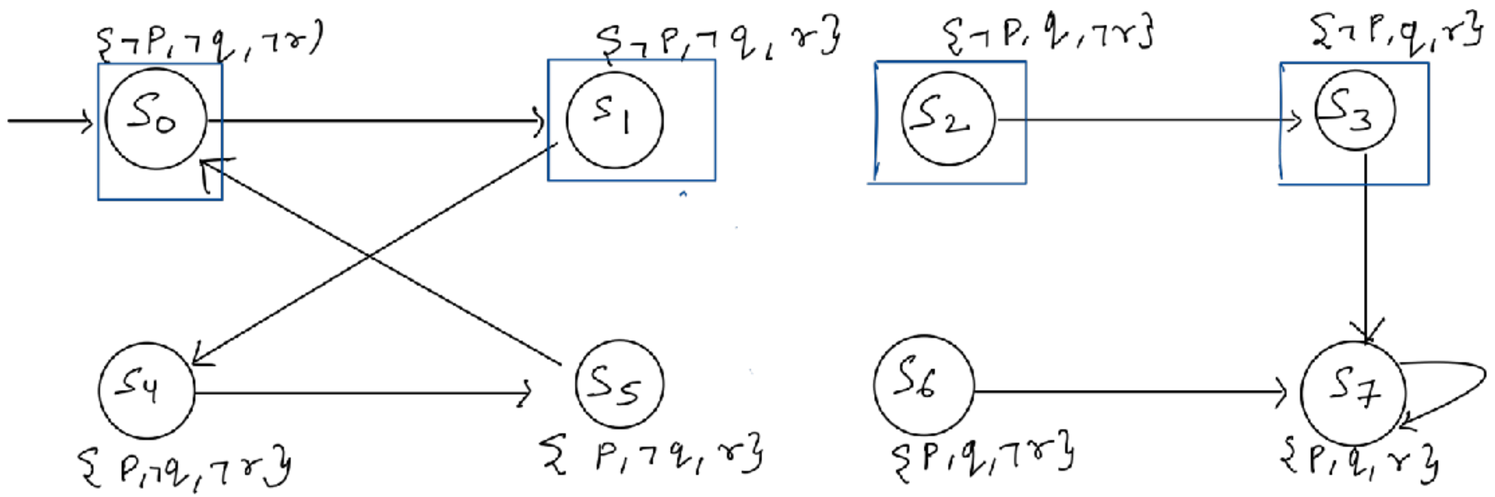
Only Bad state is S_7

Reachability analysis — can we reach to state where $p \wedge q \wedge r$ holds initial states?

$Q = \{s_o, s_1, s_2, s_3\}$

$$Q(s_o) \wedge T(s_o, s_1) \wedge \bigwedge_{i=1}^{k-1} T(s_i, s_{i+1}) \wedge \bigvee_{i=1}^{k} p(s_i)$$

A B



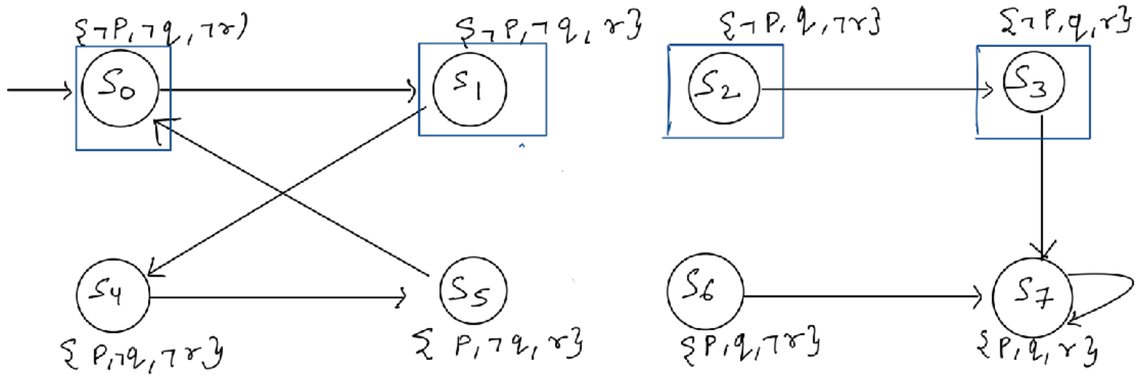
Let us consider the above example: Look carefully at the labelling function.

$F = \forall \Box \neg (p \wedge q \wedge r)$.

Only Bad state is S_7

Reachability analysis — can we reach to state where $p \wedge q \wedge r$ holds initial states?

$$Q = \{s_o, s_1, s_2, s_3\} \quad \underbrace{\bigvee_{\forall s \in Q} \{Q(s_o) \wedge T(s_o, s_1)\}}_{\text{A}} \wedge \underbrace{\bigwedge_{i=1}^{k-1} T(s_i, s_{i+1}) \wedge \bigvee_{i=1}^{k} p(s_i)}_{\text{B}}$$



Reachability analysis — can we reach to state where $p \wedge q \wedge r$ holds initial states?

$Q = \{s_o, s_1, s_2, s_3\}$

$$\bigvee_{\forall s \in Q} \{Q(s_o) \wedge T(s_o, s_1)\} \wedge \bigwedge_{i=1}^{k-1} T(s_i, s_{i+1}) \wedge \bigvee_{i=1}^{k} p(s_i) \qquad \text{K} = \mathbf{1}$$

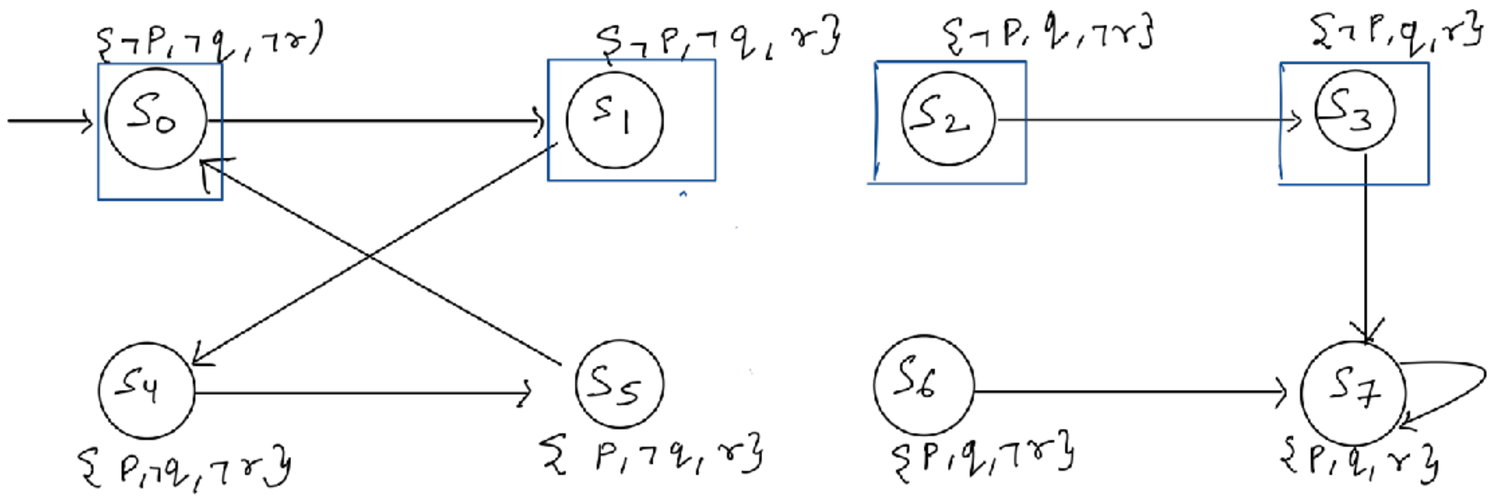
A B

$$A = [(\neg p_o \wedge \neg q_o \wedge \neg r_o) \wedge (\neg p_1 \wedge \neg q_1 \wedge r_1)] \vee [(\neg p_o \wedge \neg q_o \wedge r_o) \wedge (p_1 \wedge \neg q_1 \wedge \neg r_1)]$$

$$\vee [(\neg p_o \wedge q_o \wedge \neg r_o) \wedge (\neg p_1 \wedge q_1 \wedge r_1)] \vee [(\neg p_o \wedge q_o \wedge r_o) \wedge (p_1 \wedge q_1 \wedge r_1)]$$

$$B = (p_1 \wedge q_1 \wedge r_1)$$

$A \wedge B$ is SAT.



Let us consider the above example: Look carefully at the labelling function.

$F = \forall \Box \neg (p \wedge q \wedge r)$. Only Bad state is S_7

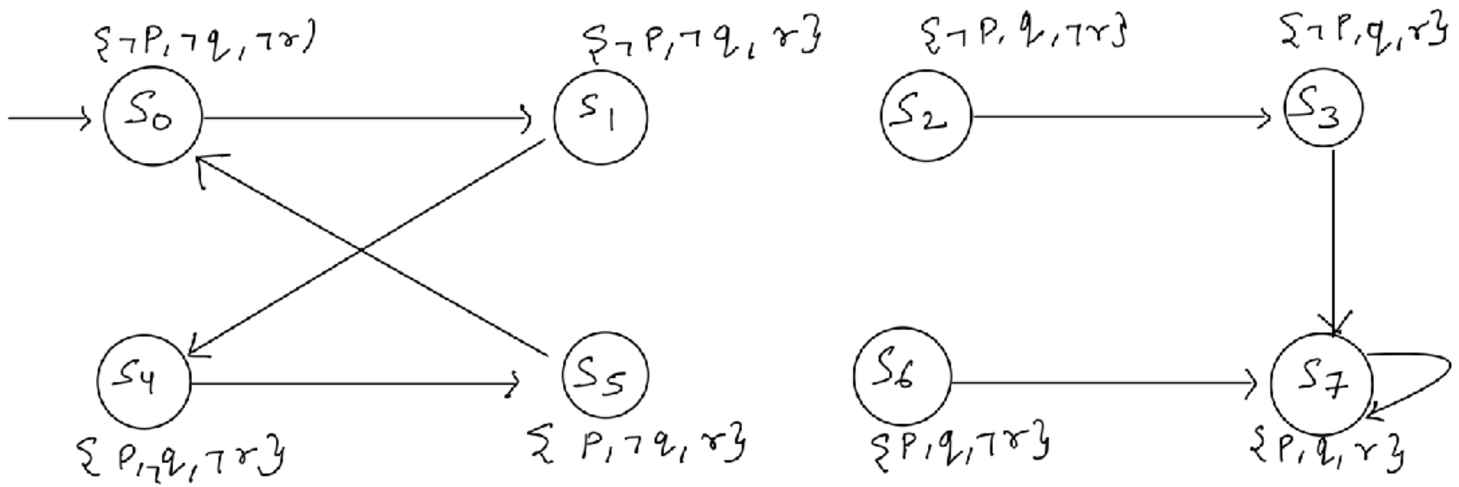
Reachability analysis — can we reach to state where $p \wedge q \wedge r$ holds initial states?

$$Q = \{s_o, s_1, s_2, s_3\} \qquad \bigvee_{\forall s \in Q} \{Q(s_o) \wedge T(s_o, s_1)\} \wedge \bigwedge_{i=1}^{k-1} T(s_i, s_{i+1}) \wedge \bigvee_{i=1}^{k} p(s_i) \qquad \text{K} = 1$$

A B

If $A \wedge B$ is SAT, check if $Q = I$ Q = I, then Return counter-example.

Else, increase k to build trust!



Let us consider the above example: Look carefully at the labelling function.

$F = \forall \Box \neg (p \wedge q \wedge r)$.

Only Bad state is S_7

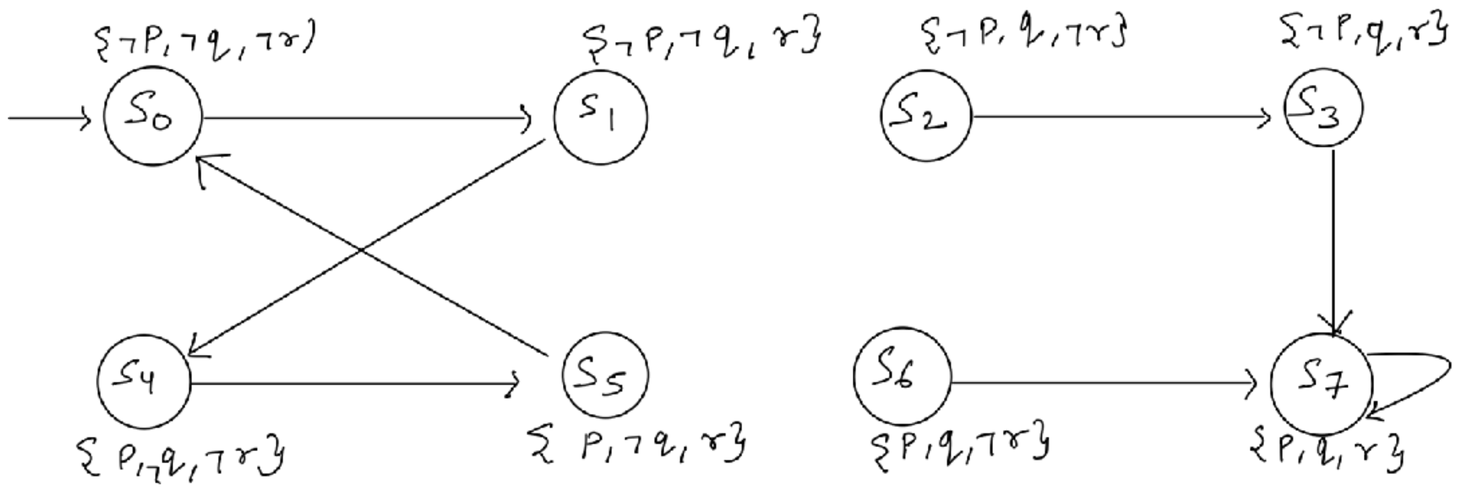
Reachability analysis — can we reach to state where $p \wedge q \wedge r$ holds initial states?

$$Q(s_o) \wedge T(s_o, s_1) \wedge \bigwedge_{i=1}^{k-1} T(s_i, s_{i+1}) \wedge \bigvee_{i=1}^{k} p(s_i) \qquad Q = \{s_o\} \quad \text{K} = 2$$

A B

$$(\neg p_o \wedge \neg q_o \wedge \neg r_o) \wedge (\neg p_1 \wedge \neg q_1 \wedge r_1) \quad \wedge (\neg p_1 \wedge \neg q_1 \wedge r_1 \wedge p_2 \wedge \neg q_2 \wedge \neg r_2) \wedge [(p_1 \wedge q_1 \wedge r_1) \vee (p_2 \wedge q_2 \wedge r_2)]$$

A B UNSAT



Let us consider the above example: Look carefully at the labelling function.

$F = \forall \square \neg(p \wedge q \wedge r)$.

Only Bad state is S_7

Reachability analysis — can we reach to state where $p \wedge q \wedge r$ holds initial states?

$$\underbrace{Q(s_o) \wedge T(s_o, s_1)}_{A} \wedge \underbrace{\bigwedge_{i=1}^{k-1} T(s_i, s_{i+1}) \wedge \bigvee_{i=1}^{k} p(s_i)}_{B}$$

$Q = \{s_o\}$ K = 2

UNSAT

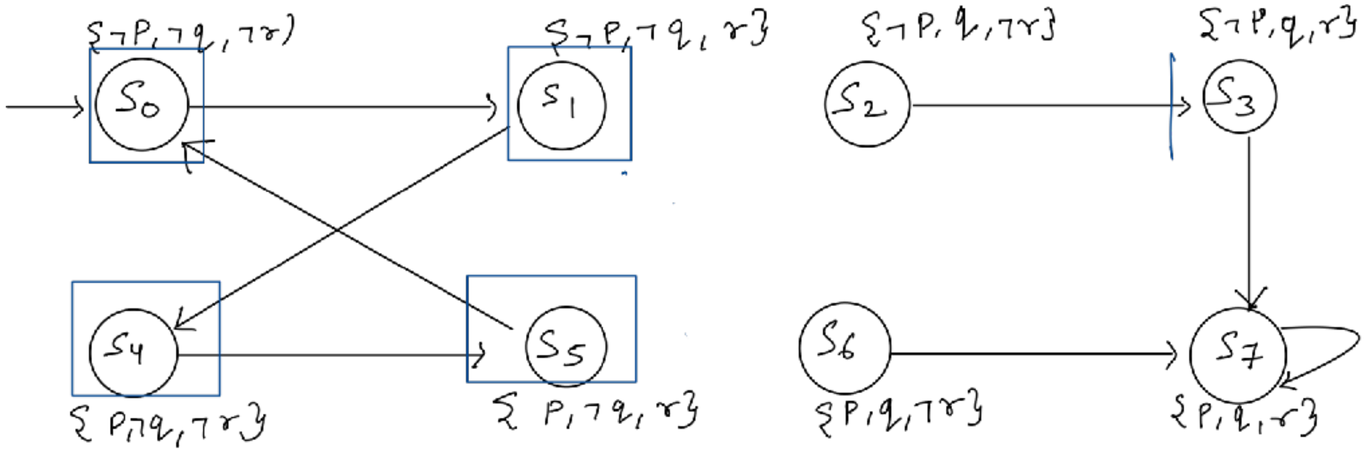
Interpolant := $\neg q_1$

$I_S = \{s_o, s_1, s_4, s_5\}$

$I_s : \{s \mid I \in L(s)\}$

$Q = Q \cup I_s$

Check the reachability with Over-approximate set



Let us consider the above example: Look carefully at the labelling function.

$F = \forall \Box \neg (p \wedge q \wedge r)$.

Only Bad state is S_7

Reachability analysis — can we reach to state where $p \wedge q \wedge r$ holds initial states?

$$\underbrace{Q(s_o) \wedge T(s_o, s_1)}_{A} \wedge \underbrace{\bigwedge_{i=1}^{k-1} T(s_i, s_{i+1}) \wedge \bigvee_{i=1}^{k} p(s_i)}_{B}$$

$Q = \{s_o\}$ K = 2

UNSAT

Interpolant := $\neg q_1$

$I_S = \{s_o, s_1, s_4, s_5\}$

$I_s : \{s \mid I \in L(s)\}$

$Q = Q \cup I_s$

Q is inductive invariant!!!

$M \models F$

Model Checking using Interpolants

General idea:

1. Perform BMC

2. If BMC is UNSAT:

Iteratively compute and refine an over-approximation of states reachable in K steps.

Compute Interpolant as over-approximation.
If interpolant is inductive
Return True.
else
use interpolant to over-approximate.

3. If BMC is SAT:

Check if over-approximation is same as initial states
otherwise increase K.

procedure *CraigReachability*(model M, $p \in AP$)
  if $S_0 \wedge \neg p$ is SAT **return** "$M \not\models$ **AG** p";
  $k := 1$;
  $Q := S_0$;
  while *true* **do**
    $A := Q(s_0) \wedge R(s_0, s_1)$;
    $B := \bigwedge_{i=1}^{k-1} R(s_i, s_{i+1}) \wedge \bigvee_{i=1}^{k} \neg p(s_i)$;
    if $A \wedge B$ is SAT **then**
      if $Q = S_0$ **then return** "$M \not\models$ **AG** p";
      Increase k
      $Q := S_0$
    else
      compute interpolant I for A and B
      if $I \subseteq Q$ **then return** "$M \models$ **AG** p";
      $Q := Q \cup I$
    end if
  end while
end procedure

Model Checking using Interpolants

Can you use interplants to compute inductive invariants?

1. Constructs an over-approximation of the reachable states
2. Terminates when it finds an inductive invariant or a counterexample

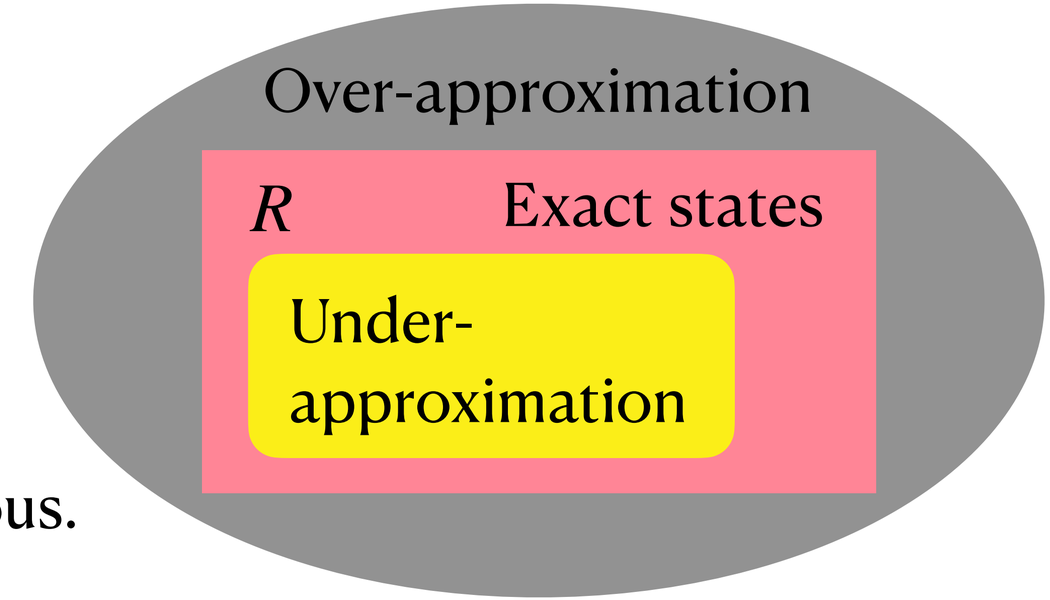
Actual reachable set: R

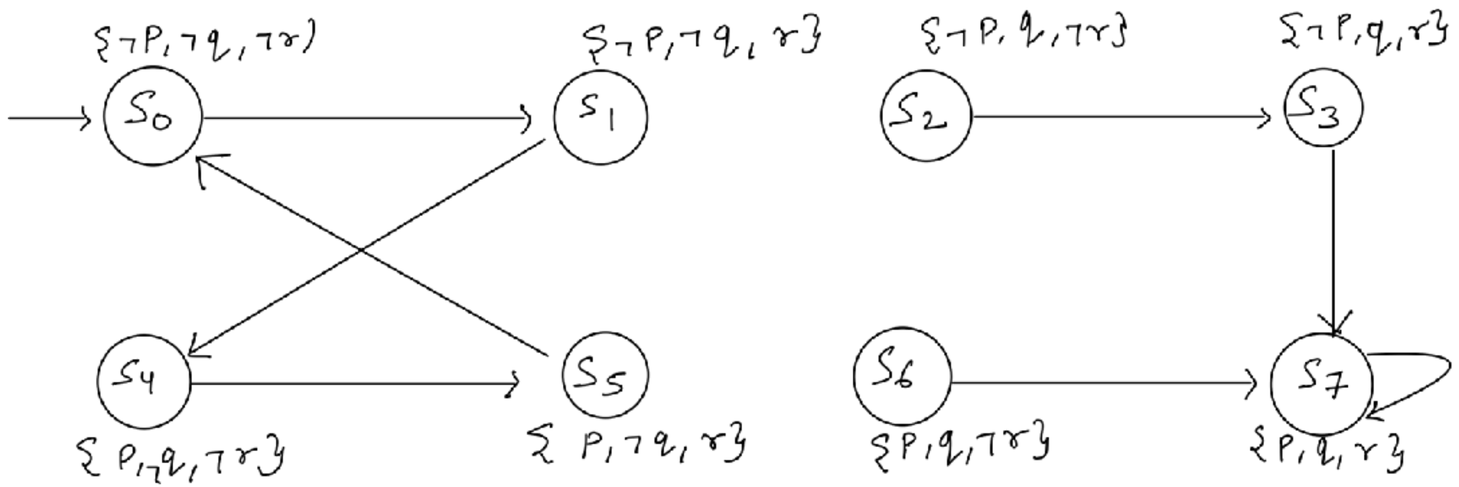
Over-approximation (O_p): $R \rightarrow O_p$

1. Proofs on over-approximation holds.
2. Counterexample can be spurious.

Under-approximation (U_p): $U_p \rightarrow R$

1. Proofs on over-approximation can be spurious.
2. Counterexample holds





$$T(p, q, r, p', q', r') \equiv (q' \leftrightarrow q) \wedge (r' \leftrightarrow (p \vee q \vee \neg r)) \wedge \left(p' \leftrightarrow ((p \oplus r) \vee (q \wedge r))\right).$$