CNF

For more efficient algorithms we need to assume the formulas have some particular boolean structure.

We will now apply DPLL to formulas in CNF.

A quantified boolean formula F is in CNF, if it is either $\bot$, or $\top$, or has the form $\exists\forall_1 p_1 \ldots \exists\forall_n p_n (C_1 \wedge \ldots \wedge C_m)$, where $C_1, \ldots, C_m$ are clauses.

Example:
$\forall p \exists q \exists s ((\neg p \vee s \vee q) \wedge (s \vee \neg q) \wedge \neg s))$

CNF

For more efficient algorithms we need to assume the formulas have some particular boolean structure.

We will now apply DPLL to formulas in CNF.

A quantified boolean formula F is in CNF, if it is either $\bot$, or $\top$, or has the form $\exists\forall_1 p_1 \ldots \exists\forall_n p_n (C_1 \wedge \ldots \wedge C_m)$, where $C_1, \ldots, C_m$ are clauses.

Example:
$\forall p \exists q \exists s ((\neg p \vee s \vee q) \wedge (s \vee \neg q) \wedge \neg s))$

CNF

For more efficient algorithms we need to assume the formulas have some particular boolean structure.

We will now apply DPLL to formulas in CNF.

A quantified boolean formula F is in CNF, if it is either $\bot$, or $\top$, or has the form $\exists\forall_1 p_1 \ldots \exists\forall_n p_n (C_1 \wedge \ldots \wedge C_m)$, where $C_1, \ldots, C_m$ are clauses.

Example:
$\forall p \exists q \exists s ((\neg p \vee s \vee q) \wedge (s \vee \neg q) \wedge \neg s))$

CNF rules

Prenexing rules + propositional CNF rules:

$$
\begin{aligned}
F \leftrightarrow G &\Rightarrow (\neg F \vee G) \wedge (\neg G \vee F), \\
F \rightarrow G &\Rightarrow \neg F \vee G, \\
\neg(F \wedge G) &\Rightarrow \neg F \vee \neg G, \\
\neg(F \vee G) &\Rightarrow \neg F \wedge \neg G, \\
\neg\neg F &\Rightarrow F, \\
(F_1 \wedge \ldots \wedge F_m) \vee G_1 \vee \ldots \vee G_n &\Rightarrow (F_1 \vee G_1 \vee \ldots \vee G_n) \wedge \\
&\qquad \ldots \qquad \wedge \\
&\quad (F_m \vee G_1 \vee \ldots \vee G_n).
\end{aligned}
$$

Unit Propagation (DPLL)

Input of DPLL:

- Q: quantifier sequence $\exists\forall_1 p_1 \ldots\ldots \exists\forall_n p_n$
- S: a set of clauses

Main simplification – unit propagation with respect to Q, S:
if S contains a unit clause, i.e. a clause consisting of one literal L of the form p or $\neg p$ then

- if Q contains $\exists p$ or p does not occur in Q
  1. remove from S every clause of the form $L \vee C'$;
  2. replace in S every clause of the form $\overline{L} \vee C'$ by the clause C'.
- if Q contains $\forall p$, then replace S by the set $\{\square\}$;

Unit Propagation (DPLL)

Input of DPLL:

- Q: quantifier sequence $\exists\forall_1 p_1 \ldots\ldots \exists\forall_n p_n$
- S: a set of clauses

Main simplification – unit propagation with respect to Q, S:
if S contains a unit clause, i.e. a clause consisting of one literal L of the form p or $\neg p$ then

- if Q contains $\exists p$ or p does not occur in Q
  1. remove from S every clause of the form $L \vee C'$;
  2. replace in S every clause of the form $\overline{L} \vee C'$ by the clause C'.
- if Q contains $\forall p$, then replace S by the set $\{\Box\}$;

Unit Propagation (DPLL)

Input of DPLL:

- Q: quantifier sequence $\exists\forall_1 p_1 \ldots\ldots \exists\forall_n p_n$
- S: a set of clauses

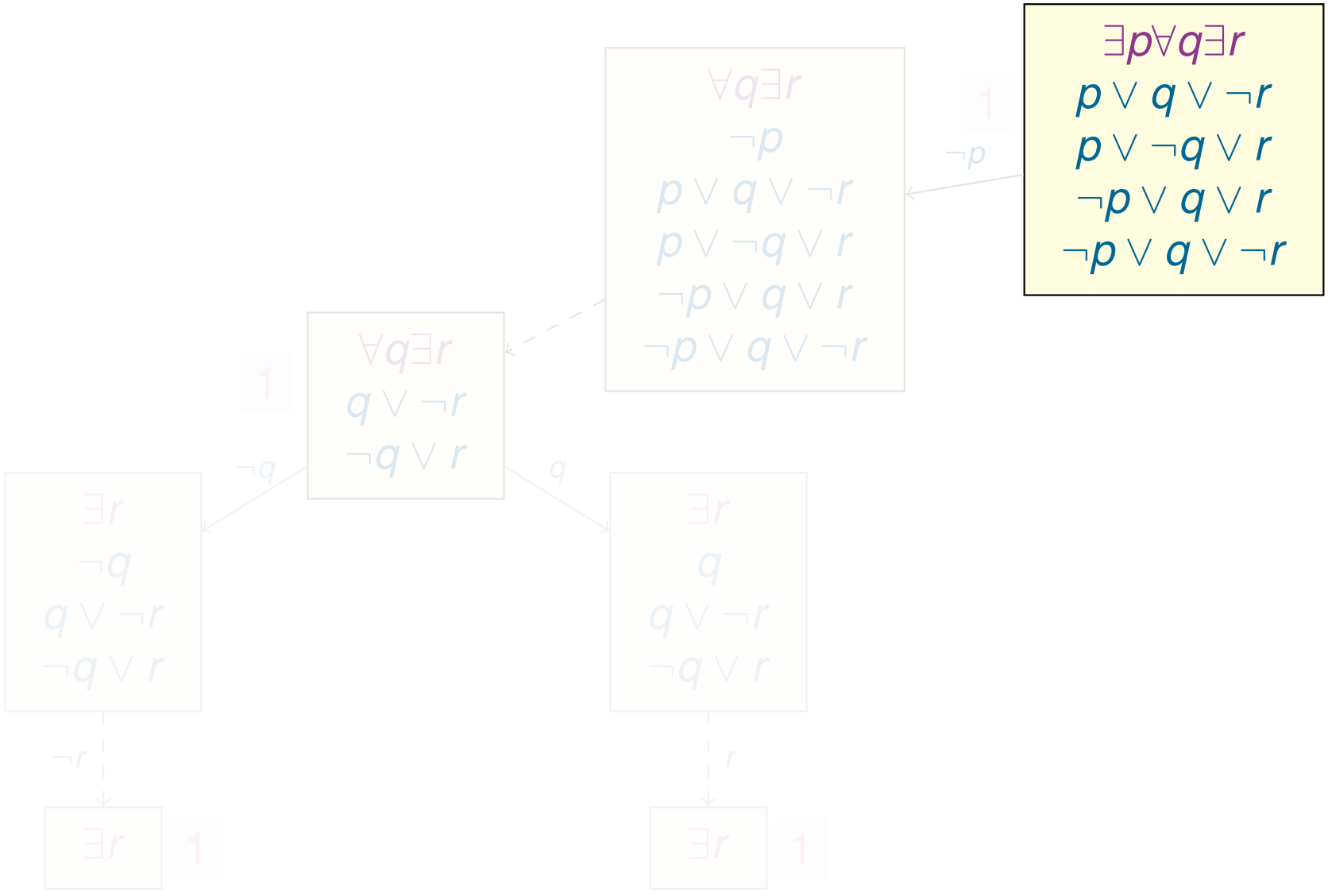
Main simplification – unit propagation with respect to Q, S:
if S contains a unit clause, i.e. a clause consisting of one literal L of the form p or $\neg p$ then

- if Q contains $\exists p$ or p does not occur in Q
  1. remove from S every clause of the form $L \vee C'$;
  2. replace in S every clause of the form $\overline{L} \vee C'$ by the clause C'.
- if Q contains $\forall p$, then replace S by the set $\{\square\}$;

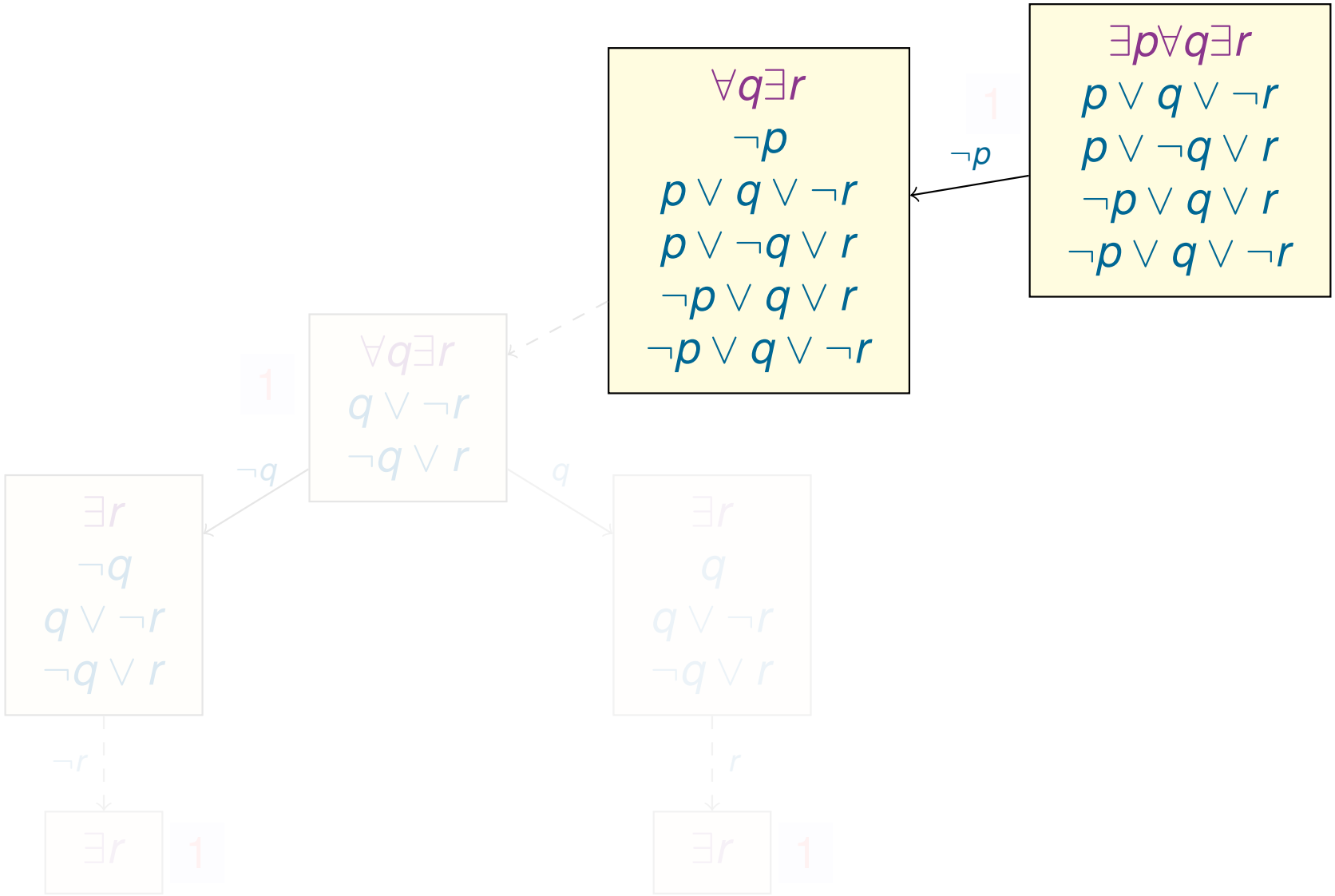
DPLL algorithm

procedure $DPLL(Q, S)$
input: quantifier sequence $Q = \exists\forall_1 p_1 \ldots \exists\forall_n p_n$, set of clauses S
output: 0 or 1
parameters: function *select_signed_atom*
begin
  $S := unit\_propagate(Q, S)$
  if S is empty **then return** 1
  if S contains $\square$ **then return** 0
  $(p, b) := select\_signed\_atom(Q, S)$
  Let Q' be obtained from Q by deleting $\exists\forall p$ from its outermost prefix
  if $b = 0$ **then** $L := \neg p$
    else $L := p$
  case $(DPLL(Q', S \cup \{L\}),\ \exists\forall)$ **of**
    $(0, \forall) \Rightarrow$ **return** 0
    $(0, \exists) \Rightarrow$ **return** $DPLL(Q', S \cup \{\overline{L}\})$
    $(1, \forall) \Rightarrow$ **return** $DPLL(Q', S \cup \{\overline{L}\})$
    $(1, \exists) \Rightarrow$ **return** 1
end

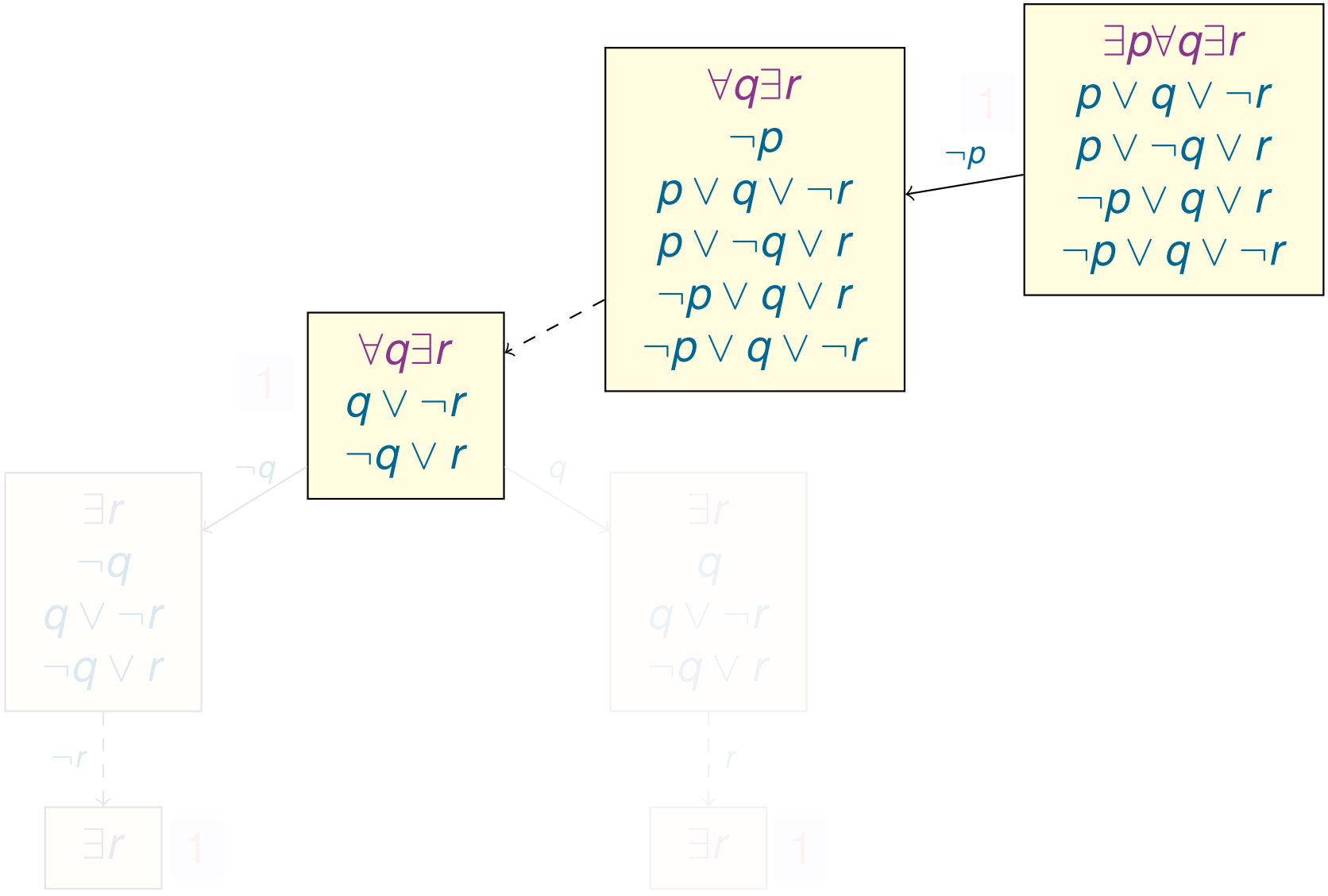
Example


∃p∀q∃r
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
1
¬p
∀q∃r
¬p
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
1
¬q
q
∀q∃r
q ∨ ¬r
¬q ∨ r
∃r
¬q
q ∨ ¬r
¬q ∨ r
∃r
q
q ∨ ¬r
¬q ∨ r
¬r
r
∃r
1
∃r
1

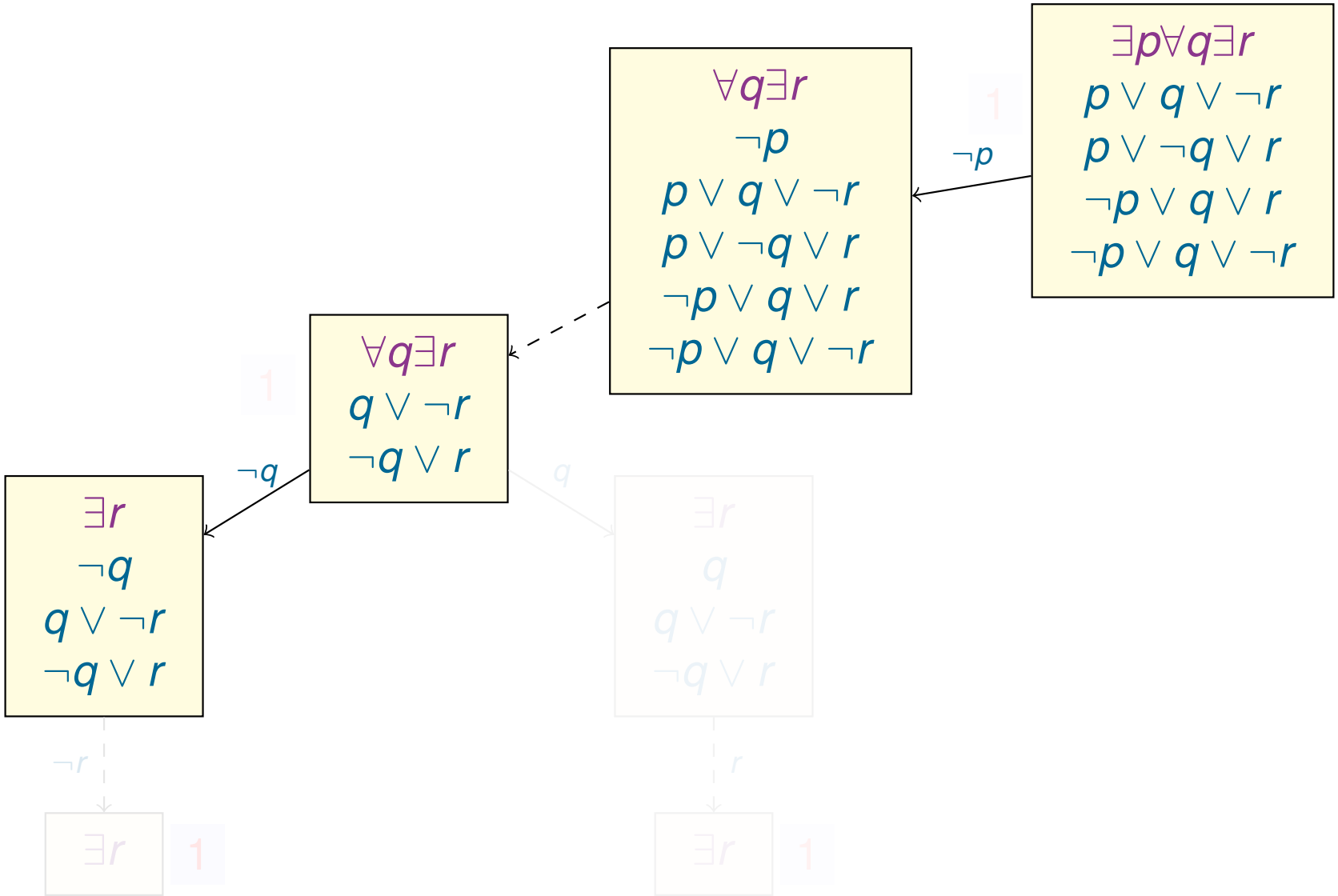
Example


∃p∀q∃r
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
1
¬p
∀q∃r
¬p
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
1
¬q
q
∀q∃r
q ∨ ¬r
¬q ∨ r
∃r
¬q
q ∨ ¬r
¬q ∨ r
∃r
q
q ∨ ¬r
¬q ∨ r
¬r
r
∃r
1
∃r
1

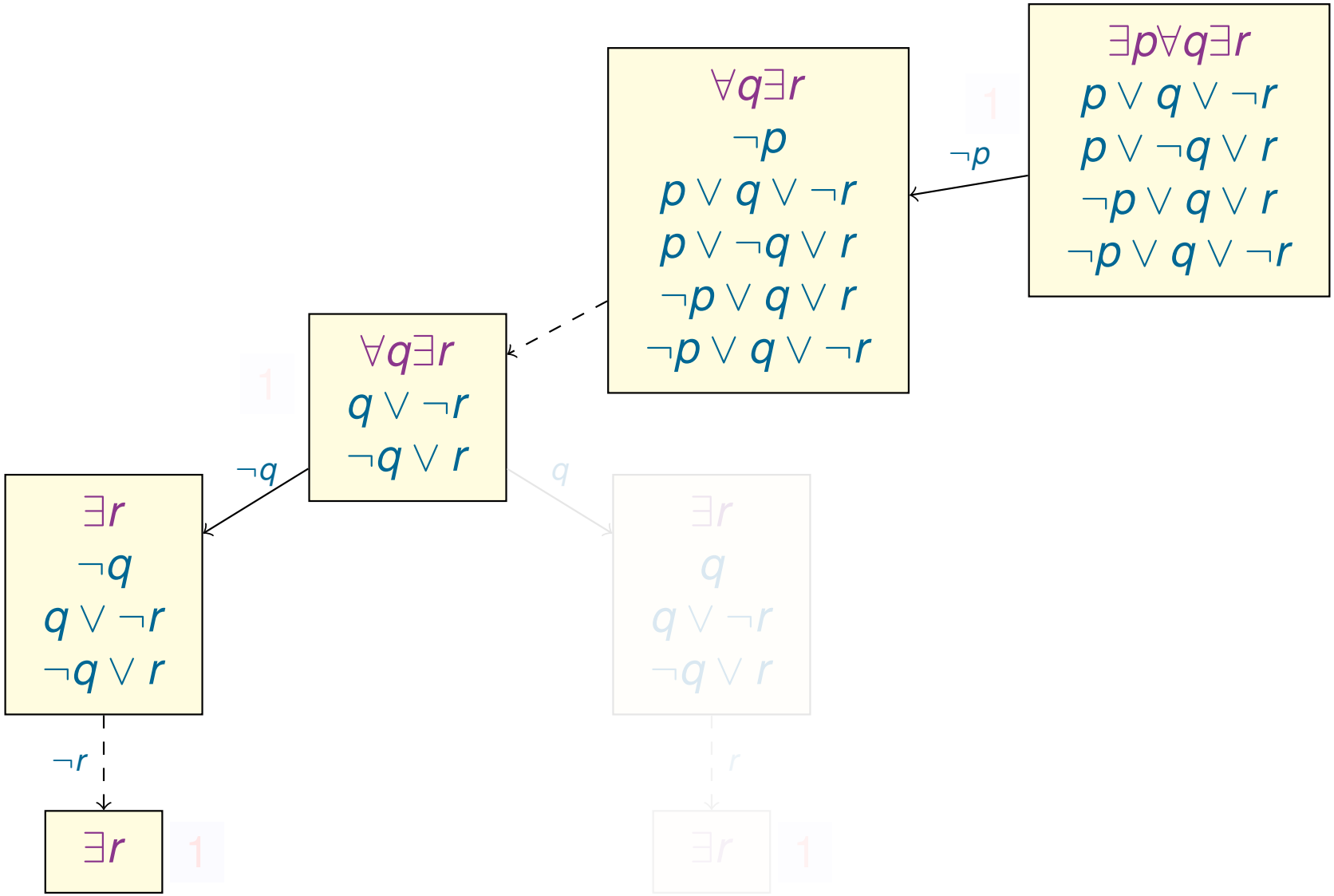
Example


∃p∀q∃r
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
¬p
∀q∃r
¬p
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
1
∀q∃r
q ∨ ¬r
¬q ∨ r
¬q
q
∃r
¬q
q ∨ ¬r
¬q ∨ r
∃r
q
q ∨ ¬r
¬q ∨ r
¬r
r
∃r
1
∃r
1

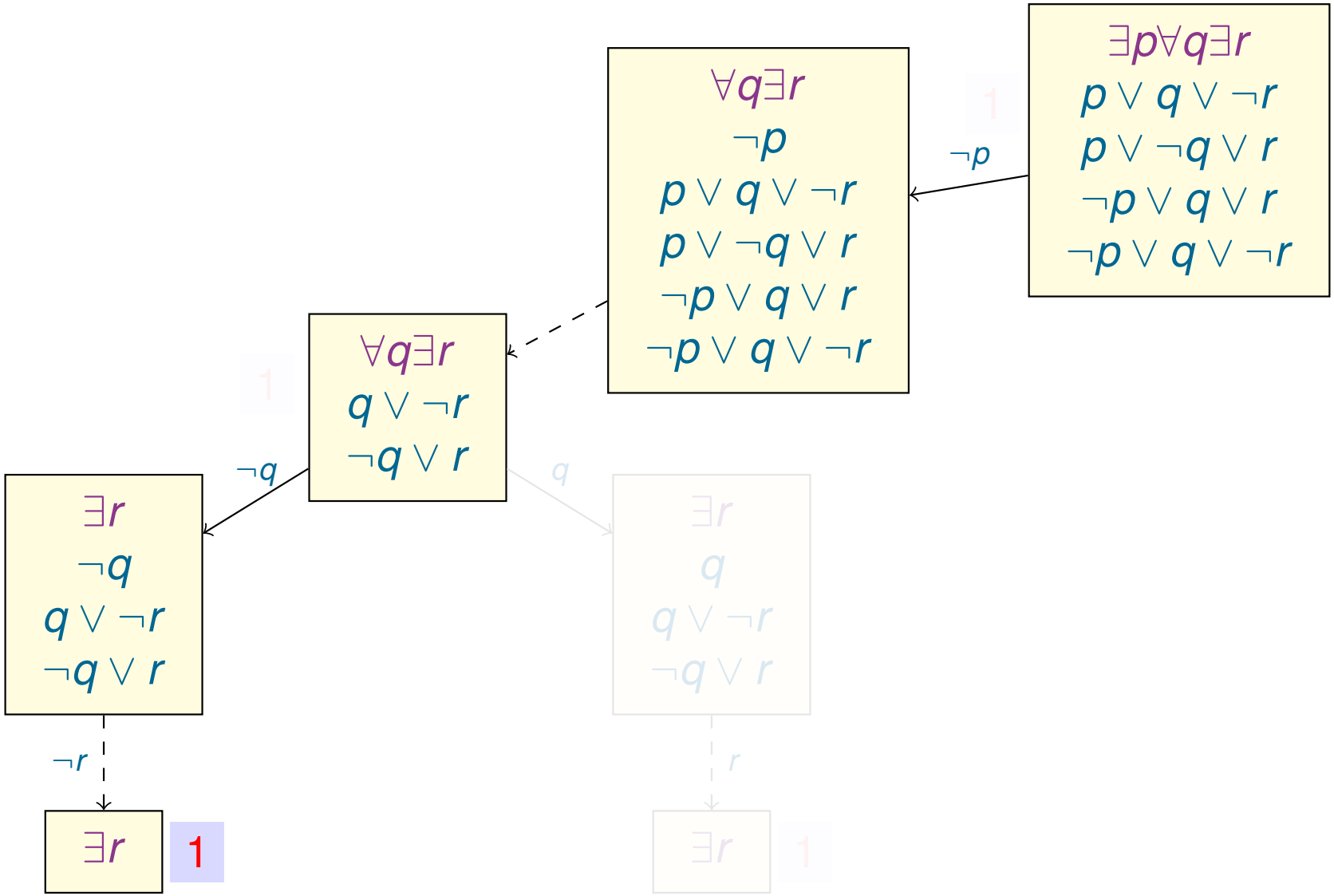
Example


$\exists p \forall q \exists r$
$p \vee q \vee \neg r$
$p \vee \neg q \vee r$
$\neg p \vee q \vee r$
$\neg p \vee q \vee \neg r$
$\neg p$
$\forall q \exists r$
$\neg p$
$p \vee q \vee \neg r$
$p \vee \neg q \vee r$
$\neg p \vee q \vee r$
$\neg p \vee q \vee \neg r$
$\forall q \exists r$
$q \vee \neg r$
$\neg q \vee r$
$\neg q$
$\exists r$
$\neg q$
$q \vee \neg r$
$\neg q \vee r$

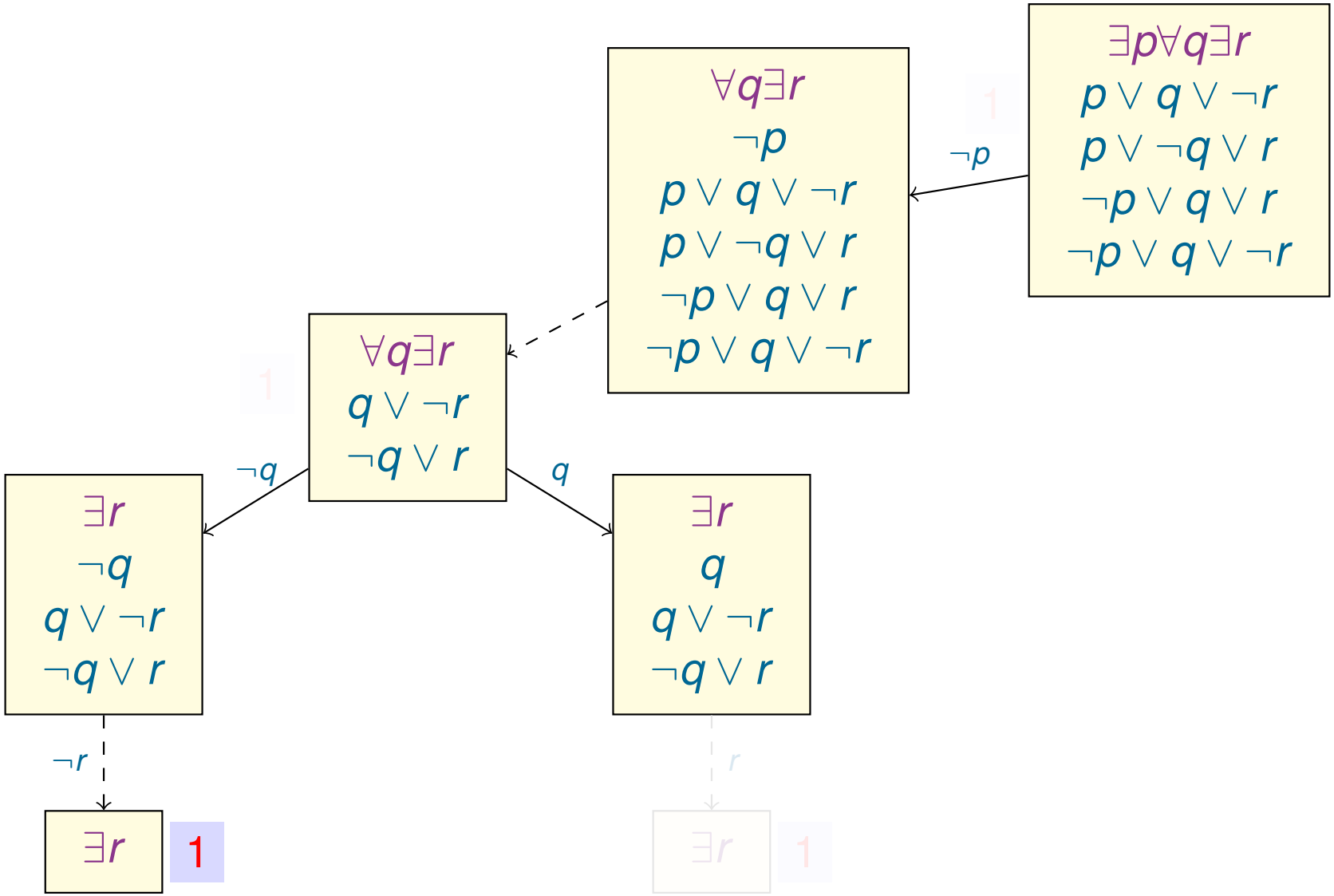
Example


∃p∀q∃r
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
¬p
∀q∃r
¬p
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
1
¬q
∀q∃r
q ∨ ¬r
¬q ∨ r
q
∃r
¬q
q ∨ ¬r
¬q ∨ r
∃r
q
q ∨ ¬r
¬q ∨ r
¬r
r
∃r
1
∃r
1

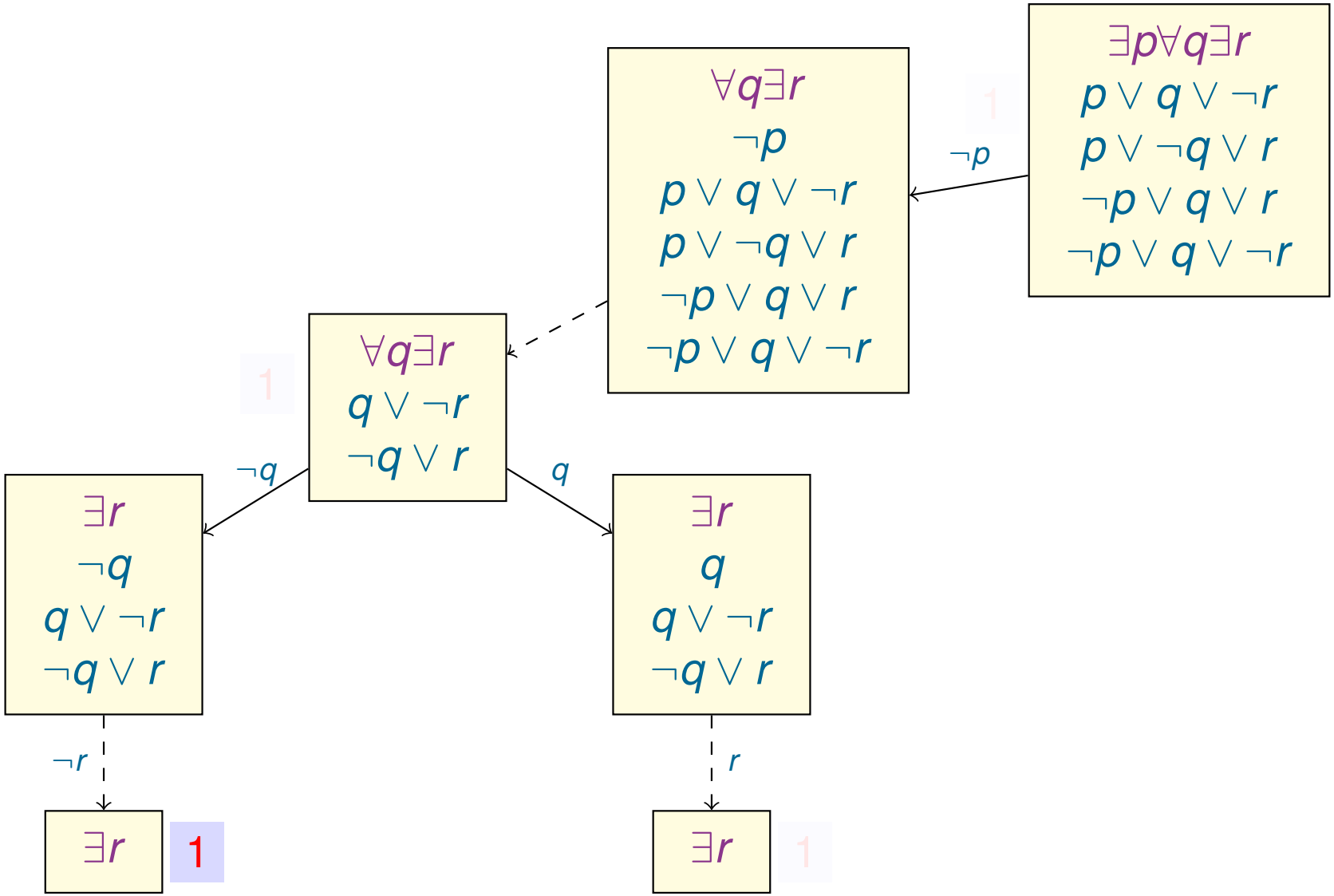
Example


∃p∀q∃r
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
¬p
∀q∃r
¬p
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
1
∀q∃r
q ∨ ¬r
¬q ∨ r
¬q
q
∃r
¬q
q ∨ ¬r
¬q ∨ r
∃r
q
q ∨ ¬r
¬q ∨ r
¬r
r
∃r
1
∃r
1

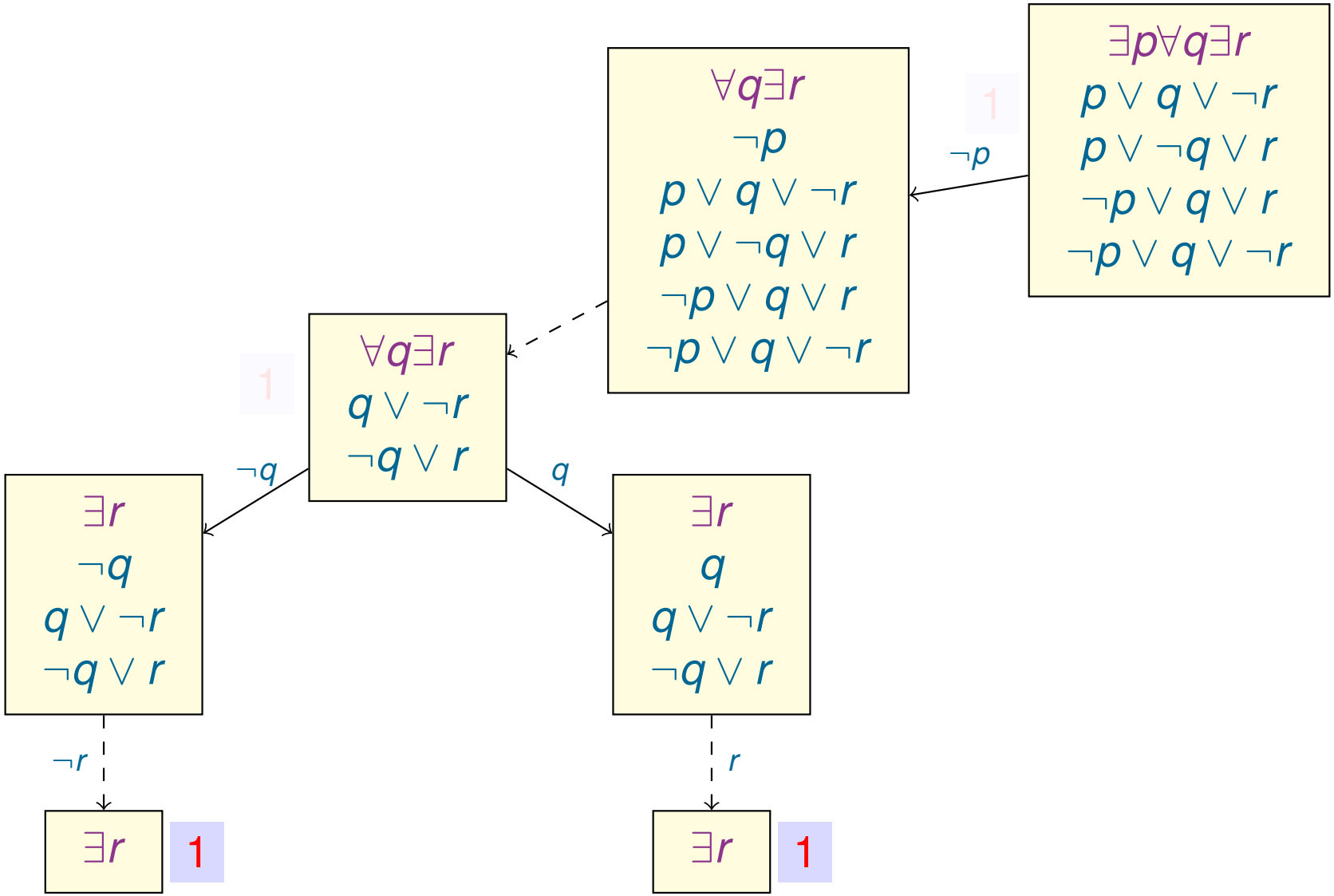
Example


∃p∀q∃r
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
¬p
∀q∃r
¬p
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
1
∀q∃r
q ∨ ¬r
¬q ∨ r
¬q
q
∃r
¬q
q ∨ ¬r
¬q ∨ r
∃r
q
q ∨ ¬r
¬q ∨ r
¬r
r
∃r
1
∃r
1

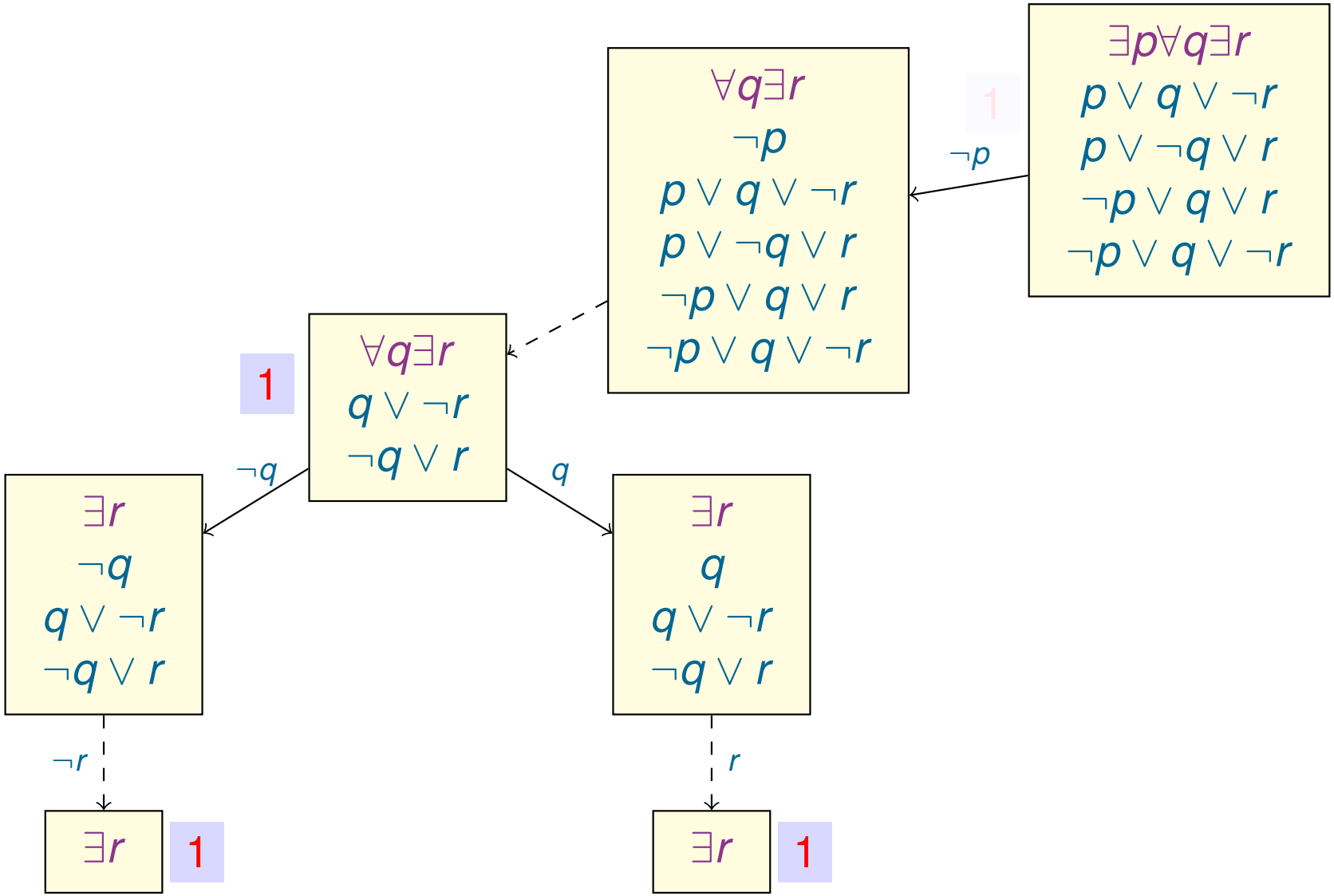
Example


∃p∀q∃r
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
¬p
∀q∃r
¬p
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
1
∀q∃r
q ∨ ¬r
¬q ∨ r
1
¬q
q
∃r
¬q
q ∨ ¬r
¬q ∨ r
∃r
q
q ∨ ¬r
¬q ∨ r
¬r
r
∃r
1
∃r
1

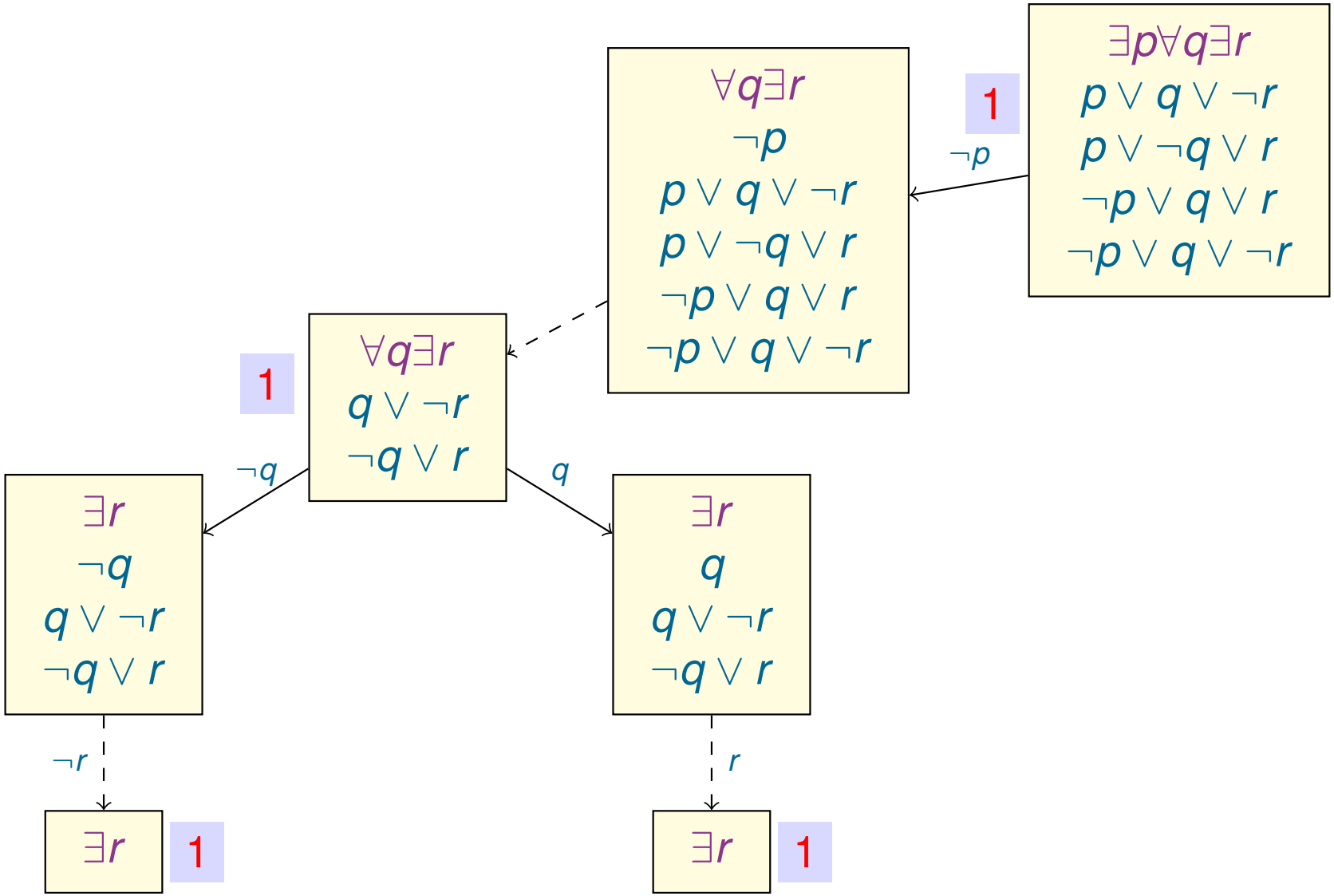
Example


∃p∀q∃r
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
¬p
1
∀q∃r
¬p
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
1
∀q∃r
q ∨ ¬r
¬q ∨ r
¬q
q
∃r
¬q
q ∨ ¬r
¬q ∨ r
∃r
q
q ∨ ¬r
¬q ∨ r
¬r
r
∃r
1
∃r
1

Example


∃p∀q∃r
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
1
¬p
∀q∃r
¬p
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
1
∀q∃r
q ∨ ¬r
¬q ∨ r
¬q
q
∃r
¬q
q ∨ ¬r
¬q ∨ r
∃r
q
q ∨ ¬r
¬q ∨ r
¬r
r
∃r
1
∃r
1

Example


∃p∀q∃r
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
1
¬p
∀q∃r
¬p
p ∨ q ∨ ¬r
p ∨ ¬q ∨ r
¬p ∨ q ∨ r
¬p ∨ q ∨ ¬r
1
¬q
q
∀q∃r
q ∨ ¬r
¬q ∨ r
∃r
¬q
q ∨ ¬r
¬q ∨ r
∃r
q
q ∨ ¬r
¬q ∨ r
¬r
r
∃r
1
∃r
1

Pure literal rule

Let Q be quantifier prefix and S set of clauses.
Let literal L be pure in S (i.e. $\overline{L}$ does not occur in S) then:

- If the variable of L is existentially quantified in Q then we can remove all clauses in which L occurs.
- If the variable of L is universally quantified then we can remove L from all clauses where L occurs.

Universal Literal Deletion

Consider a quantifier prefix Q and a conjunction of clauses S.

- p is universal in Q, if Q contains $\forall p$.
- p is existential in Q, if Q contains $\exists p$.
- A variable p is quantified before a variable q if p occurs before q in Q.

Example: If Q is $\forall p \exists q \forall r$ then p is quantified before both q and r; and q is quantified before r.

Theorem

Let Q be a quantifier prefix and S a conjunction of clauses. Suppose that

1. *C is a clause in S;*
2. *a variable p in C is universal in Q;*
3. *all existential variables in C are quantified before p.*

Then the deletion of the literal containing p from C does not change the truth value of QS.

Universal Literal Deletion

Consider a quantifier prefix Q and a conjunction of clauses S.

- p is universal in Q, if Q contains $\forall p$.
- p is existential in Q, if Q contains $\exists p$.
- A variable p is quantified before a variable q if p occurs before q in Q.

Example: If Q is $\forall p \exists q \forall r$ then p is quantified before both q and r; and q is quantified before r.

Theorem

Let Q be a quantifier prefix and S a conjunction of clauses. Suppose that

1. *C is a clause in S;*
2. *a variable p in C is universal in Q;*
3. *all existential variables in C are quantified before p.*

Then the deletion of the literal containing p from C does not change the truth value of QS.

Universal Literal Deletion

Consider a quantifier prefix Q and a conjunction of clauses S.

- p is universal in Q, if Q contains $\forall p$.
- p is existential in Q, if Q contains $\exists p$.
- A variable p is quantified before a variable q if p occurs before q in Q.

Example: If Q is $\forall p \exists q \forall r$ then p is quantified before both q and r; and q is quantified before r.

Theorem

Let Q be a quantifier prefix and S a conjunction of clauses. Suppose that

1. *C is a clause in S;*
2. *a variable p in C is universal in Q;*
3. *all existential variables in C are quantified before p.*

Then the deletion of the literal containing p from C does not change the truth value of QS.

Universal Literal Deletion

Consider a quantifier prefix Q and a conjunction of clauses S.

- p is universal in Q, if Q contains $\forall p$.
- p is existential in Q, if Q contains $\exists p$.
- A variable p is quantified before a variable q if p occurs before q in Q.

Example: If Q is $\forall p \exists q \forall r$ then p is quantified before both q and r; and q is quantified before r.

Theorem

Let Q be a quantifier prefix and S a conjunction of clauses. Suppose that

1. *C is a clause in S;*
2. *a variable p in C is universal in Q;*
3. *all existential variables in C are quantified before p.*

Then the deletion of the literal containing p from C does not change the truth value of QS.

Example

$\exists p \exists q \forall r \exists s((p \vee \neg r) \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$
$\exists p \exists q \forall r \exists s(p \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$
$\exists q \forall r \exists s((\neg q \vee r) \wedge (q \vee s) \wedge (q \vee r \vee \neg s)) \Rightarrow$
$\exists q \exists s(\neg q \wedge (q \vee s) \wedge (q \vee \neg s)) \Rightarrow$
$\exists s(s \wedge \neg s) \Rightarrow$
$\square$

- Apply universal literal deletion to $p \vee \neg r$
- Apply unit propagation
- Apply the pure literal rule to r
- Apply unit propagation

Example

$\exists p \exists q \forall r \exists s((p \vee \neg r) \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$
$\exists p \exists q \forall r \exists s(p \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$
$\exists q \forall r \exists s((\neg q \vee r) \wedge (q \vee s) \wedge (q \vee r \vee \neg s)) \Rightarrow$
$\exists q \exists s(\neg q \wedge (q \vee s) \wedge (q \vee \neg s)) \Rightarrow$
$\exists s(s \wedge \neg s) \Rightarrow$
$\square$

- Apply universal literal deletion to $p \vee \neg r$
- Apply unit propagation
- Apply the pure literal rule to r
- Apply unit propagation

Example

$\exists p \exists q \forall r \exists s((p \vee \neg r) \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$
$\exists p \exists q \forall r \exists s(p \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$
$\exists q \forall r \exists s((\neg q \vee r) \wedge (q \vee s) \wedge (q \vee r \vee \neg s)) \Rightarrow$
$\exists q \exists s(\neg q \wedge (q \vee s) \wedge (q \vee \neg s)) \Rightarrow$
$\exists s(s \wedge \neg s) \Rightarrow$
$\square$

- Apply universal literal deletion to $p \vee \neg r$
- Apply unit propagation
- Apply the pure literal rule to r
- Apply unit propagation

Example

$\exists p \exists q \forall r \exists s((p \vee \neg r) \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$
$\exists p \exists q \forall r \exists s(p \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$
$\exists q \forall r \exists s((\neg q \vee r) \wedge (q \vee s) \wedge (q \vee r \vee \neg s)) \Rightarrow$
$\exists q \exists s(\neg q \wedge (q \vee s) \wedge (q \vee \neg s)) \Rightarrow$
$\exists s(s \wedge \neg s) \Rightarrow$
$\square$

- Apply universal literal deletion to $p \vee \neg r$
- Apply unit propagation
- Apply the pure literal rule to r
- Apply unit propagation

Example

$$\exists p \exists q \forall r \exists s((p \vee \neg r) \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$$
$$\exists p \exists q \forall r \exists s(p \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$$
$$\exists q \forall r \exists s((\neg q \vee r) \wedge (q \vee s) \wedge (q \vee r \vee \neg s)) \Rightarrow$$
$$\exists q \exists s(\neg q \wedge (q \vee s) \wedge (q \vee \neg s)) \Rightarrow$$
$$\exists s(s \wedge \neg s) \Rightarrow$$
$$\square$$

- Apply universal literal deletion to $p \vee \neg r$
- Apply unit propagation
- Apply the pure literal rule to r
- Apply unit propagation

Example

$$\exists p \exists q \forall r \exists s((p \vee \neg r) \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$$
$$\exists p \exists q \forall r \exists s(p \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$$
$$\exists q \forall r \exists s((\neg q \vee r) \wedge (q \vee s) \wedge (q \vee r \vee \neg s)) \Rightarrow$$
$$\exists q \exists s(\neg q \wedge (q \vee s) \wedge (q \vee \neg s)) \Rightarrow$$
$$\exists s(s \wedge \neg s) \Rightarrow$$
$$\square$$

- Apply universal literal deletion to $p \vee \neg r$
- Apply unit propagation
- Apply the pure literal rule to r
- Apply unit propagation

Example

$$\exists p \exists q \forall r \exists s((p \vee \neg r) \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$$
$$\exists p \exists q \forall r \exists s(p \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$$
$$\exists q \forall r \exists s((\neg q \vee r) \wedge (q \vee s) \wedge (q \vee r \vee \neg s)) \Rightarrow$$
$$\exists q \exists s(\neg q \wedge (q \vee s) \wedge (q \vee \neg s)) \Rightarrow$$
$$\exists s(s \wedge \neg s) \Rightarrow$$
$$\square$$

- Apply universal literal deletion to $p \vee \neg r$
- Apply unit propagation
- Apply the pure literal rule to r
- Apply unit propagation

Example

$$\exists p \exists q \forall r \exists s((p \vee \neg r) \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$$
$$\exists p \exists q \forall r \exists s(p \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$$
$$\exists q \forall r \exists s((\neg q \vee r) \wedge (q \vee s) \wedge (q \vee r \vee \neg s)) \Rightarrow$$
$$\exists q \exists s(\neg q \wedge (q \vee s) \wedge (q \vee \neg s)) \Rightarrow$$
$$\exists s(s \wedge \neg s) \Rightarrow$$
$$\square$$

- Apply universal literal deletion to $p \vee \neg r$
- Apply unit propagation
- Apply the pure literal rule to r
- Apply unit propagation

Example

$$\exists p \exists q \forall r \exists s((p \vee \neg r) \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$$
$$\exists p \exists q \forall r \exists s(p \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$$
$$\exists q \forall r \exists s((\neg q \vee r) \wedge (q \vee s) \wedge (q \vee r \vee \neg s)) \Rightarrow$$
$$\exists q \exists s(\neg q \wedge (q \vee s) \wedge (q \vee \neg s)) \Rightarrow$$
$$\exists s(s \wedge \neg s) \Rightarrow$$
$$\square$$

- Apply universal literal deletion to $p \vee \neg r$
- Apply unit propagation
- Apply the pure literal rule to r
- Apply unit propagation

Example

$$\exists p \exists q \forall r \exists s((p \vee \neg r) \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$$
$$\exists p \exists q \forall r \exists s(p \wedge (\neg q \vee r) \wedge (\neg p \vee q \vee s) \wedge (\neg p \vee q \vee r \vee \neg s)) \Rightarrow$$
$$\exists q \forall r \exists s((\neg q \vee r) \wedge (q \vee s) \wedge (q \vee r \vee \neg s)) \Rightarrow$$
$$\exists q \exists s(\neg q \wedge (q \vee s) \wedge (q \vee \neg s)) \Rightarrow$$
$$\exists s(s \wedge \neg s) \Rightarrow$$
$$\square$$

- Apply universal literal deletion to $p \vee \neg r$
- Apply unit propagation
- Apply the pure literal rule to r
- Apply unit propagation

QBF and OBDD

We know how to apply boolean operations to OBDDs. Can we also apply quantification to OBDDs in a straighforward way?

Quantification: given an OBDD representing a formula F, find an OBDD representing $\exists_1 p_1 \ldots \exists_n p_n F$

There is no simple algorithm for quantification in general, but there is one when $\exists_1 \ldots \exists_n$ are the same quantifier.

QBF and OBDD

We know how to apply boolean operations to OBDDs. Can we also apply quantification to OBDDs in a straighforward way?

Quantification: given an OBDD representing a formula F, find an OBDD representing $\exists\forall_1 p_1 \ldots \exists\forall_n p_n F$

There is no simple algorithm for quantification in general, but there is one when $\exists\forall_1 \ldots \exists\forall_n$ are the same quantifier.

QBF and OBDD

We know how to apply boolean operations to OBDDs. Can we also apply quantification to OBDDs in a straighforward way?

Quantification: given an OBDD representing a formula F, find an OBDD representing $\exists\forall_1 p_1 \ldots \exists\forall_n p_n F$

There is no simple algorithm for quantification in general, but there is one when $\exists\forall_1 \ldots \exists\forall_n$ are the same quantifier.

Quantification for OBDDs

We can use the following properties of QBFs:

- $\exists p\,(\ \mathit{if}\ p\ \mathit{then}\ F\ \mathit{else}\ G) \equiv F \vee G$;
- $\forall p\,(\ \mathit{if}\ p\ \mathit{then}\ F\ \mathit{else}\ G) \equiv F \wedge G$;
- If $p \neq q$, then
  $\exists p\,(\ \mathit{if}\ q\ \mathit{then}\ F\ \mathit{else}\ G) \equiv \ \ \mathit{if}\ q\ \mathit{then}\ \exists pF\ \mathit{else}\ \exists pG$

Quantification for OBDDs

We can use the following properties of QBFs:

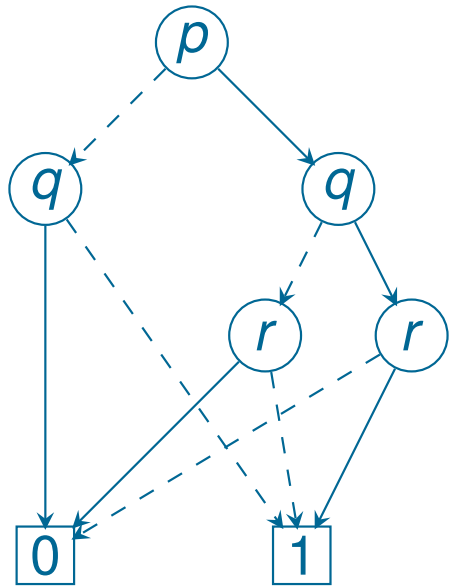
- $\exists p\,(\ \mathit{if}\ p\ \mathit{then}\ F\ \mathit{else}\ G) \equiv F \vee G$;
- $\forall p\,(\ \mathit{if}\ p\ \mathit{then}\ F\ \mathit{else}\ G) \equiv F \wedge G$;
- If $p \neq q$, then
  $\exists\forall p\,(\ \mathit{if}\ q\ \mathit{then}\ F\ \mathit{else}\ G) \equiv \quad \mathit{if}\ q\ \mathit{then}\ \ \exists\forall pF\ \mathit{else}\ \exists\forall pG$

Existential quantification algorithm for OBDDs

procedure $ex(\{p_1, \ldots, p_k\}, \{n_1, \ldots, n_m\})$
parameters: global dag D
input: nodes $n_1, \ldots, n_m$ representing $F_1, \ldots, F_m$ in D
output: a node n representing $\exists p_1 \ldots \exists p_k (F_1 \vee \ldots \vee F_m)$ in (modified) D
begin
  if $m = 0$ **then return** $\boxed{0}$
  if some n_i is $\boxed{1}$ **then return** $\boxed{1}$
  if some n_i is $\boxed{0}$ **then**
    return $ex(\{p_1, \ldots, p_k\}, \{n_1, \ldots, n_{i-1}, n_{i+1}, \ldots, n_m\})$
  $p := max\_atom(n_1, \ldots, n_m)$
  forall $i = 1 \ldots m$
    if n_i is labelled by p
      then $(l_i, r_i) := (neg(n_i), pos(n_i))$
      else $(l_i, r_i) := (n_i, n_i)$
  if $p \in \{p_1, \ldots, p_k\}$
    then return $ex(\{p_1, \ldots, p_k\} - \{p\}, \{l_1, \ldots, l_m, r_1, \ldots, r_m\})$
    else
      $k_1 := ex(\{p_1, \ldots, p_k\}, \{l_1, \ldots, l_m\})$
      $k_2 := ex(\{p_1, \ldots, p_k\}, \{r_1, \ldots, r_m\})$
      return $integrate(k_1, p, k_2, D)$
end

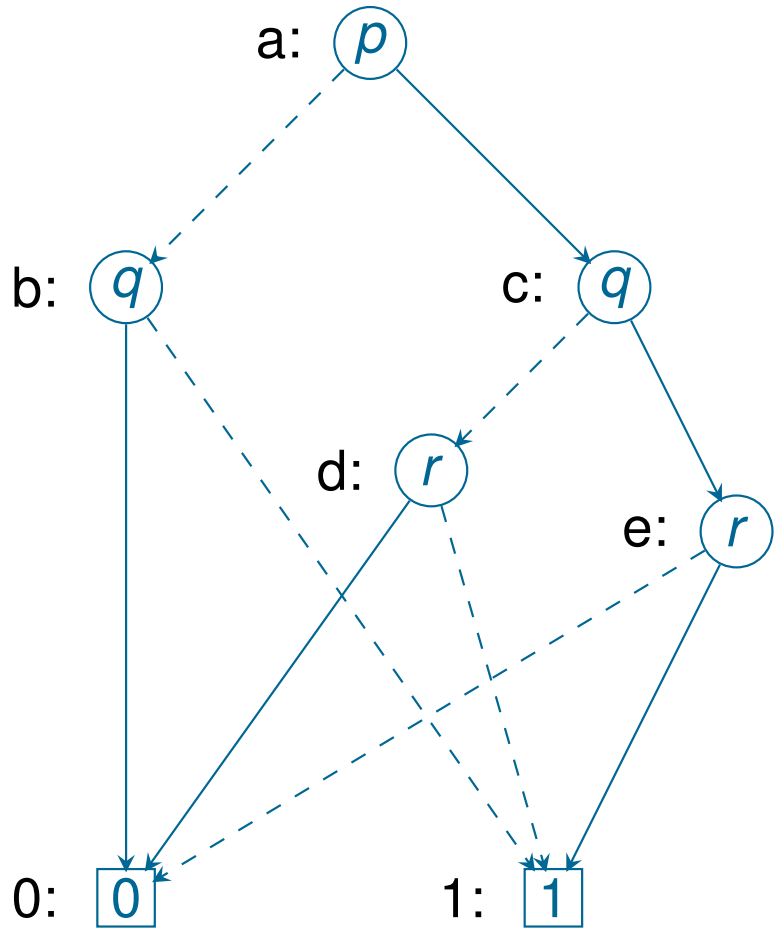
Example

Take the order $p > q > r$ and the formula $\exists p \exists r(p \leftrightarrow ((p \rightarrow r) \leftrightarrow q))$.



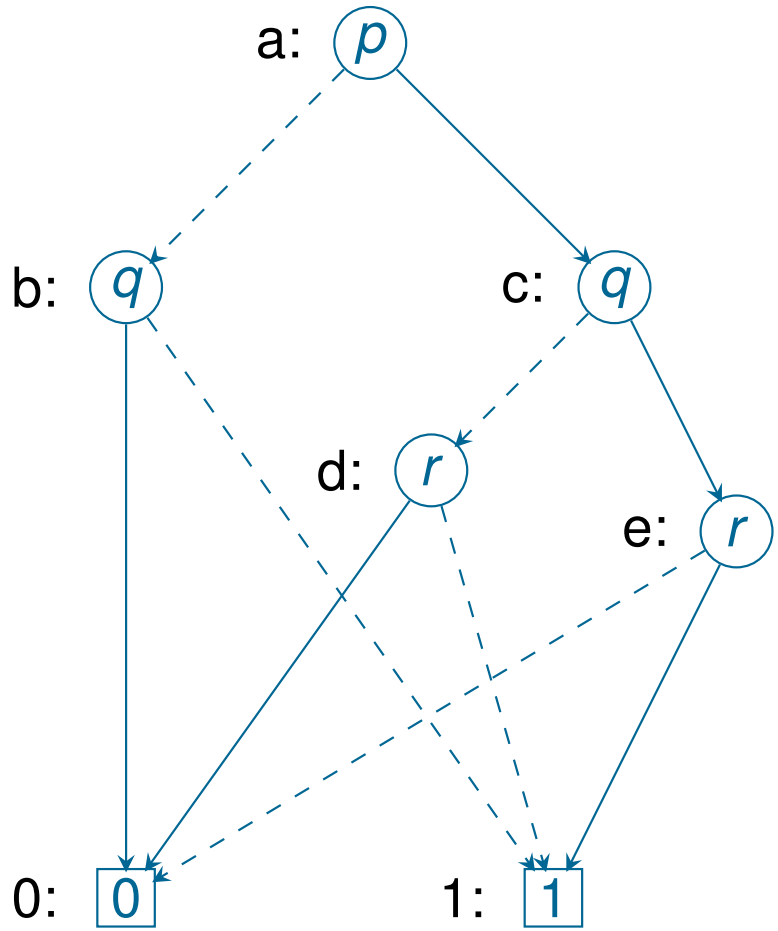
Example

$ex(\{p, r\}, \{a\})$

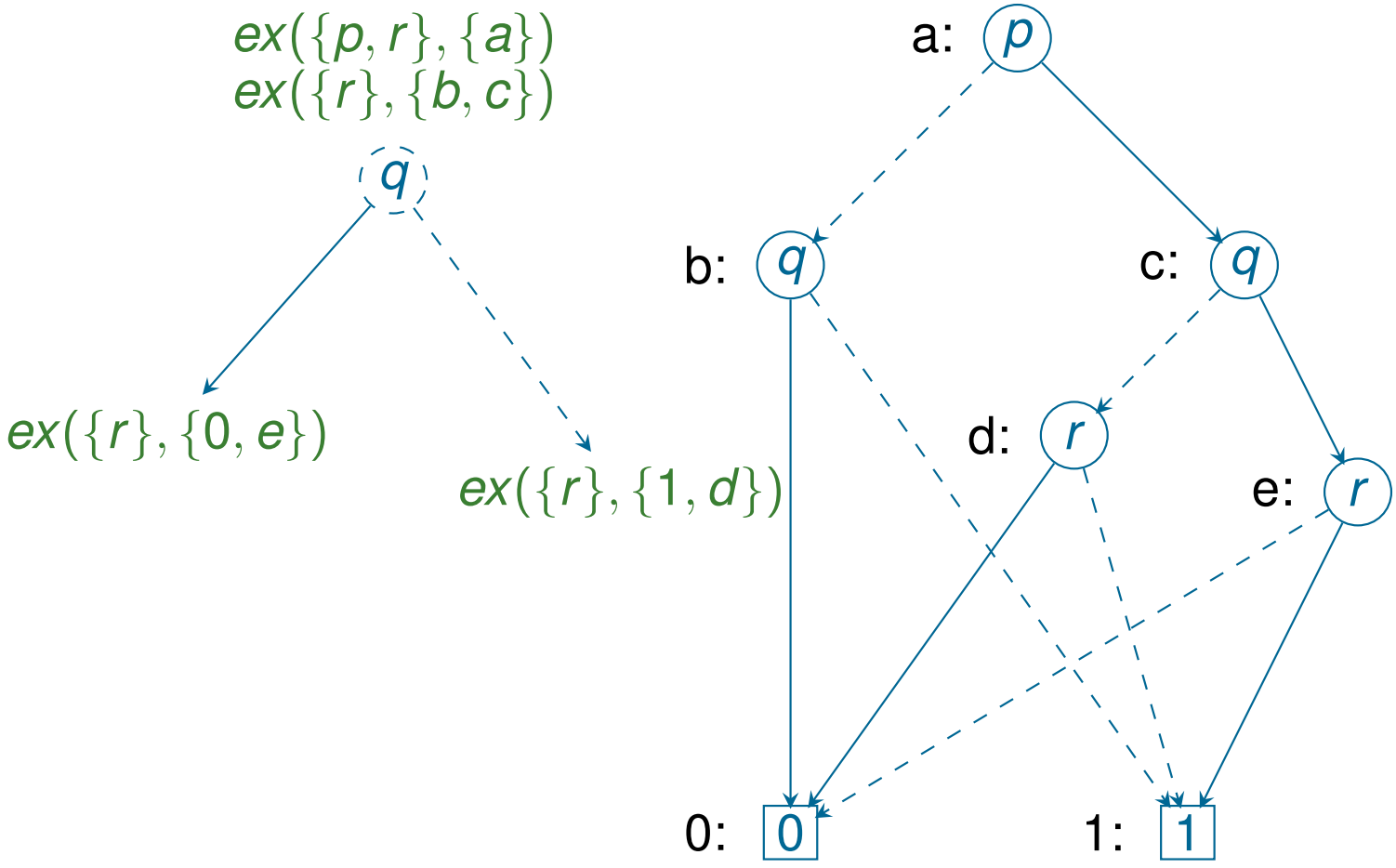


Example

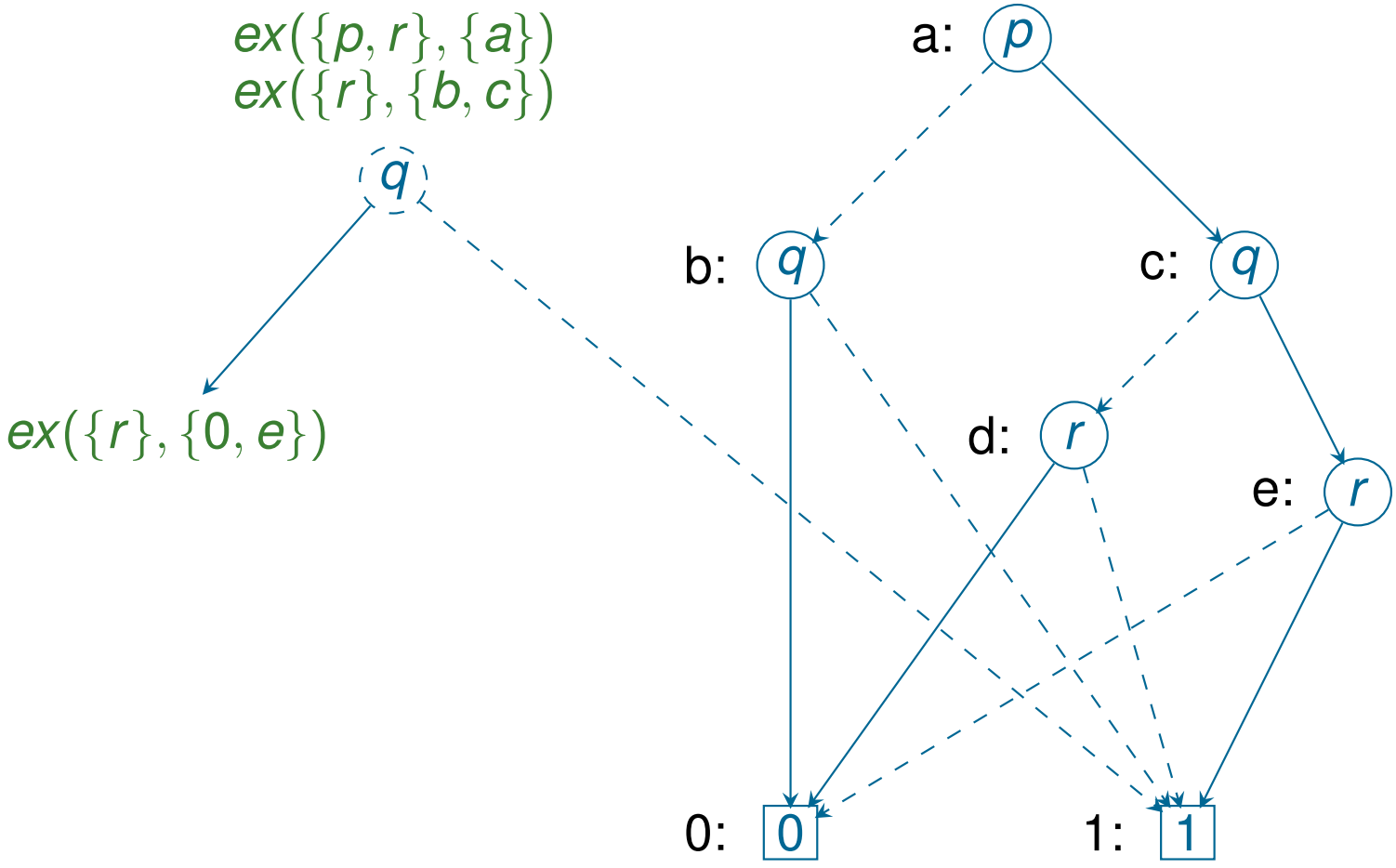
$ex(\{p, r\}, \{a\})$
$ex(\{r\}, \{b, c\})$



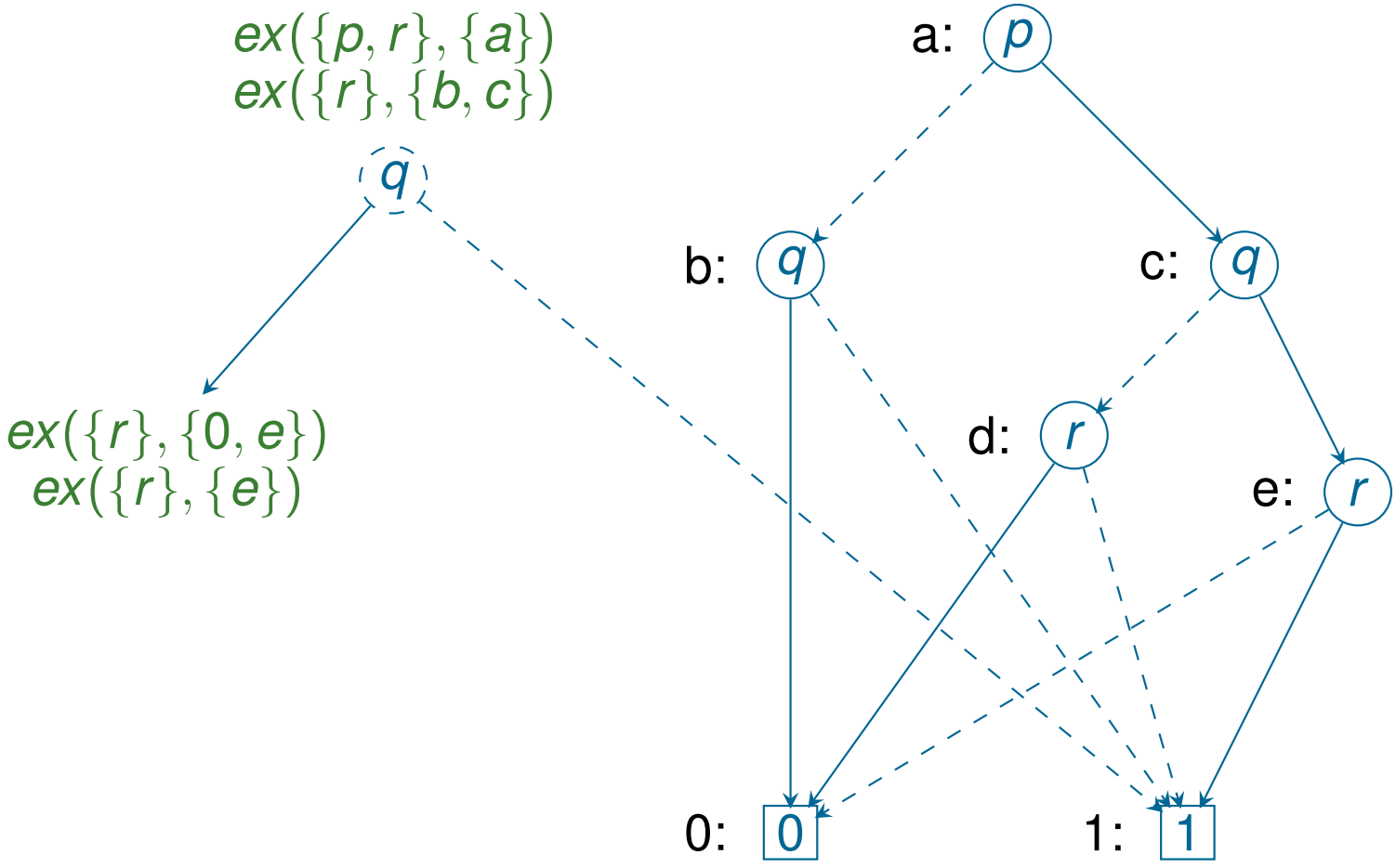
Example



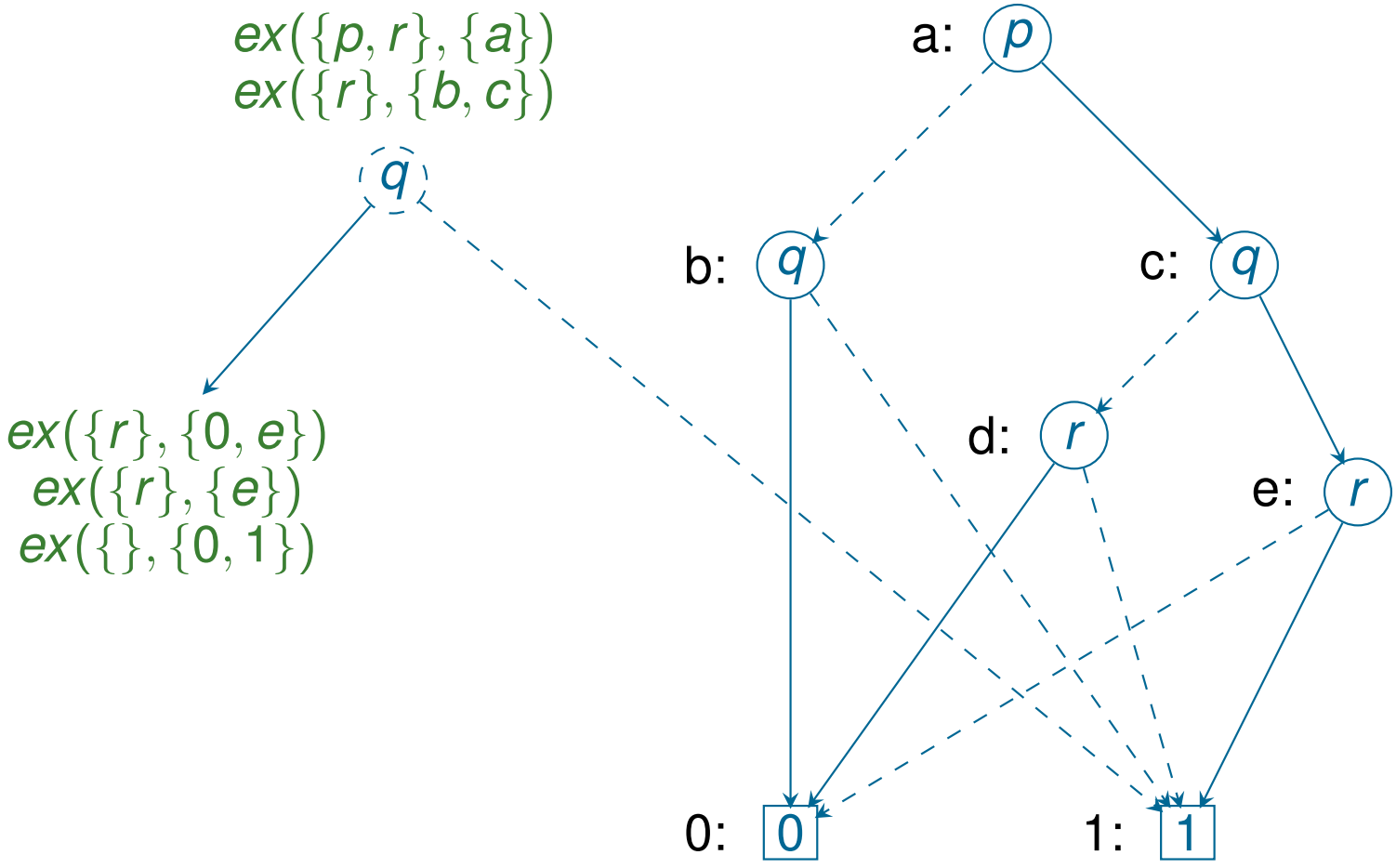
Example



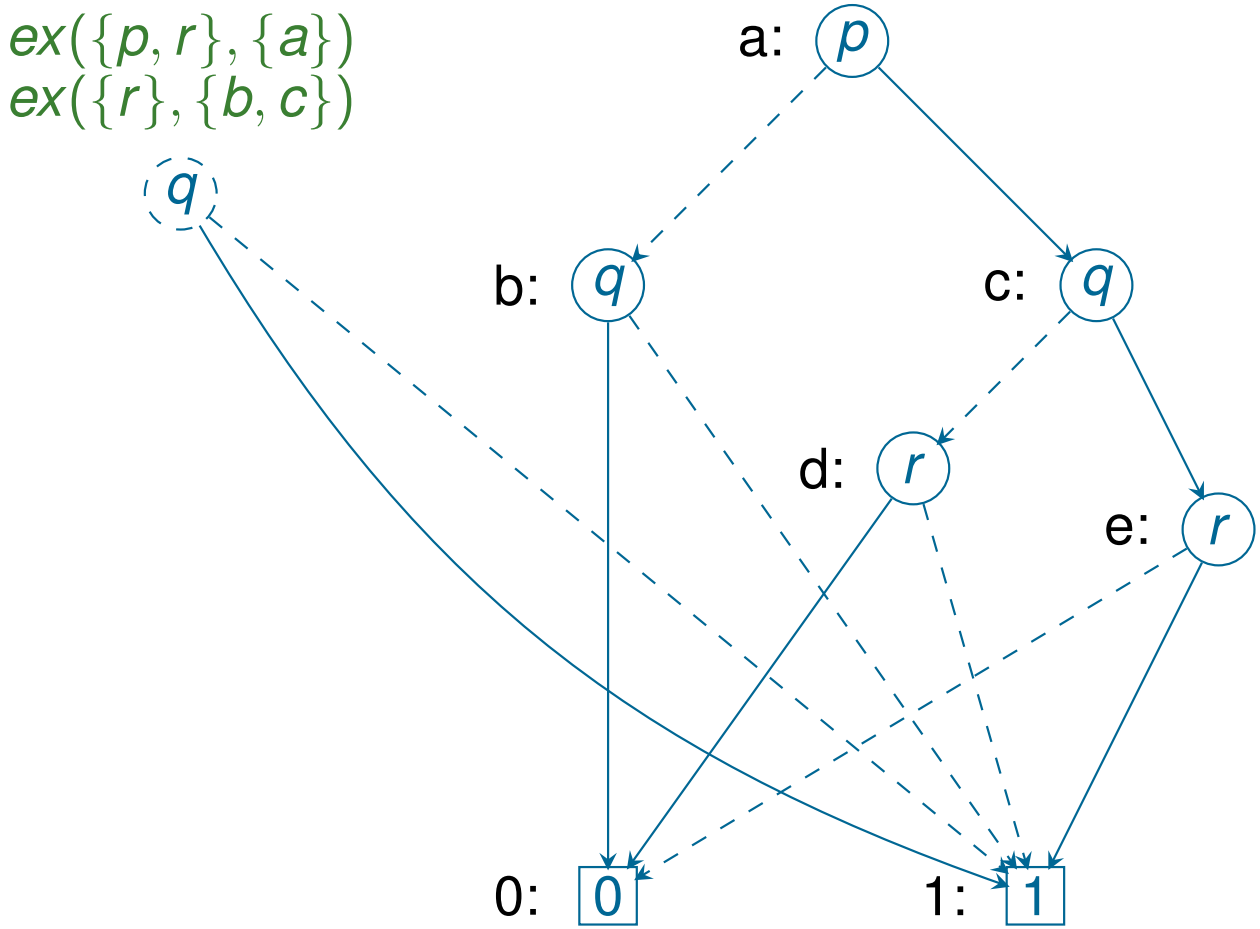
Example



Example



Example



Example

$ex(\{p, r\}, \{a\}) = 1$