Resolution Inference Rule

\[
\begin{array}{c}
\ell_1 \lor \cdots \lor \ell_i \lor \cdots \lor \ell_k, \quad m_1 \lor \cdots \lor m_j \lor \cdots \lor m_k \\
\hline
\text{SUBST}(\theta, \ell_1 \lor \cdots \lor \ell_{i-1} \lor \ell_{i+1} \lor \cdots \lor \ell_k \lor m_1 \lor \cdots \lor m_{j-1} \lor m_{j+1} \lor \cdots \lor m_n)
\end{array}
\]

where \( \theta = \text{UNIFY}(\ell_i, m_j) \).
Resolution Inference Rule

\[
\begin{align*}
& l_1 \lor \cdots \lor l_i \lor \cdots \lor l_k, \quad m_1 \lor \cdots \lor m_j \lor \cdots \lor m_k \\
\hline
\text{SUBST}(\theta, l_1 \lor \cdots \lor l_{i-1} \lor l_{i+1} \lor \cdots \lor l_k \lor m_1 \lor \cdots \lor m_{j-1} \lor m_{j+1} \lor \cdots \lor m_n)
\end{align*}
\]

where \( \theta = \text{UNIFY}(l_i, m_j). \)

\[
\text{Animal}(F(x)) \lor \text{Loves}(G(x), x) \quad (\neg \text{Loves}(u, v) \lor \neg \text{Kills}(u, v))
\]
Resolution Inference Rule

\[
\frac{l_1 \lor \cdots \lor l_i \lor \cdots \lor l_k, \quad m_1 \lor \cdots \lor m_j \lor \cdots \lor m_k}{\text{SUBST}(\theta, l_1 \lor \cdots \lor l_{i-1} \lor l_{i+1} \lor \cdots \lor l_k \lor m_1 \lor \cdots \lor m_{j-1} \lor m_{j+1} \lor \cdots \lor m_n)}
\]

where \( \theta = \text{UNIFY}(l_i, m_j) \).

\[
\text{Animal}(F(x)) \lor \text{Loves}(G(x), x) \quad \text{(-Loves (u, v) \lor \neg \text{Kills}(u, v))}
\]

unifier: \( \theta = \{u/G(x), v/x\} \)
Resolution Inference Rule

\[ l_1 \lor \cdots \lor l_i \lor \cdots \lor l_k, \quad m_1 \lor \cdots \lor m_j \lor \cdots \lor m_k \]

\text{SUBST}(\theta, 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)\]

where \( \theta = \text{UNIFY}(l_i, m_j). \)

\[ \text{Animal}(F(x)) \lor \text{Loves}(G(x), x) \quad (\neg \text{Loves}(u, v) \lor \neg \text{Kills}(u, v)) \]

unifier: \( \theta = \{u/G(x), v/x\} \)

resolvent: \( \text{Animal}(F(x)) \lor \neg \text{Kills}(G(x), x) \)
Resolution Inference Rule

\[ l_1 \lor \cdots \lor l_i \lor \cdots \lor l_k, \quad m_1 \lor \cdots \lor m_j \lor \cdots \lor m_k \]

\[ \text{SUBST}(\theta, l_1 \lor \cdots \lor l_{i-1} \lor l_{i+1} \lor \cdots \lor l_k \lor m_1 \lor \cdots \lor m_{j-1} \lor m_{j+1} \lor \cdots \lor m_n) \]

where \( \theta = \text{UNIFY}(l_i, m_j) \).

\[ \text{Animal}(F(x)) \lor \text{Loves}(G(x), x) \]
\[ \quad (\neg \text{Loves}(u, v) \lor \neg \text{Kills}(u, v)) \]

unifier: \( \theta = \{u/G(x), v/x\} \)

resolvent: \[ \text{Animal}(F(x)) \lor \neg \text{Kills}(G(x), x) \]

- Binary resolution as given above does not yield a complete inference procedure.
- Full resolution does. It resolves subsets of literals in each clause that are unifiable.
Example Proof 1

The crime example:

\[ \neg \text{American}(x) \lor \neg \text{Weapon}(y) \lor \neg \text{Sells}(x, y, z) \lor \neg \text{Hostile}(z) \lor \text{Criminal}(x) \]
\[ \neg \text{Missile}(x) \lor \neg \text{Owns}(\text{Nono}, x) \lor \text{Sells}(\text{West}, x, \text{Nono}) \]
\[ \neg \text{Missile}(x) \lor \text{Weapon}(x) \]
\[ \neg \text{Enemy}(x, \text{America}) \lor \text{Hostile}(x) \]
Example Proof 1

The crime example:

\[ \neg \text{American}(x) \lor \neg \text{Weapon}(y) \lor \neg \text{Sells}(x, y, z) \lor \neg \text{Hostile}(z) \lor \text{Criminal}(x) \]

\[ \neg \text{Missile}(x) \lor \neg \text{Owns}(\text{Nono}, x) \lor \text{Sells}(\text{West}, x, \text{Nono}) \]

\[ \neg \text{Missile}(x) \lor \text{Weapon}(x) \]

\[ \neg \text{Enemy}(x, \text{America}) \lor \text{Hostile}(x) \]

We prove \text{Criminal}(\text{West}) by adding

\[ \neg \text{Criminal}(\text{West}) \]

and deriving the empty clause \( \emptyset \).
Resolution Proof 1

Like backward chaining.
Example Proof 2 (with Skolemization)

Everyone who loves all animals is loved by someone.
Anyone who kills an animal is loved by no one.
Jack loves all animals.
Either Jack or Curiosity killed the cat, who is named Tuna.
Did Curiosity kill the cat?
Example Proof 2 (with Skolemization)

Everyone who loves all animals is loved by someone. Anyone who kills an animal is loved by no one. Jack loves all animals. Either Jack or Curiosity killed the cat, who is named Tuna. Did Curiosity kill the cat?

A. \( \forall x \ (\forall y \text{ Animal}(y) \Rightarrow \text{Loves}(x, y)) \Rightarrow \exists y \text{ Loves}(y, x) \)
Example Proof 2 (with Skolemization)

Everyone who loves all animals is loved by someone.
Anyone who kills an animal is loved by no one.
Jack loves all animals.
Either Jack or Curiosity killed the cat, who is named Tuna.
Did Curiosity kill the cat?

A. \[ \forall x \ (\forall y \ Animal(y) \Rightarrow Loves(x, y)) \Rightarrow \exists y \ Loves(y, x) \]

B. \[ \forall x \ (\exists z \ Animal(z) \land Kills(x, z)) \Rightarrow (\forall y \ \neg Loves(y, x)) \]
Everyone who loves all animals is loved by someone.
Anyone who kills an animal is loved by no one.
Jack loves all animals.
Either Jack or Curiosity killed the cat, who is named Tuna.
Did Curiosity kill the cat?

A. $\forall x \ (\forall y \ Animal(y) \Rightarrow Loves(x, y)) \Rightarrow \exists y \ Loves(y, x))$

B. $\forall x \ (\exists z \ Animal(z) \land Kills(x, z)) \Rightarrow (\forall y \ \neg Loves(y, x))$

C. $\forall x \ Animal(x) \Rightarrow Loves(Jack, x)$
Example Proof 2 (with Skolemization)

Everyone who loves all animals is loved by someone.
Anyone who kills an animal is loved by no one.
Jack loves all animals.
Either Jack or Curiosity killed the cat, who is named Tuna.
Did Curiosity kill the cat?

A. \( \forall x \ (\forall y \ Animal(y) \Rightarrow Loves(x, y)) \Rightarrow \exists y \ Loves(y, x) \)  
B. \( \forall x \ (\exists z \ Animal(z) \land Kills(x, z)) \Rightarrow (\forall y \ \neg Loves(y, x)) \)  
C. \( \forall x \ Animal(x) \Rightarrow Loves(Jack, x) \)  
D. \( Kills(Jack, Tuna) \lor Kills(Curiosity, Tuna) \)
Example Proof 2 (with Skolemization)

Everyone who loves all animals is loved by someone.  
Anyone who kills an animal is loved by no one.  
Jack loves all animals.  
Either Jack or Curiosity killed the cat, who is named Tuna.  
Did Curiosity kill the cat?

A. $\forall x \ (\forall y \ Animal(y) \Rightarrow Loves(x,y)) \Rightarrow \exists y \ Loves(y,x))$

B. $\forall x \ (\exists z \ Animal(z) \wedge Kills(x,z)) \Rightarrow (\forall y \ \neg Loves(y,x))$

C. $\forall x \ Animal(x) \Rightarrow Loves(Jack,x)$

D. $Kills(Jack,Tuna) \lor Kills(Curiosity,Tuna)$

E. $Cat(Tuna)$

F. $\forall x \ Cat(x) \Rightarrow Animal(x)$
Example Proof 2 (with Skolemization)

Everyone who loves all animals is loved by someone.
Anyone who kills an animal is loved by no one.
Jack loves all animals.
Either Jack or Curiosity killed the cat, who is named Tuna.
Did Curiosity kill the cat?

A. $\forall x \ (\forall y \ \text{Animal}(y) \Rightarrow \text{Loves}(x, y)) \Rightarrow \exists y \ \text{Loves}(y, x))$

B. $\forall x \ (\exists z \ \text{Animal}(z) \land \text{Kills}(x, z)) \Rightarrow (\forall y \ \neg \text{Loves}(y, x))$

C. $\forall x \ \text{Animal}(x) \Rightarrow \text{Loves}(\text{Jack}, x)$

D. $\text{Kills}(\text{Jack}, \text{Tuna}) \lor \text{Kills}(\text{Curiosity}, \text{Tuna})$

E. $\text{Cat}(\text{Tuna})$

F. $\forall x \ \text{Cat}(x) \Rightarrow \text{Animal}(x)$

¬G. $\neg \text{Kills}(\text{Curiosity}, \text{Tuna})$
Converting to CNF

A. \( \forall x \ (\forall y \ \text{Animal}(y) \Rightarrow \text{Loves}(x, y)) \Rightarrow (\exists y \ \text{Loves}(x, y)) \)
Converting to CNF

A.  \( \forall x \ ( \forall y \ Animal(y) \Rightarrow Loves(x, y)) \Rightarrow (\exists y \ Loves(x, y)) \)
Converting to CNF

A. \( \forall x \ (\forall y \ Animal(y) \Rightarrow Loves(x, y)) \Rightarrow (\exists y \ Loves(x, y)) \)

A1. \( Animal(F(x)) \lor Loves(G(x), x) \)

A2. \( \neg Loves(x, F(x)) \lor Loves(G(x), x) \)
Converting to CNF

A. \[ \forall x (\forall y \text{Animal}(y) \implies \text{Loves}(x, y)) \implies (\exists y \text{Loves}(x, y)) \]

A1. \[ \text{Animal}(F(x)) \lor \text{Loves}(G(x), x) \]

A2. \[ \neg \text{Loves}(x, F(x)) \lor \text{Loves}(G(x), x) \]

B. \[ \neg \text{Animal}(z) \lor \neg \text{Kills}(x, z) \lor \neg \text{Loves}(y, x) \]

C. \[ \neg \text{Animal}(x) \lor \text{Loves}(\text{Jack}, x) \]

D. \[ \text{Kills}(\text{Jack}, \text{Tuna}) \lor \text{Kills}(\text{Curiosity}, \text{Tuna}) \]

E. \[ \text{Cat}(\text{Tuna}) \]

F. \[ \neg \text{Cat}(x) \lor \text{Animal}(x) \]

\[ \neg G. \quad \neg \text{Kills}(\text{Curiosity}, \text{Tuna}) \]
Resolution Proof 2
Resolution Proof 2

\[
\neg \text{Loves}(x_2, F(x_2)) \quad \text{Loves}(\text{Jack}, x_1)
\]
Resolution Proof 2

\[
\neg \text{Loves}(x_2, F(x_2)) \quad \text{Loves}(\text{Jack}, x_1)
\]
Resolution Proof 2

\[ \neg \text{Loves}(x_2, F(x_2)) \]

\[ \text{Loves}(\text{Jack}, x_1) \]

\[ \theta = \{ x_1/F(\text{Jack}), x_2/\text{Jack} \} \]
Resolution Proof 2

\[ \neg \text{Loves}(x_2, F(x_2)) \quad \text{Loves}(\text{Jack}, x_1) \]

\[ \theta = \{x_1/F(\text{Jack}), x_2/\text{Jack}\} \]
Suppose Curiosity did not kill Tuna. We know that either Jack or Curiosity did; thus Jack must have. Now, Tuna is a cat and cats are animals, so Tuna is an animal. Because anyone who kills an animal is loved by no one, we know that no one loves Jack. On the other hand, Jack loves all animals, so someone loves him; so we have a contradiction. Therefore, Curiosity killed the cat.
Completeness of Resolution

**Theorem**  If a set $S$ of sentences is unsatisfiable, then resolution will always be able to derive a contradiction.

- Not all logical consequences of $S$ can be generated using resolution.

- A sentence entailed by $S$ can always be established using resolution.
Completeness of Resolution

**Theorem**  If a set $S$ of sentences is unsatisfiable, then resolution will always be able to derive a contradiction.

- Not all logical consequences of $S$ can be generated using resolution.

- A sentence entailed by $S$ can always be established using resolution.

We can use resolution to find all answers to a question $Q(x)$ by proving that $KB \land \neg Q(x)$ is unsatisfiable.