Proof Using Resolution

Outline

I. Rule of resolution

II. Resolution refutation

* Figures are from the textbook site unless the source is specifically cited.

I. Resolution

An inference algorithm i is

sound if $KB \models \alpha$ whenever $KB \vdash_i \alpha$

complete if $KB \vdash_i \alpha$ whenever $KB \models \alpha$

I. Resolution

An inference algorithm i is

sound if *KB* $\models \alpha$ whenever *KB* $\vdash_i \alpha$

complete if *KB* $\vdash_i \alpha$ whenever *KB* $\models \alpha$

- Inference rules covered so far are sound.
- The inference algorithms using them may not be complete.

I. Resolution

An inference algorithm i is

- sound if $KB \models \alpha$ whenever $KB \vdash_i \alpha$
- complete if $KB \vdash_i \alpha$ whenever $KB \models \alpha$

- Inference rules covered so far are sound.
- The inference algorithms using them may not be complete.

resolution + a complete search algorithm $=$ a complete inference algorithm

I. Resolution

An inference algorithm i is

sound if *KB* $\models \alpha$ whenever *KB* $\vdash_i \alpha$

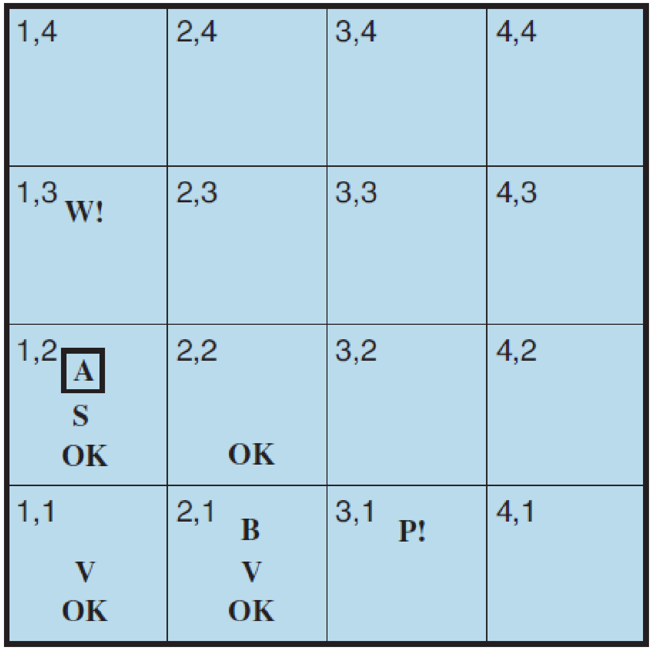
complete if *KB* $\vdash_i \alpha$ whenever *KB* $\models \alpha$

- Inference rules covered so far are sound.
- The inference algorithms using them may not be complete.

resolution + a complete search algorithm $=$ a complete inference algorithm

single inference rule

Wumpus World Revisited



KB:

R_1: $\neg P_{1,1}$

R_2: $B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$

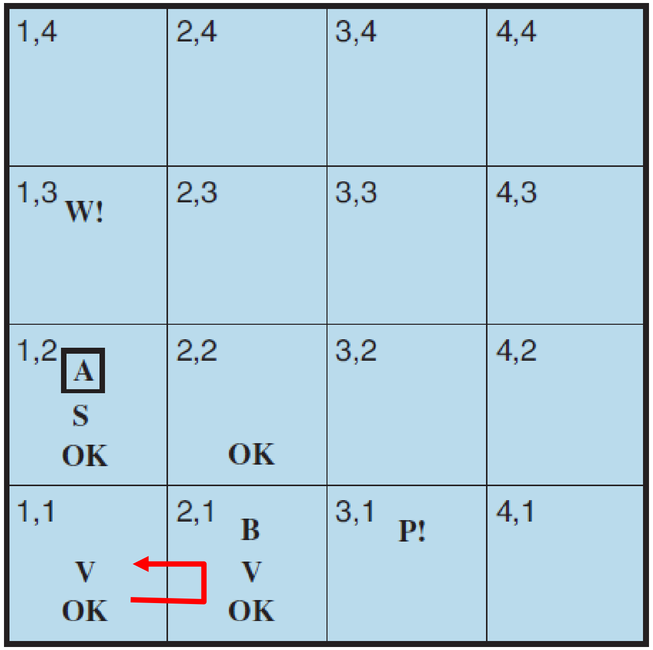
R_3: $B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$

R_4: $\neg B_{1,1}$

R_5: $B_{2,1}$

Rules

Wumpus World Revisited



Agent: [1,1] → [2,1] → [1,1]

KB:

R_1: $\neg P_{1,1}$

R_2: $B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$

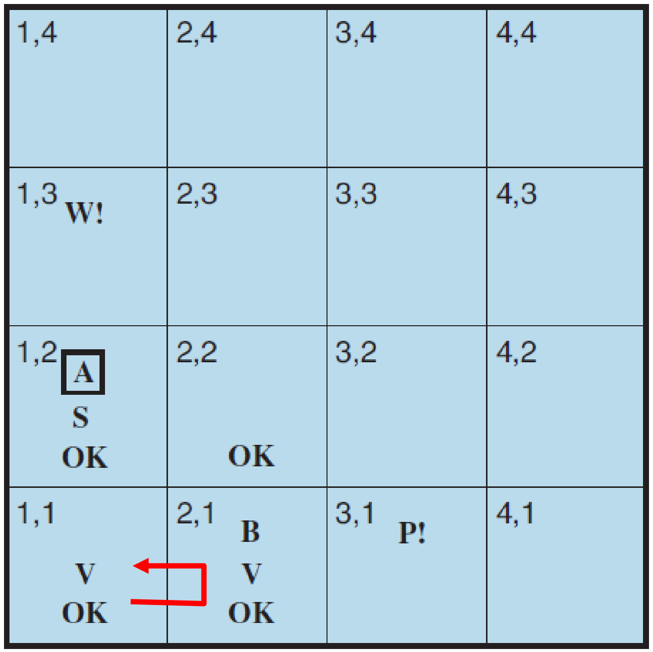
R_3: $B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$

R_4: $\neg B_{1,1}$

R_5: $B_{2,1}$

Rules

Wumpus World Revisited



Agent: [1,1] → [2,1] → [1,1]

KB:

R_1: $\neg P_{1,1}$

R_2: $B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$

R_3: $B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$

R_4: $\neg B_{1,1}$

R_5: $B_{2,1}$

Rules

R_6: $\left(B_{1,1} \Rightarrow \left(P_{1,2} \vee P_{2,1}\right)\right) \wedge \left(\left(P_{1,2} \vee P_{2,1}\right) \Rightarrow B_{1,1}\right)$

R_7: $\left(P_{1,2} \vee P_{2,1}\right) \Rightarrow B_{1,1}$

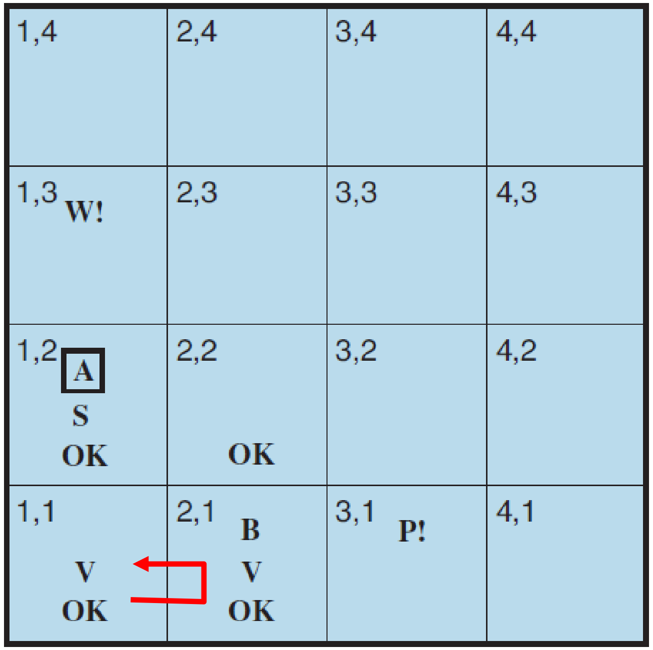
R_8: $\neg B_{1,1} \Rightarrow \neg\left(P_{1,2} \vee P_{2,1}\right)$

R_9: $\neg\left(P_{1,2} \vee P_{2,1}\right)$ // R_4, R_8

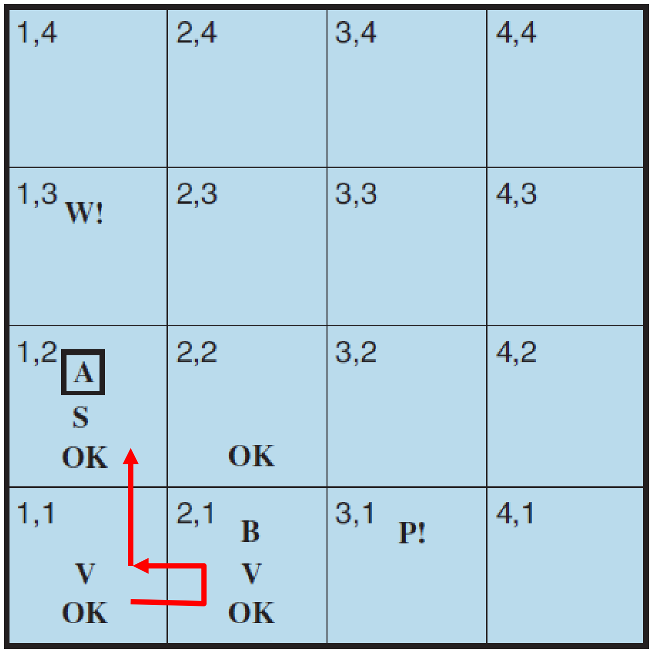
R_{10}: $\neg P_{1,2} \wedge \neg P_{2,1}$

Added to KB via inferences

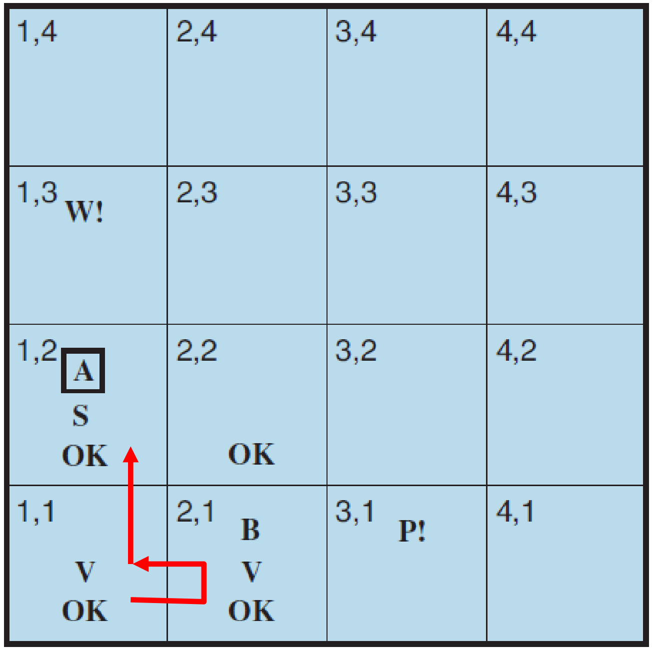
(cont'd)



(cont'd)

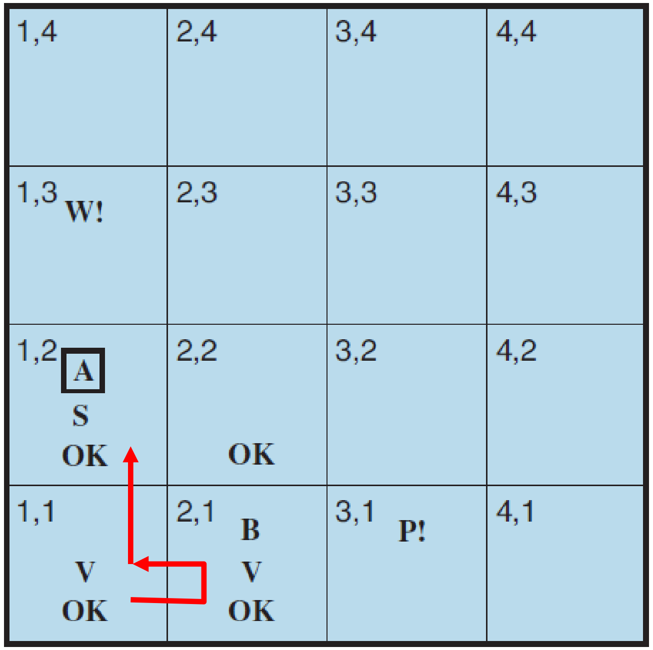


(cont'd)



[1,1] → [1,2]: stench but no breeze

(cont'd)



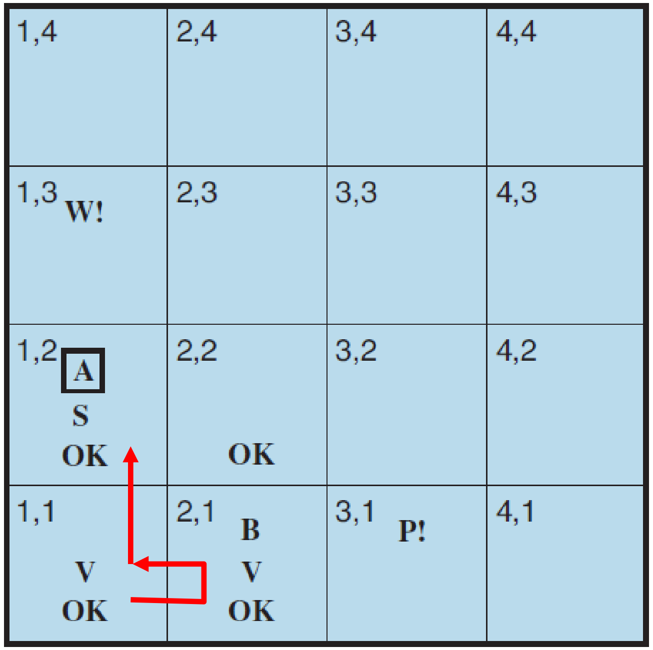
[1,1] → [1,2]: stench but no breeze

Add to KB:

R_{11}: $\neg B_{1,2}$

R_{12}: $B_{1,2} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{1,3})$

(cont'd)



[1,1] → [1,2]: stench but no breeze

Add to KB:

R_{11}: $\neg B_{1,2}$

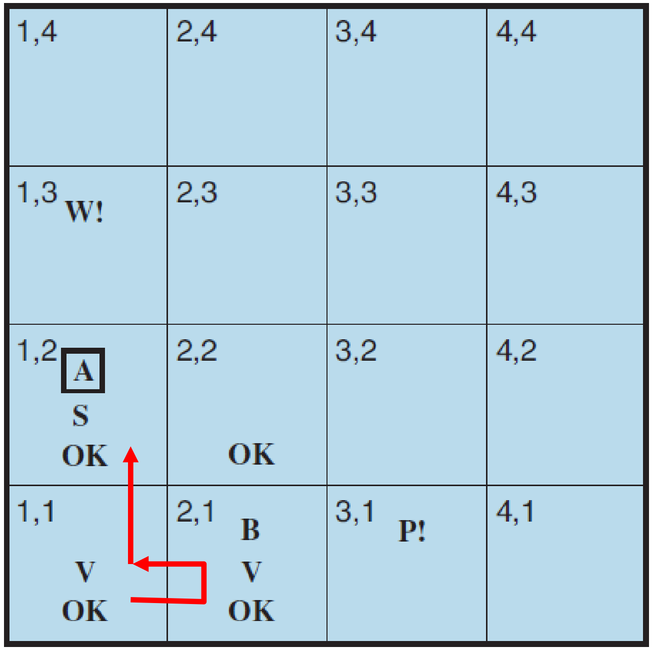
R_{12}: $B_{1,2} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{1,3})$

$\Downarrow$ Similarly, as in deriving R_{10}

R_{13}: $\neg P_{2,2}$

R_{14}: $\neg P_{1,3}$

(cont'd)



[1,1] → [1,2]: stench but no breeze

Add to KB:

R_{11}: $\neg B_{1,2}$

R_{12}: $B_{1,2} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{1,3})$

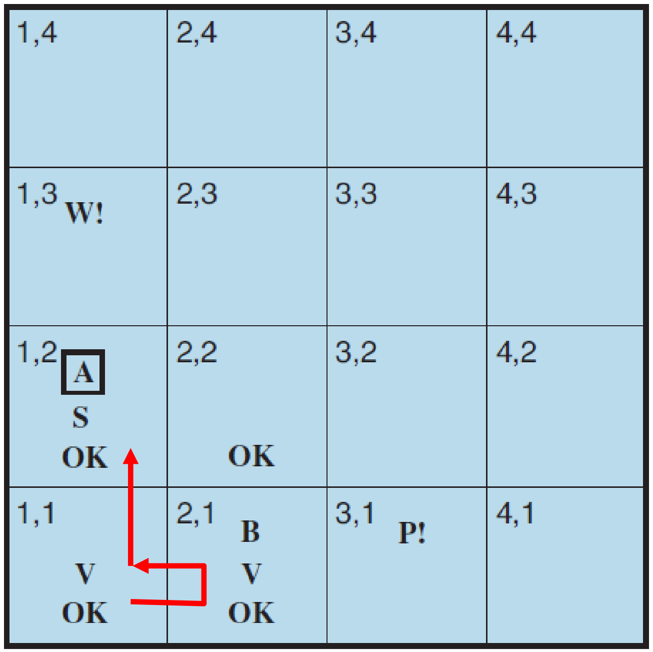
$\Downarrow$ Similarly, as in deriving R_{10}

R_{13}: $\neg P_{2,2}$

R_{14}: $\neg P_{1,3}$

R_3: $B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$

(cont'd)



[1,1] → [1,2]: stench but no breeze

Add to KB:

R_{11}: $\neg B_{1,2}$

R_{12}: $B_{1,2} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{1,3})$

$\Downarrow$ Similarly, as in deriving R_{10}

R_{13}: $\neg P_{2,2}$

R_{14}: $\neg P_{1,3}$

R_3: $B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$

$\Downarrow$ biconditional elimination

R_5: $B_{2,1}$

R_{15}: $P_{1,1} \vee P_{2,2} \vee P_{3,1}$

Resolvent

R_{13}: $\neg P_{2,2}$ R_{15}: $P_{1,1} \vee P_{2,2} \vee P_{3,1}$

Resolvent

R_{13}: $\neg P_{2,2}$ $\qquad$ R_{15}: $P_{1,1} \vee P_{2,2} \vee P_{3,1}$

resolving the two literals that are negations of each other

Resolvent

$R_{13}: \quad \neg P_{2,2}$ $\qquad$ $R_{15}: P_{1,1} \lor P_{2,2} \lor P_{3,1}$

resolving the two literals that are negations of each other

$\boxed{R_{16}: P_{1,1} \lor P_{3,1}}$ (*resolvent*)

Resolvent

R_{13}: $\neg P_{2,2}$ $\qquad$ R_{15}: $P_{1,1} \vee P_{2,2} \vee P_{3,1}$

resolving the two literals that are negations of each other

R_{16}: $P_{1,1} \vee P_{3,1}$ (*resolvent*)

If there's a pit in one of $[1,1]$, $[2,2]$, and $[3,1]$ and it's not in $[2,2]$,then it's in $[1,1]$ or $[3,1]$.

Resolvent

R_{13}: $\neg P_{2,2}$ R_{15}: $P_{1,1} \vee P_{2,2} \vee P_{3,1}$

resolving the two literals that are negations of each other

R_{16}: $P_{1,1} \vee P_{3,1}$ (*resolvent*)

If there's a pit in one of [1,1], [2,2], and [3,1] and it's not in [2,2],then it's in [1,1] or [3,1].

Resolvent

R_{13}: $\neg P_{2,2}$ R_{15}: $P_{1,1} \vee P_{2,2} \vee P_{3,1}$

resolving the two literals that are negations of each other

R_{16}: $P_{1,1} \vee P_{3,1}$ (*resolvent*)

If there's a pit in one of [1,1], [2,2], and [3,1] and it's not in [2,2],then it's in [1,1] or [3,1].

R_1: $\neg P_{1,1}$ R_{16}: $P_{1,1} \vee P_{3,1}$

Resolvent

R_{13}: $\neg P_{2,2}$

R_{15}: $P_{1,1} \vee P_{2,2} \vee P_{3,1}$

resolving the two literals that are negations of each other

R_{16}: $P_{1,1} \vee P_{3,1}$ (*resolvent*)

If there's a pit in one of [1,1], [2,2], and [3,1] and it's not in [2,2],then it's in [1,1] or [3,1].

R_1: $\neg P_{1,1}$

R_{16}: $P_{1,1} \vee P_{3,1}$

Resolvent

R_{13}: $\neg P_{2,2}$ $\quad\quad$ R_{15}: $P_{1,1} \vee P_{2,2} \vee P_{3,1}$

resolving the two literals that are negations of each other

$\boxed{R_{16}: P_{1,1} \vee P_{3,1}}$ (*resolvent*)

If there's a pit in one of [1,1], [2,2], and [3,1] and it's not in [2,2],then it's in [1,1] or [3,1].

R_1: $\neg P_{1,1}$ $\quad\quad$ R_{16}: $P_{1,1} \vee P_{3,1}$

$\boxed{R_{17}: P_{3,1}}$

Simple Resolution Rule

$$\frac{l_1 \vee \cdots \vee l_i \vee \cdots \vee l_k, \qquad m}{l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee \cdots \vee l_k}$$

(l_i and m are complementary literals, i.e.,$l_i = \neg m$ or $m = \neg l_i$.)

Simple Resolution Rule

$$\frac{l_1 \vee \cdots \vee l_i \vee \cdots \vee l_k, \qquad m}{l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee \cdots \vee l_k}$$

(l_i and m are complementary literals, i.e.,$l_i = \neg m$ or $m = \neg l_i$.)

Since m is true, then l_i must be false. But one of $l_1, \ldots, l_k$ must be true. Therefore, we can exclude l_i and assert that one of the remaining $k - 1$ literals must be true.

Simple Resolution Rule

$$\frac{l_1 \vee \cdots \vee l_i \vee \cdots \vee l_k, \qquad m}{l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee \cdots \vee l_k}$$

(l_i and m are complementary literals, i.e.,$l_i = \neg m$ or $m = \neg l_i$.)

Since m is true, then l_i must be false. But one of $l_1, \dots, l_k$ must be true. Therefore, we can exclude l_i and assert that one of the remaining $k - 1$ literals must be true.

Clause: a disjunction of literals.

$$R_{15}\text{:}\ P_{1,1} \vee P_{2,2} \vee P_{3,1}$$

Simple Resolution Rule

$$\frac{l_1 \vee \cdots \vee l_i \vee \cdots \vee l_k, \qquad m}{l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee \cdots \vee l_k}$$

(l_i and m are complementary literals, i.e.,$l_i = \neg m$ or $m = \neg l_i$.)

Since m is true, then l_i must be false. But one of $l_1, \ldots, l_k$ must be true. Therefore, we can exclude l_i and assert that one of the remaining $k - 1$ literals must be true.

Clause: a disjunction of literals.

$$R_{15}\colon\ P_{1,1} \vee P_{2,2} \vee P_{3,1}$$

Simple Resolution Rule

$$\frac{l_1 \vee \cdots \vee l_i \vee \cdots \vee l_k, \qquad m}{l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee \cdots \vee l_k}$$

(l_i and m are complementary literals, i.e.,$l_i = \neg m$ or $m = \neg l_i$.)

Since m is true, then l_i must be false. But one of $l_1, \ldots, l_k$ must be true. Therefore, we can exclude l_i and assert that one of the remaining $k-1$ literals must be true.

Clause: a disjunction of literals.

$$R_{15}\text{: } P_{1,1} \vee P_{2,2} \vee P_{3,1}$$

Unit clause: a single literal.

Simple Resolution Rule

$$\frac{l_1 \vee \cdots \vee l_i \vee \cdots \vee l_k, \qquad m}{l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee \cdots \vee l_k}$$

(l_i and m are complementary literals, i.e.,$l_i = \neg m$ or $m = \neg l_i$.)

Since m is true, then l_i must be false. But one of $l_1, \ldots, l_k$ must be true. Therefore, we can exclude l_i and assert that one of the remaining $k-1$ literals must be true.

Clause: a disjunction of literals.

$$R_{15}: \; P_{1,1} \vee P_{2,2} \vee P_{3,1}$$

Unit clause: a single literal.

$$R_1: \; \neg P_{2,2} \qquad\qquad R_5: \; B_{2,1}$$

Simple Resolution Rule

$$\frac{l_1 \vee \cdots \vee l_i \vee \cdots \vee l_k, \qquad m}{l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee \cdots \vee l_k}$$

(l_i and m are complementary literals, i.e.,$l_i = \neg m$ or $m = \neg l_i$.)

Since m is true, then l_i must be false. But one of $l_1, \ldots, l_k$ must be true. Therefore, we can exclude l_i and assert that one of the remaining $k - 1$ literals must be true.

Clause: a disjunction of literals.

R_{15}: $P_{1,1} \vee P_{2,2} \vee P_{3,1}$

$$\frac{P_{1,1} \vee P_{2,2} \vee P_{3,1}, \qquad \neg P_{2,2}}{P_{1,1} \vee P_{3,1}}$$

Unit clause: a single literal.

R_1: $\neg P_{2,2}$ $\qquad$ R_5: $B_{2,1}$

Full Resolution Rule

l_i and m_j are complementary literals:

$$\frac{l_1 \lor \cdots \lor l_i \lor \cdots \lor l_k, \qquad m_1 \lor \cdots \lor m_j \lor \cdots \lor m_k}{l_1 \lor \cdots \lor l_{i-1} \lor l_{i+1} \lor \cdots \lor l_k \lor m_1 \lor \cdots \lor m_{j-1} \lor m_{j+1} \lor \cdots \lor m_n}$$

Full Resolution Rule

l_i and m_j are complementary literals:

$$\frac{l_1 \lor \cdots \lor l_i \lor \cdots \lor l_k, \qquad m_1 \lor \cdots \lor m_j \lor \cdots \lor m_k}{l_1 \lor \cdots \lor l_{i-1} \lor l_{i+1} \lor \cdots \lor l_k \lor m_1 \lor \cdots \lor m_{j-1} \lor m_{j+1} \lor \cdots \lor m_n}$$

If l_i is true, then m_j is false. Hence $m_1 \lor \cdots \lor m_{j-1} \lor m_{j+1} \lor \cdots \lor m_n$ must be true.
If l_i is false, then $l_1 \lor \cdots \lor l_{i-1} \lor l_{i+1} \lor \cdots \lor l_k$ must be true.

Full Resolution Rule

l_i and m_j are complementary literals:

$$\frac{l_1 \vee \cdots \vee l_i \vee \cdots \vee l_k, \qquad m_1 \vee \cdots \vee m_j \vee \cdots \vee m_k}{l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee \cdots \vee l_k \vee m_1 \vee \cdots \vee m_{j-1} \vee m_{j+1} \vee \cdots \vee m_n}$$

If l_i is true, then m_j is false. Hence $m_1 \vee \cdots \vee m_{j-1} \vee m_{j+1} \vee \cdots \vee m_n$ must be true.
If l_i is false, then $l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \vee \cdots \vee l_k$ must be true.

$$\frac{P_{1,1} \vee P_{3,1}, \qquad \neg P_{1,1} \vee \neg P_{2,2}}{P_{3,1} \vee \neg P_{2,2}}$$

One Pair at a Time

Only one pair of complementary literals can be resolved at each step.

$$P \vee \neg Q \vee R, \qquad \neg P \vee Q$$

One Pair at a Time

Only one pair of complementary literals can be resolved at each step.

$$\frac{P \lor \neg Q \lor R, \qquad \neg P \lor Q}{\neg Q \lor R \lor Q}$$

One Pair at a Time

Only one pair of complementary literals can be resolved at each step.

$$\frac{P \vee \neg Q \vee R, \qquad\qquad \neg P \vee Q}{\neg Q \vee R \vee Q \quad \equiv \textit{true}}$$

One Pair at a Time

Only one pair of complementary literals can be resolved at each step.

$$\frac{P \lor \neg Q \lor R, \qquad \neg P \lor Q}{\neg Q \lor R \lor Q \quad \equiv \mathit{true}}$$

One Pair at a Time

Only one pair of complementary literals can be resolved at each step.

$$\frac{P \lor \neg Q \lor R, \qquad \neg P \lor Q}{\neg Q \lor R \lor Q \quad \equiv \textit{true}}$$

✓

$$\frac{P \lor \neg Q \lor R, \qquad \neg P \lor Q}{R}$$

One Pair at a Time

Only one pair of complementary literals can be resolved at each step.

$$\frac{P \vee \neg Q \vee R, \qquad \neg P \vee Q}{\neg Q \vee R \vee Q \quad \equiv \mathit{true}}$$

$$\frac{P \vee \neg Q \vee R, \qquad \neg P \vee Q}{R}$$

Incorrect conclusion!

One Pair at a Time

Only one pair of complementary literals can be resolved at each step.

$$\frac{P \vee \neg Q \vee R, \qquad \neg P \vee Q}{\neg Q \vee R \vee Q \quad \equiv \textit{true}}$$

$$\frac{P \vee \neg Q \vee R, \qquad \neg P \vee Q}{R}$$

Incorrect conclusion!

Conjunctive Normal Form

The resolution rule applies to clauses only.

Conjunctive normal form (CNF): a conjunction of clauses

Conjunctive Normal Form

The resolution rule applies to clauses only.

Conjunctive normal form (CNF): a conjunction of clauses

$$
\begin{aligned}
CNFSentence &\rightarrow Clause_1 \wedge \cdots \wedge Clause_n \\
Clause &\rightarrow Literal_1 \vee \cdots \vee Literal_m \\
Fact &\rightarrow Symbol \\
Literal &\rightarrow Symbol \mid \neg Symbol \\
Symbol &\rightarrow P \mid Q \mid R \mid \ldots
\end{aligned}
$$

Conjunctive Normal Form

The resolution rule applies to clauses only.

Conjunctive normal form (CNF): a conjunction of clauses

$$
\begin{aligned}
CNFSentence &\rightarrow Clause_1 \wedge \cdots \wedge Clause_n \\
Clause &\rightarrow Literal_1 \vee \cdots \vee Literal_m \\
Fact &\rightarrow Symbol \\
Literal &\rightarrow Symbol \mid \neg Symbol \\
Symbol &\rightarrow P \mid Q \mid R \mid \ldots
\end{aligned}
$$

Every sentence of propositional logic is equivalent to a CNF.

Converting to CNF

1. Eliminate $\Leftrightarrow$.

$$\alpha \Leftrightarrow \beta$$

Converting to CNF

1. Eliminate ⇔.

$$\alpha \Leftrightarrow \beta$$

replaced with

$$(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$$

Converting to CNF

1. Eliminate $\Leftrightarrow$.

$$\alpha \Leftrightarrow \beta$$

replaced with

$$(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$$

$$B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$$

$$(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Longrightarrow B_{1,1})$$

Converting to CNF

1. Eliminate $\Leftrightarrow$.

$$\alpha \Leftrightarrow \beta$$

replaced with

$$(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$$

$$B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$$

$$(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Longrightarrow B_{1,1})$$

2. Eliminate $\Longrightarrow$.

Converting to CNF

1. Eliminate $\Leftrightarrow$.

$$\alpha \Leftrightarrow \beta$$

replaced with

$$(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$$

$$B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$$

$$(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Longrightarrow B_{1,1})$$

2. Eliminate $\Longrightarrow$.

$$\alpha \Rightarrow \beta$$

$$\neg\alpha \vee \beta$$

Converting to CNF

1. Eliminate $\Leftrightarrow$.

$$\alpha \Leftrightarrow \beta$$

replaced with

$$(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$$

2. Eliminate $\Longrightarrow$.

$$\alpha \Rightarrow \beta$$

$$\neg\alpha \vee \beta$$

$$B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$$

$$(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Longrightarrow B_{1,1})$$

$$(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge (\neg(P_{1,2} \vee P_{2,1}) \vee B_{1,1})$$

Converting to CNF

1. Eliminate $\Leftrightarrow$.

$$\alpha \Leftrightarrow \beta$$

replaced with

$$(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$$

$$B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$$

$$(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Longrightarrow B_{1,1})$$

$$(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge (\neg(P_{1,2} \vee P_{2,1}) \vee B_{1,1})$$

2. Eliminate $\Longrightarrow$.

$$\alpha \Rightarrow \beta$$

$$\neg\alpha \vee \beta$$

3. Move $\neg$ inwards, repeatedly applying

$$\neg(\neg\alpha) \equiv \alpha$$
$$\neg(\alpha \vee \beta) \equiv (\neg\alpha \wedge \neg\beta)$$
$$\neg(\alpha \wedge \beta) \equiv (\neg\alpha \vee \neg\beta)$$

Converting to CNF

1. Eliminate $\Leftrightarrow$.

$$\alpha \Leftrightarrow \beta$$

replaced with

$$(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$$

2. Eliminate $\Longrightarrow$.

$$\alpha \Rightarrow \beta$$

$$\neg\alpha \vee \beta$$

3. Move $\neg$ inwards, repeatedly applying

$$\neg(\neg\alpha) \equiv \alpha$$
$$\neg(\alpha \vee \beta) \equiv (\neg\alpha \wedge \neg\beta)$$
$$\neg(\alpha \wedge \beta) \equiv (\neg\alpha \vee \neg\beta)$$

$$B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$$

$$(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Longrightarrow B_{1,1})$$

$$(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge (\neg(P_{1,2} \vee P_{2,1}) \vee B_{1,1})$$

(¬B 1,1 ∨ ¬ P 1,2 ∨ P 2,1) ∧ ((¬ P 1,2 ∧ ¬, I

1,1 1 1,1 , 1 1 2,1 2 1,2 1 2,1 , 2 2 1,2 1,1 1,1 1,

Converting to CNF

1. Eliminate $\Leftrightarrow$.

$$\alpha \Leftrightarrow \beta$$

replaced with

$$(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$$

$$B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$$

$$(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Longrightarrow B_{1,1})$$

2. Eliminate $\Longrightarrow$.

$$\alpha \Rightarrow \beta$$

$$\neg\alpha \vee \beta$$

$$(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge (\neg(P_{1,2} \vee P_{2,1}) \vee B_{1,1})$$

3. Move $\neg$ inwards, repeatedly applying

$$\neg(\neg\alpha) \equiv \alpha$$
$$\neg(\alpha \vee \beta) \equiv (\neg\alpha \wedge \neg\beta)$$
$$\neg(\alpha \wedge \beta) \equiv (\neg\alpha \vee \neg\beta)$$

$(\neg B\ 1{,}1 \vee \neg\ P\ 1{,}2 \vee\ \ P\ 2{,}1\ ) \wedge \big((\neg\ P\ 1{,}2 \wedge \neg,$
1,1 1 1,1 , 1 1 2,1 2 1,2 1 2,1 , 2 2 1,2 1,1 1,1 1,

4. Apply the distributivity law

Converting to CNF

1. Eliminate $\Leftrightarrow$.

$$\alpha \Leftrightarrow \beta$$

replaced with

$$(\alpha \Rightarrow \beta) \land (\beta \Rightarrow \alpha)$$

2. Eliminate $\Longrightarrow$.

$$\alpha \Rightarrow \beta$$

$$\neg\alpha \lor \beta$$

3. Move $\neg$ inwards, repeatedly applying

$$\neg(\neg\alpha) \equiv \alpha$$
$$\neg(\alpha \lor \beta) \equiv (\neg\alpha \land \neg\beta)$$
$$\neg(\alpha \land \beta) \equiv (\neg\alpha \lor \neg\beta)$$

4. Apply the distributivity law

$$B_{1,1} \Leftrightarrow (P_{1,2} \lor P_{2,1})$$

$$(B_{1,1} \Rightarrow (P_{1,2} \lor P_{2,1})) \land ((P_{1,2} \lor P_{2,1}) \Longrightarrow B_{1,1})$$

$$(\neg B_{1,1} \lor P_{1,2} \lor P_{2,1}) \land (\neg(P_{1,2} \lor P_{2,1}) \lor B_{1,1})$$

(¬B 1,1 ∨ ¬ P 1,2 ∨ P 2,1) ∧ ((¬ P 1,2 ∧ ¬, I
1,1 1 1,1 , 1 1 2,1 2 1,2 1 2,1 , 2 2 1,2 1,1 1,1 1,

(¬B 1,1 ∨ ¬ P 1,2 ∨ P 2,1) ∧ (¬ P 1,2 ∨ B 1,1) ∧ (¬ P
1,1 1 1,1 , 1 1 2,1 1 1,1 , 2 2 1,2 1 2,1 , 2 2 1,2 1,1 1,1 1,1

Converting to CNF

1. Eliminate ⇔.

$$\alpha \Leftrightarrow \beta$$

replaced with

$$(\alpha \Rightarrow \beta) \land (\beta \Rightarrow \alpha)$$

$$B_{1,1} \Leftrightarrow (P_{1,2} \lor P_{2,1})$$

$$(B_{1,1} \Rightarrow (P_{1,2} \lor P_{2,1})) \land ((P_{1,2} \lor P_{2,1}) \Longrightarrow B_{1,1})$$

2. Eliminate ⟹.

$$\alpha \Rightarrow \beta$$

$$\neg\alpha \lor \beta$$

$$(\neg B_{1,1} \lor P_{1,2} \lor P_{2,1}) \land (\neg(P_{1,2} \lor P_{2,1}) \lor B_{1,1})$$

3. Move ¬ inwards, repeatedly applying

$$\neg(\neg\alpha) \equiv \alpha$$
$$\neg(\alpha \lor \beta) \equiv (\neg\alpha \land \neg\beta)$$
$$\neg(\alpha \land \beta) \equiv (\neg\alpha \lor \neg\beta)$$

(¬B 1,1 ∨ ¬ P 1,2 ∨ P 2,1) ∧ ((¬ P 1,2 ∧ ¬, I
1,1 1 1,1 , 1 1 2,1 2 1,2 1 2,1 , 2 2 1,2 1,1 1,1 1,

4. Apply the distributivity law

(¬B 1,1 ∨ ¬ P 1,2 ∨ P 2,1) ∧ (¬ P 1,2 ∨ B 1,1) ∧ (¬ P
1,1 1 1,1 , 1 1 2,1 1 1,1 , 2 2 1,2 1 2,1 , 2 2 1,2 1,1 1,1 1,

II. Proof by Resolution – An Example

KB:

P

$P \rightarrow (Q \vee R)$

$Q \rightarrow S$

$R \rightarrow (S \wedge T)$

II. Proof by Resolution – An Example

KB:

P

$P \rightarrow (Q \lor R)$ ----→ $\neg P \lor Q \lor R$

$Q \rightarrow S$

$R \rightarrow (S \land T)$

II. Proof by Resolution – An Example

KB:

P

$P \rightarrow (Q \vee R)$ ----→ $\neg P \vee Q \vee R$

$Q \rightarrow S$ ----→ $\neg Q \vee S$

$R \rightarrow (S \wedge T)$

II. Proof by Resolution – An Example

KB:

P		
$P \to (Q \lor R)$	$\dashrightarrow$	$\neg P \lor Q \lor R$
$Q \to S$	$\dashrightarrow$	$\neg Q \lor S$
$R \to (S \land T)$	$\dashrightarrow$	$\neg R \lor (S \land T)$

II. Proof by Resolution – An Example

KB:

P

$P \rightarrow (Q \vee R)$ $\dashrightarrow$ $\neg P \vee Q \vee R$

$Q \rightarrow S$ $\dashrightarrow$ $\neg Q \vee S$

$R \rightarrow (S \wedge T)$ $\dashrightarrow$ $\neg R \vee (S \wedge T)$

$\dashrightarrow$ $(\neg R \vee S) \wedge (\neg R \vee T)$

II. Proof by Resolution – An Example

KB:

P

$P \rightarrow (Q \vee R)$ ⟶ $\neg P \vee Q \vee R$

$Q \rightarrow S$ ⟶ $\neg Q \vee S$

$R \rightarrow (S \wedge T)$ ⟶ $\neg R \vee (S \wedge T)$

⟶ $(\neg R \vee S) \wedge (\neg R \vee T)$

Q: $KB \vdash S$?

II. Proof by Resolution – An Example

KB:

P

$P \to (Q \vee R)$ ----→ $\neg P \vee Q \vee R$

$Q \to S$ ----→ $\neg Q \vee S$

$R \to (S \wedge T)$ ----→ $\neg R \vee (S \wedge T)$

----→ $(\neg R \vee S) \wedge (\neg R \vee T)$

Q: $KB \vdash S$?

1. Converting sentences to CNF

II. Proof by Resolution – An Example

KB:

P

$P \rightarrow (Q \vee R)$ ----→ $\neg P \vee Q \vee R$

$Q \rightarrow S$ ----→ $\neg Q \vee S$

$R \rightarrow (S \wedge T)$ ----→ $\neg R \vee (S \wedge T)$

----→ $(\neg R \vee S) \wedge (\neg R \vee T)$

Q: $KB \vdash S$?

1. Converting sentences to CNF

KB: P

$\neg P \vee Q \vee R$

$\neg Q \vee S$

$(\neg R \vee S) \wedge (\neg R \vee T)$

II. Proof by Resolution – An Example

KB:

P

$P \rightarrow (Q \vee R)$ ----→ $\neg P \vee Q \vee R$

$Q \rightarrow S$ ----→ $\neg Q \vee S$

$R \rightarrow (S \wedge T)$ ----→ $\neg R \vee (S \wedge T)$

----→ $(\neg R \vee S) \wedge (\neg R \vee T)$

Q: *KB* $\vdash S$?

1. Converting sentences to CNF
2. Spilt each conjunction into clauses.

KB: P

$\neg P \vee Q \vee R$

$\neg Q \vee S$

$(\neg R \vee S) \wedge (\neg R \vee T)$

II. Proof by Resolution – An Example

KB:

P

$P \rightarrow (Q \vee R)$ $\dashrightarrow$ $\neg P \vee Q \vee R$

$Q \rightarrow S$ $\dashrightarrow$ $\neg Q \vee S$

$R \rightarrow (S \wedge T)$ $\dashrightarrow$ $\neg R \vee (S \wedge T)$

$\dashrightarrow$ $(\neg R \vee S) \wedge (\neg R \vee T)$

Q: *KB* $\vdash S$?

1. Converting sentences to CNF
2. Spilt each conjunction into clauses.

KB:

P

$\neg P \vee Q \vee R$

$\neg Q \vee S$

$(\neg R \vee S) \wedge (\neg R \vee T)$

$$\left\{ \begin{array}{l} \neg R \vee S \\ \neg R \vee T \end{array} \right.$$

II. Proof by Resolution – An Example

KB:

P	
$P \rightarrow (Q \vee R)$	$\dashrightarrow \neg P \vee Q \vee R$
$Q \rightarrow S$	$\dashrightarrow \neg Q \vee S$
$R \rightarrow (S \wedge T)$	$\dashrightarrow \neg R \vee (S \wedge T)$
	$\dashrightarrow (\neg R \vee S) \wedge (\neg R \vee T)$

Q: $KB \vdash S$?

1. Converting sentences to CNF

KB:

P

$\neg P \vee Q \vee R$

$\neg Q \vee S$

$(\neg R \vee S) \wedge (\neg R \vee T)$

2. Spilt each conjunction into clauses.

KB:

P

$\neg P \vee Q \vee R$

$\neg Q \vee S$

$\neg R \vee S$

$\neg R \vee T$

Proof by Resolution

KB (updated):

(1) P

(2) $\neg P \vee Q \vee R$

(3) $\neg Q \vee S$

(4) $\neg R \vee S$

(5) $\neg R \vee T$

(1) P (2) $\neg P \vee Q \vee R$

Proof by Resolution

KB (updated):

(1) P

(2) $\neg P \vee Q \vee R$

(3) $\neg Q \vee S$

(4) $\neg R \vee S$

(5) $\neg R \vee T$

(1) P (2) $\neg P \vee Q \vee R$

resolve

(6) $Q \vee R$

Proof by Resolution

KB (updated):

(1) P

(2) $\neg P \lor Q \lor R$

(3) $\neg Q \lor S$

(4) $\neg R \lor S$

(5) $\neg R \lor T$

(1) P (2) $\neg P \lor Q \lor R$

resolve

(3) $\neg Q \lor S$ (6) $Q \lor R$

Proof by Resolution

KB (updated):

(1) P

(2) $\neg P \lor Q \lor R$

(3) $\neg Q \lor S$

(4) $\neg R \lor S$

(5) $\neg R \lor T$

(1) P (2) $\neg P \lor Q \lor R$

resolve

(3) $\neg Q \lor S$ (6) $Q \lor R$

(7) $S \lor R$

Proof by Resolution

KB (updated):

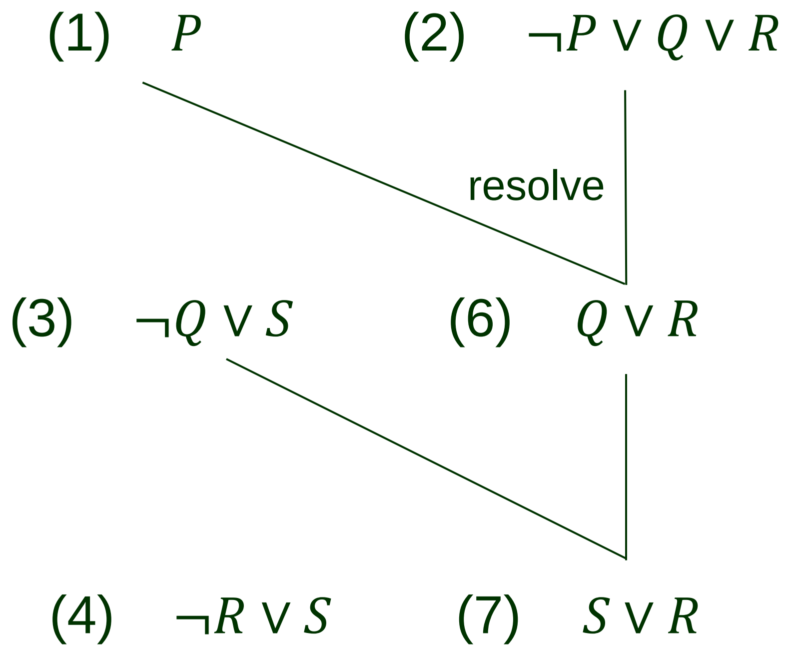
(1) P

(2) $\neg P \lor Q \lor R$

(3) $\neg Q \lor S$

(4) $\neg R \lor S$

(5) $\neg R \lor T$



Proof by Resolution

KB (updated):

(1) P
(2) $\neg P \vee Q \vee R$
(3) $\neg Q \vee S$
(4) $\neg R \vee S$
(5) $\neg R \vee T$

(1) P (2) $\neg P \vee Q \vee R$

resolve

(3) $\neg Q \vee S$ (6) $Q \vee R$

(4) $\neg R \vee S$ (7) $S \vee R$

(8) $S \vee S \equiv S$

Proof by Resolution

KB (updated):

(1) P
(2) $\neg P \lor Q \lor R$
(3) $\neg Q \lor S$
(4) $\neg R \lor S$
(5) $\neg R \lor T$

(1) P (2) $\neg P \lor Q \lor R$

resolve

(3) $\neg Q \lor S$ (6) $Q \lor R$

(4) $\neg R \lor S$ (7) $S \lor R$

(8) $S \lor S \equiv S$

Resolution tree

Resolution Refutation

(Proof by contradiction)

To show that $KB \vDash \alpha$, we show that $KB \wedge \neg\alpha$ is unsatisfiable. .

Resolution Refutation

(Proof by contradiction)
To show that $KB \models \alpha$, we show that $KB \land \neg\alpha$ is unsatisfiable. .

KB (about a summer day):

```
(1) If it is raining and you are outside then you will get wet.
(2) If it is warm and there is no rain then it is a pleasant day.
(3) You are not wet.
(4) You are outside.
(5) It is a warm day.
```

* Example taken from http://watson.latech.edu/book/intelligence/intelligenceApproaches2b2.html

Resolution Refutation

(Proof by contradiction)

To show that $KB \vDash \alpha$, we show that $KB \wedge \neg\alpha$ is unsatisfiable. .

KB (about a summer day):

```
(1) If it is raining and you are outside then you will get wet.
(2) If it is warm and there is no rain then it is a pleasant day.
(3) You are not wet.
(4) You are outside.
(5) It is a warm day.
```

Prove

```
It is a pleasant day.
```

* Example taken from http://watson.latech.edu/book/intelligence/intelligenceApproaches2b2.html

KB in Propositional Sentences

KB (rewritten):

```
(1)  ( rain ∧ outside ) ⇒ wet
(2)  ( warm ∧ ¬rain ) ⇒ pleasant
(3)  ¬wet
(4)  outside
(5)  warm
```

KB in Propositional Sentences

KB (rewritten):

```
(1)  ( rain ∧ outside ) ⇒ wet
(2)  ( warm ∧ ¬rain ) ⇒ pleasant
(3)  ¬wet
(4)  outside
(5)  warm
```

converted into clauses

```
(1)  ¬rain ∨ ¬outside ∨ wet
(2)  ¬warm ∨ rain ∨ pleasant
(3)  ¬wet
(4)  outside
(5)  warm
```

KB in Propositional Sentences

KB (rewritten):

```
(1)  ( rain ∧ outside ) ⇒ wet
(2)  ( warm ∧ ¬rain ) ⇒ pleasant
(3)  ¬wet
(4)  outside
(5)  warm
```

⇓ converted into clauses

```
(1)  ¬rain ∨ ¬outside ∨ wet
(2)  ¬warm ∨ rain ∨ pleasant
(3)  ¬wet
(4)  outside
(5)  warm
```

We add `¬pleasant` to KB and try to derive `false`.

Resolution Refutation Tree

¬pleasant (2) ¬warm ∨ rain ∨ pleasant

Resolution Refutation Tree

¬pleasant (2) ¬warm ∨ rain ∨ pleasant

¬warm ∨ rain

Resolution Refutation Tree

¬pleasant (2) ¬warm ∨ rain ∨ pleasant

¬warm ∨ rain (5) warm

Resolution Refutation Tree

¬pleasant (2) ¬warm ∨ rain ∨ pleasant

¬warm ∨ rain (5) warm

rain

Resolution Refutation Tree

¬pleasant (2) ¬warm ∨ rain ∨ pleasant

¬warm ∨ rain (5) warm

rain

(1) ¬rain ∨ ¬outside ∨ wet

Resolution Refutation Tree

¬ pleasant   (2) ¬ warm ∨ rain ∨ pleasant

¬ warm ∨ rain   (5) warm

rain   (1) ¬rain ∨ ¬ outside ∨ wet

¬ outside ∨ wet

Resolution Refutation Tree

¬pleasant　　(2) ¬warm ∨ rain ∨ pleasant

¬warm ∨ rain　　(5) warm

rain　　(1) ¬rain ∨ ¬outside ∨ wet

¬outside ∨ wet　　(4) outside

Resolution Refutation Tree

¬pleasant (2) ¬warm ∨ rain ∨ pleasant

¬warm ∨ rain (5) warm

rain (1) ¬rain ∨ ¬outside ∨ wet

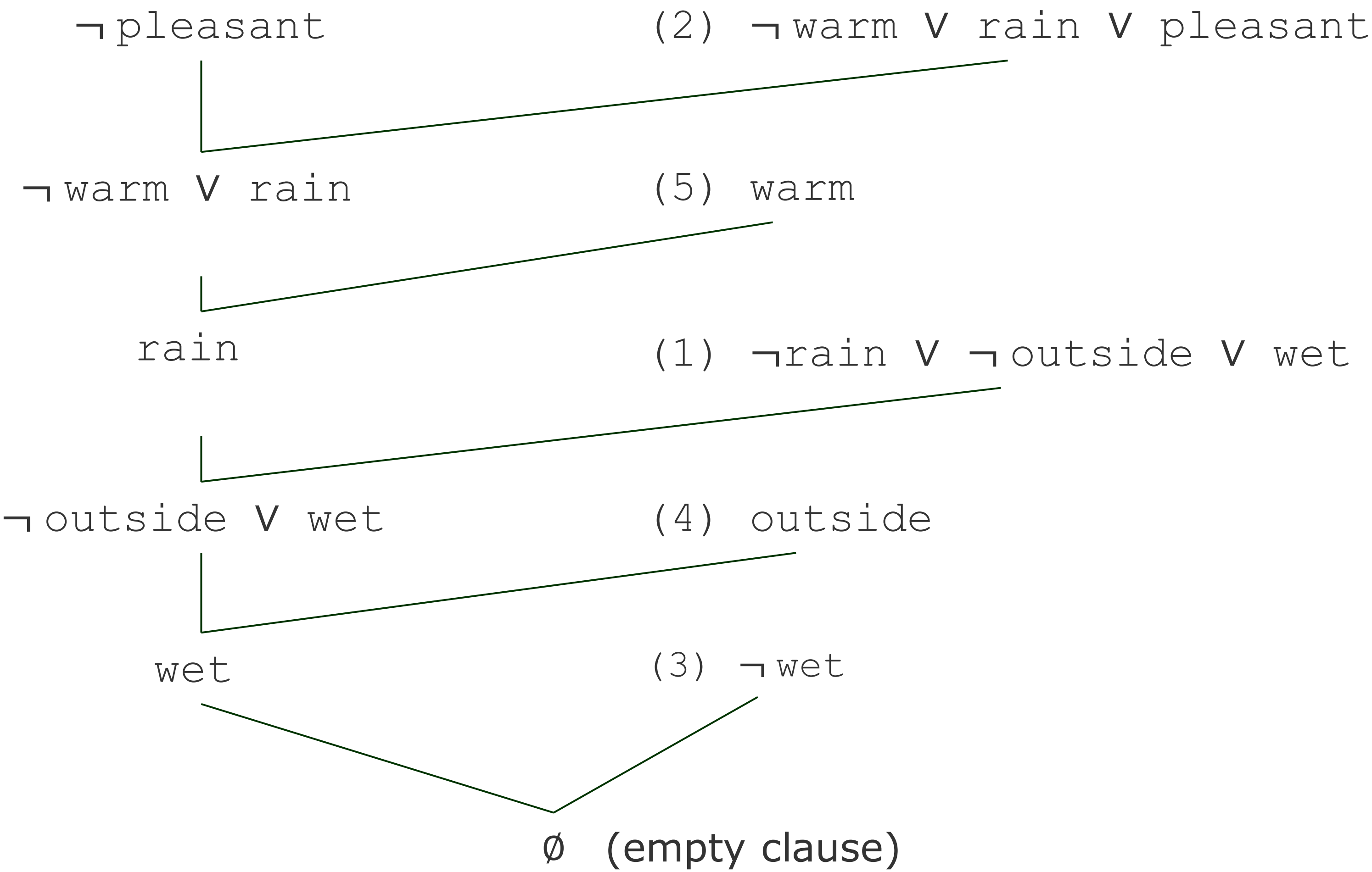
¬outside ∨ wet (4) outside

wet

Resolution Refutation Tree

¬pleasant	(2) ¬warm ∨ rain ∨ pleasant
¬warm ∨ rain	(5) warm
rain	(1) ¬rain ∨ ¬outside ∨ wet
¬outside ∨ wet	(4) outside
wet	(3) ¬wet

Resolution Refutation Tree



Resolution Refutation Tree

¬pleasant   (2) ¬warm ∨ rain ∨ pleasant

¬warm ∨ rain   (5) warm

rain   (1) ¬rain ∨ ¬outside ∨ wet

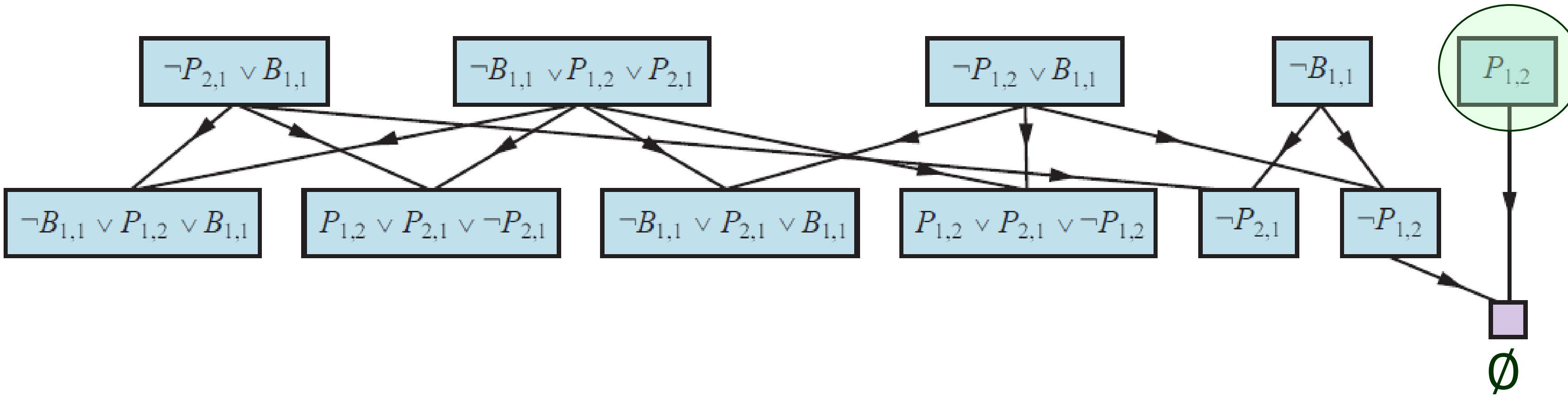
¬outside ∨ wet   (4) outside

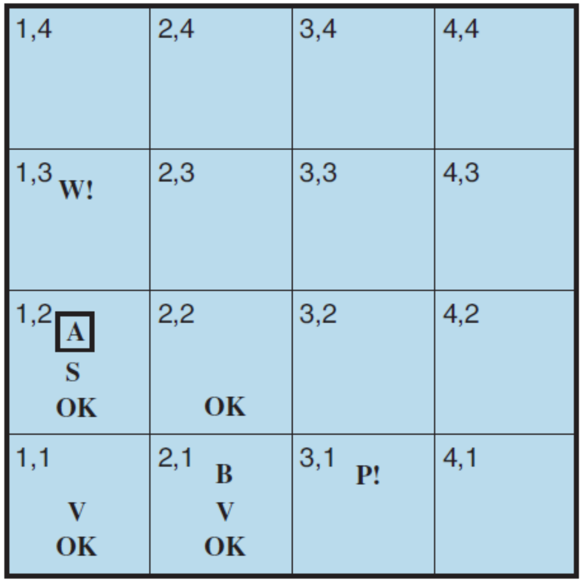
wet   (3) ¬wet

∅ (empty clause)

contradiction!

Proving $\neg P_{1,2}$ in the Wumpus World





Resolution Algorithm

function PL-RESOLUTION(KB, α) **returns** *true* or *false*
  inputs: KB, the knowledge base, a sentence in propositional logic
    α, the query, a sentence in propositional logic

  $clauses \leftarrow$ the set of clauses in the CNF representation of $KB \wedge \neg\alpha$
  $new \leftarrow \{\,\}$
  while *true* **do**
    for each pair of clauses C_i, C_j **in** *clauses* **do**
      $resolvents \leftarrow$ PL-RESOLVE(C_i, C_j)
      if *resolvents* contains the empty clause **then return** *true*
      $new \leftarrow new \cup resolvents$
    if $new \subseteq clauses$ **then return** *false* // no new clauses can be added.
    $clauses \leftarrow clauses \cup new$

The process ends in one of two situations below:

- No new clauses can be added, in which case *KB* does not entail α;
- Two clauses resolve to yield the empty clause, in which case *KB* entails α.

Completeness of Resolution

Given a set of clauses S, its *resolution closure* $RC(S)$ includes all the clauses in S as well as all the resolvents from repeated applications of the resolution rule.

Completeness of Resolution

Given a set of clauses *S*, its *resolution closure* $RC(S)$ includes all the clauses in S as well as all the resolvents from repeated applications of the resolution rule.

> $RC(S)$ is finite because only 3^n distinct clauses can be constructed out of n propositional symbols appearing in *S*.

Completeness of Resolution

Given a set of clauses *S*, its *resolution closure* *RC*(S) includes all the clauses in S as well as all the resolvents from repeated applications of the resolution rule.

RC(S) is finite because only 3^n distinct clauses can be constructed out of n propositional symbols appearing in *S*.

Ground Resolution Theorem: If S is unsatisfiable, then *RC*(S) contains the empty clause ∅.

Completeness of Resolution

Given a set of clauses *S*, its *resolution closure* *RC*(S) includes all the clauses in S as well as all the resolvents from repeated applications of the resolution rule.

RC(S) is finite because only 3^n distinct clauses can be constructed out of n propositional symbols appearing in *S*.

Ground Resolution Theorem: If S is unsatisfiable, then *RC*(S) contains the empty clause ∅.

Constructive proof by explicitly generating an assignment for S if $\emptyset \notin$ *RC*(S).

Completeness of Resolution

Given a set of clauses *S*, its *resolution closure* $RC(S)$ includes all the clauses in S as well as all the resolvents from repeated applications of the resolution rule.

$RC(S)$ is finite because only 3^n distinct clauses can be constructed out of n propositional symbols appearing in *S*.

Ground Resolution Theorem: If S is unsatisfiable, then $RC(S)$ contains the empty clause $\emptyset$.

Constructive proof by explicitly generating an assignment for S if $\emptyset \notin RC(S)$.

Completeness of Resolution

Given a set of clauses *S*, its *resolution closure* *RC*(S) includes all the clauses in S as well as all the resolvents from repeated applications of the resolution rule.

> *RC*(S) is finite because only 3^n distinct clauses can be constructed out of n propositional symbols appearing in *S*.

Ground Resolution Theorem: If S is unsatisfiable, then *RC*(S) contains the empty clause ∅.

Constructive proof by explicitly generating an assignment for S if $\emptyset \notin$ *RC*(S).