Representación de Conocimiento (3)

Carlos Hurtado L.
Depto de Ciencias de la Computación, Universidad de Chile

Clase Pasada

- Regla de Resolución.
- Refutación mediante resolución.

Clase pasada: Resolución

- Introducido por Alan Robin en 1965.
- Procedimiento de prueba correcto y completo para Cálculo de Proposiciones
- Se generaliza a Cálculo de Predicados de Primer Orden
- Base de motor de inferencia en Prolog y demostradores de teoremas, entre otros sistemas.
  - Carine, Gandalf, SPASS.

Clase Pasada: Regla de Resolución

$$\frac{(p, q_1, \ldots, q_n) \quad (\neg p, r_1, \ldots, r_n)}{(q_1, \ldots, q_k, r_1, \ldots, r_n)}$$

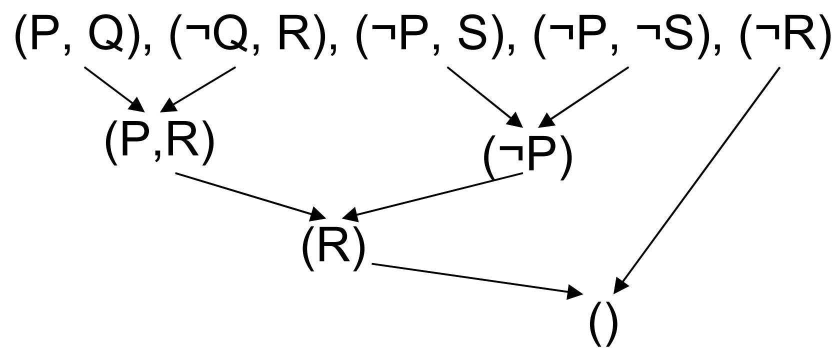
Resolvente

Otra forma de ver la regla de resolución:

$$\frac{(\neg q_1 \wedge \ldots \wedge \neg q_k \Rightarrow p) \quad (p \Rightarrow r_1 \vee \ldots \vee r_n)}{(\neg q_1 \wedge \ldots \wedge \neg q_k \Rightarrow r_1 \vee \ldots \vee r_n)}$$

Clase Pasada: Ejemplo

- KB={ (P, Q), (¬Q, R), (¬P, S), (¬P, ¬S)}, α=R.
- Agregamos (¬α) a KB
- Aplicamos regla de resolución para construir una refutación:



- Lo que genera la siguiente refutación:
  (P, Q), (¬Q, R), (P,R), (¬P, S), (¬P, ¬S), (¬P), (R), (¬R),()
- Por lo tanto, KB $\models$ α.

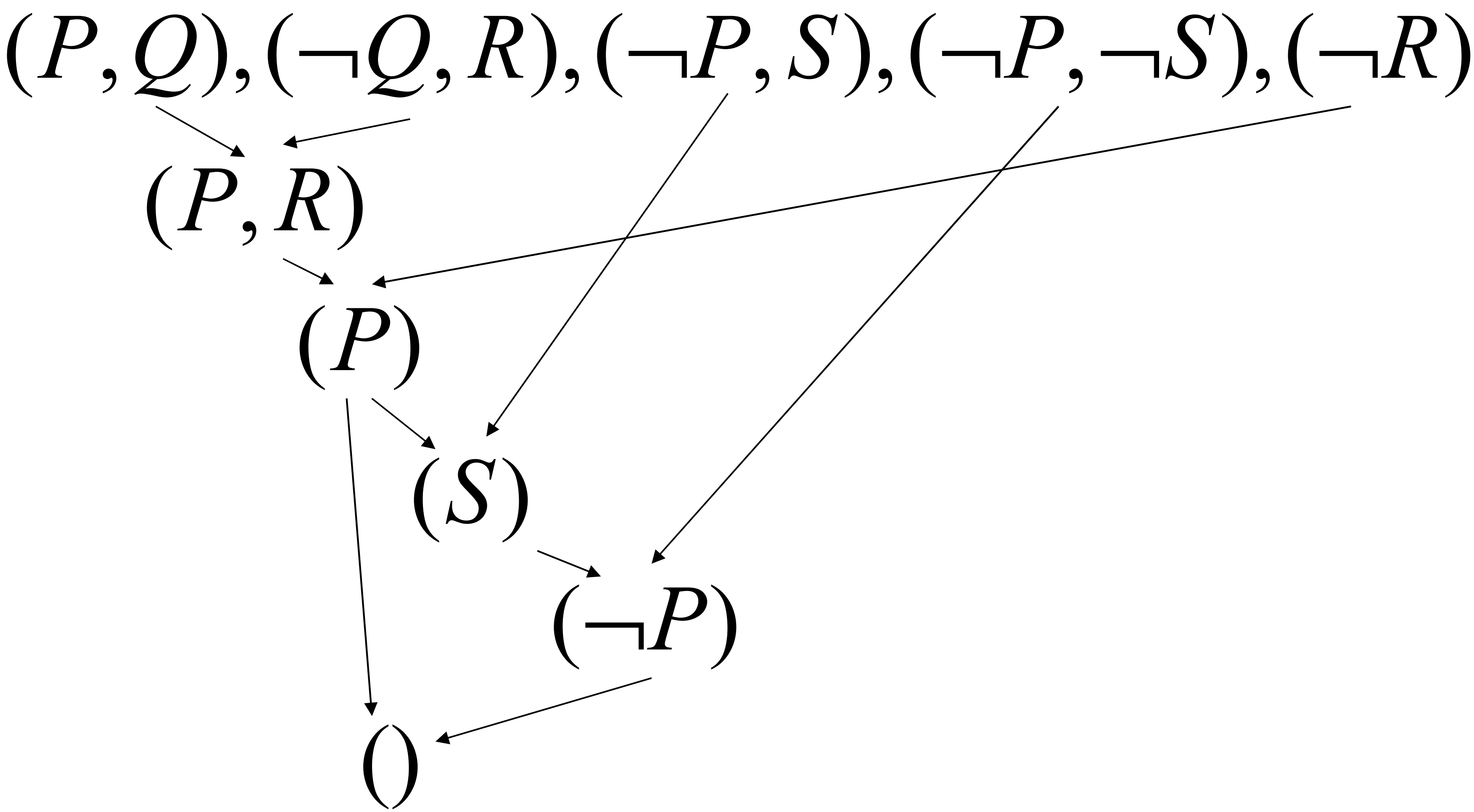
Clase Pasada: Algoritmos de Resolución

- Existen distintas formas de implementar el proceso de resolución.
- Estrategias de ordenamiento: definen el orden en que se realizan las resoluciones
    - Primero en Anchura
    - Primero en Profundidad
- Estrategias de refinamiento:
  - Imponen restricciones en las resoluciones a realizar:
    - Conjunto Soporte
    - Resolución Lineal
    - Resolución Unitaria

Clase Pasada: Primero en Anchura

$$\frac{\dfrac{\dfrac{(P,Q),(\neg Q,R),(\neg P,S),(\neg P,\neg S),(\neg R)}{(P,R),(Q,S),(Q,\neg S),(\neg Q),(\neg P)}}{(R),(Q),(S),(\neg S),(R,S)(R,\neg S),(P),(Q,\neg P)}}{....()}$$

Clase Pasada: Primero en Profundidad



Control sorpresa

Encuentre una refutación para las siguientes cláusulas.

$$A \vee C$$

$$A \vee \neg C \vee F$$

$$\neg A \vee E \vee F$$

$$\neg C \vee \neg E \vee F$$

$$\neg A \vee \neg F$$

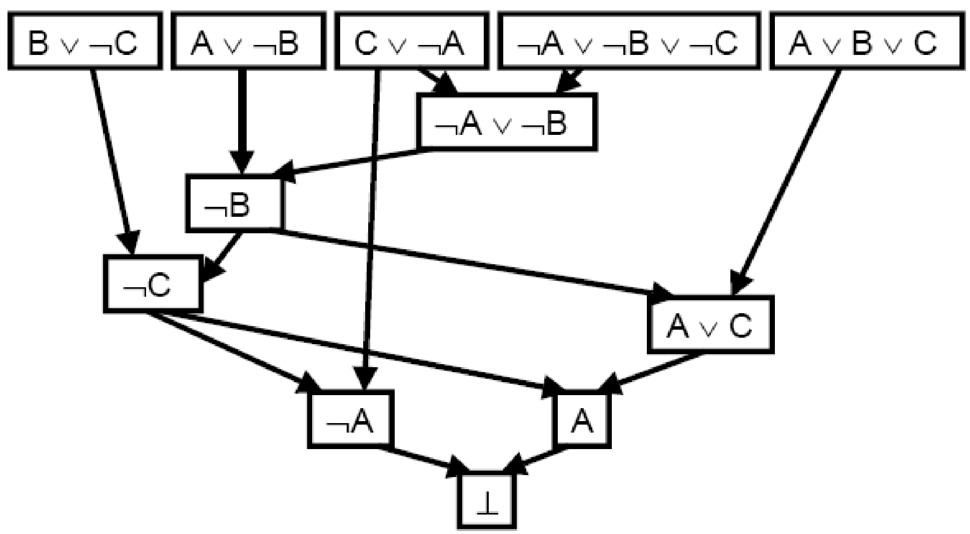
$$C \vee F$$

Estrategias de refinamiento

- Imponen restricciones en las resoluciones a realizar.
  - Eliminación de cláusulas
  - Conjunto Soporte
  - Resolución Lineal
  - Resolución Unitaria
- Podan el espacio de posibles resoluciones

Eliminación de Cláusulas

- En general, en un proceso de resolución no podemos eliminar cláusulas generadas ya que pueden ser usadas en resoluciones posteriores.



Eliminación de Cláusulas

- Recordar que escribimos las cláusulas como conjunto de literales. Ejemplo: ($\neg$c,b,a).
- Supongamos que no tenemos cláusulas triviales como ($\neg$c,c,a).

Eliminación de Cláusulas

- Lema: Dadas dos cláusulas $C1$, $C2$: $C1 \models C2$ ssi $C1 \subseteq C2$
- Demo (si): trivial: Si $C1 \subseteq C2$ entonces $Mod(C1) \subseteq Mod(C2)$
- Demo (sólo si):
  - Supongamos que no se cumple. Luego existe c en $C1$ tal que c no está en $C2$.
  - Construimos una interpretación I, tal que I(c)=V y para todo c' en $C2$, I(c')=F.
  - Es fácil verificar que I($C1$)=V e I($C2$)=F.
  - Esto contradice que $C1 \models C2$.

Eliminación de Cláusulas

- Dado un conjunto de cláusulas S, sea Base(S) el conjunto de cláusulas en S minimales (con respecto a inclusión).
- Ejemplo,
  - S={(a,b), (¬c,b,a), (b,c),(b,c,d)},
  - Base(S)={(a,b), (b,c)},
- Usando el lema anterior es fácil verificar que Mod(S)=Mod(Base(S)):

  Ej. Mod(S)

  = Mod((a,b)) inter Mod((¬c,b,a)) inter Mod((b,c)) inter Mod((b,c,d))

  = Mod((a,b)) inter Mod((b,c))

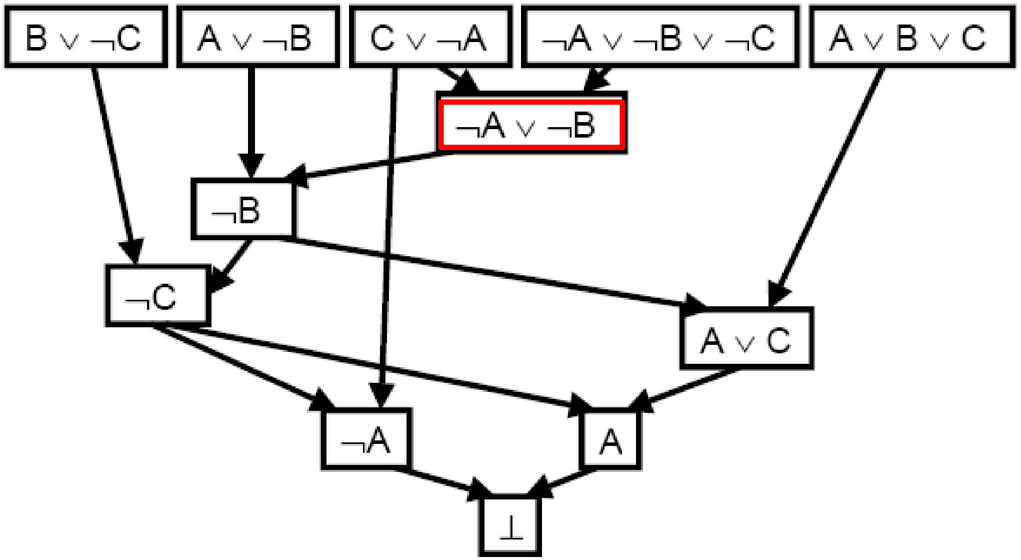
  =Mod(Base(S))

Eliminación de Cláusulas

- Ejemplo,
  - $S=\{(a,b), (\neg c,b,a), (b,c),(b,c,d)\}$,
  - $Base(S)=\{(a,b), (b,c)\}$,
- Luego Base(S) y S son lógicamente equivalentes y por lo tanto podemos aplicar el proceso de resolución sobre Base(S) en lugar de S.

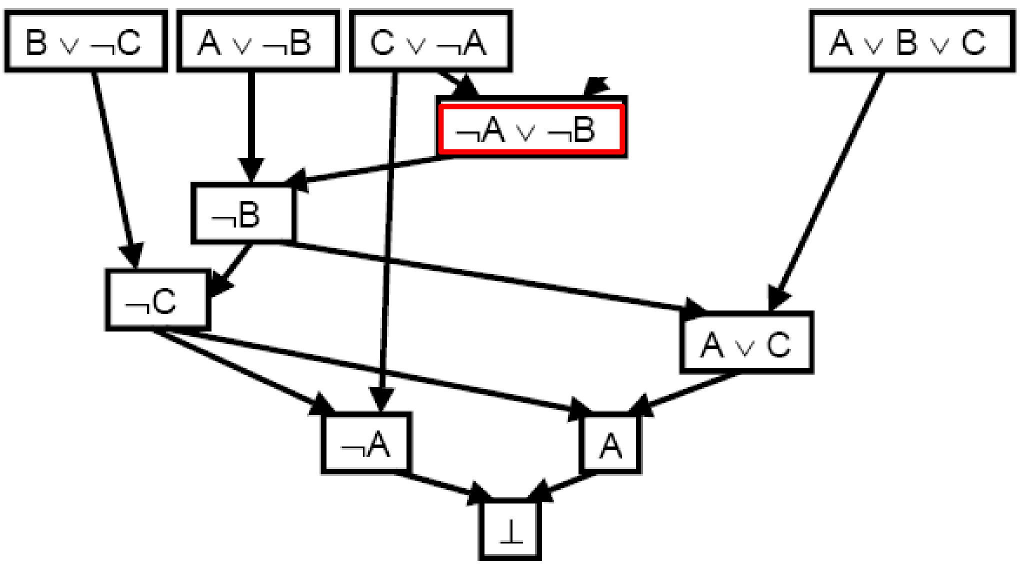
Eliminación de Cláusulas

- El resultado del proceso de resolución no cambia si eliminamos cláusulas no minimales. Ejemplo:



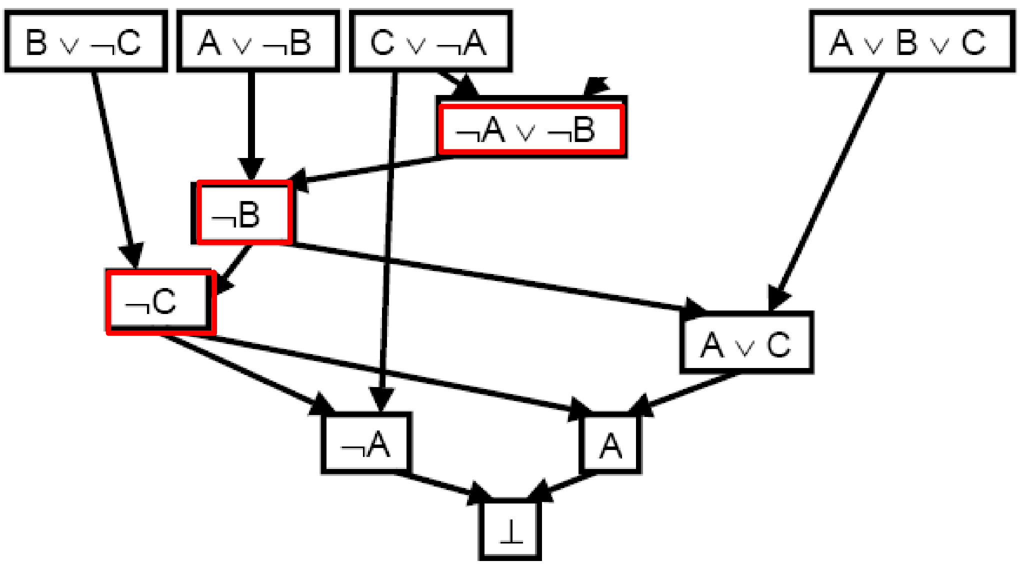
Eliminación de Cláusulas

- El resultado del proceso de resolución no cambia si eliminamos cláusulas no minimales. Ejemplo:



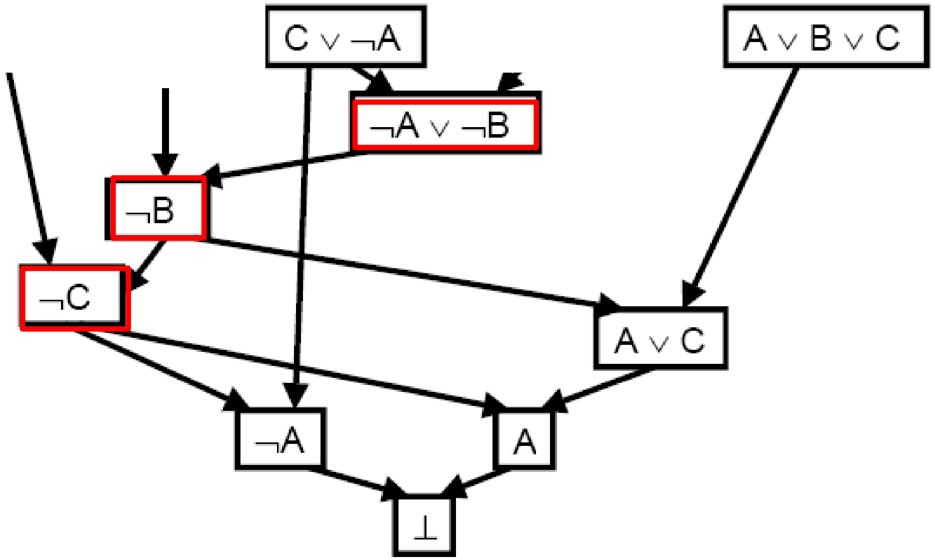
Eliminación de Cláusulas

- El resultado del proceso de resolución no cambia si eliminamos cláusulas no minimales. Ejemplo:



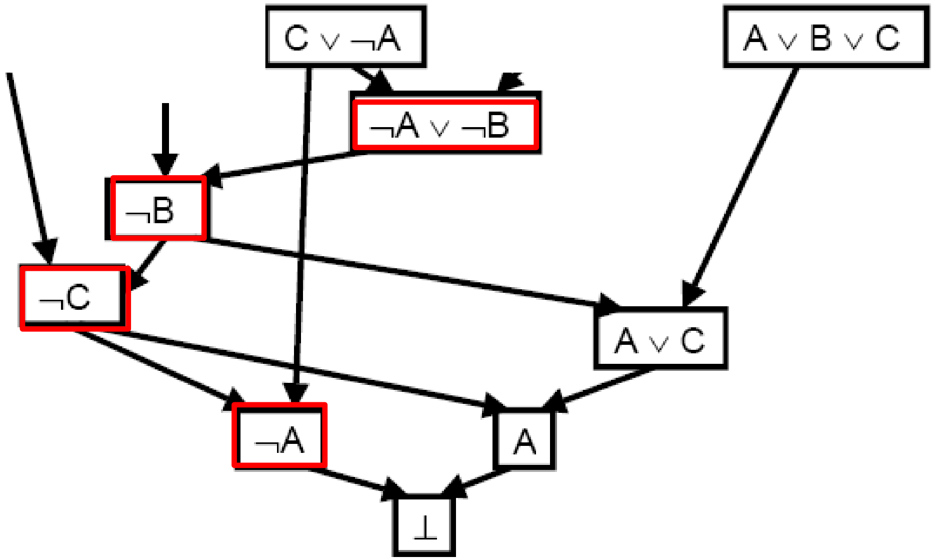
Eliminación de Cláusulas

- El resultado del proceso de resolución no cambia si eliminamos cláusulas no minimales. Ejemplo:



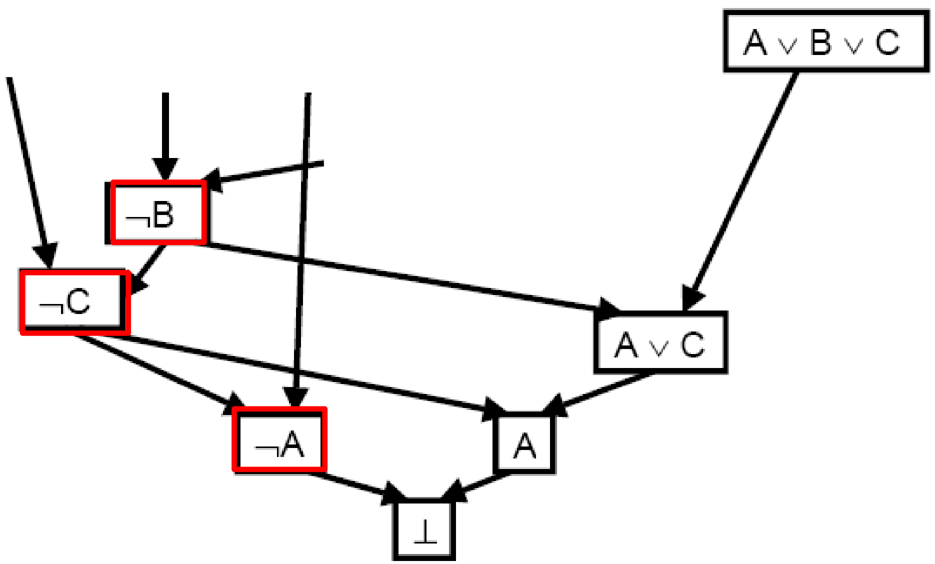
Eliminación de Cláusulas

- El resultado del proceso de resolución no cambia si eliminamos cláusulas no minimales. Ejemplo:



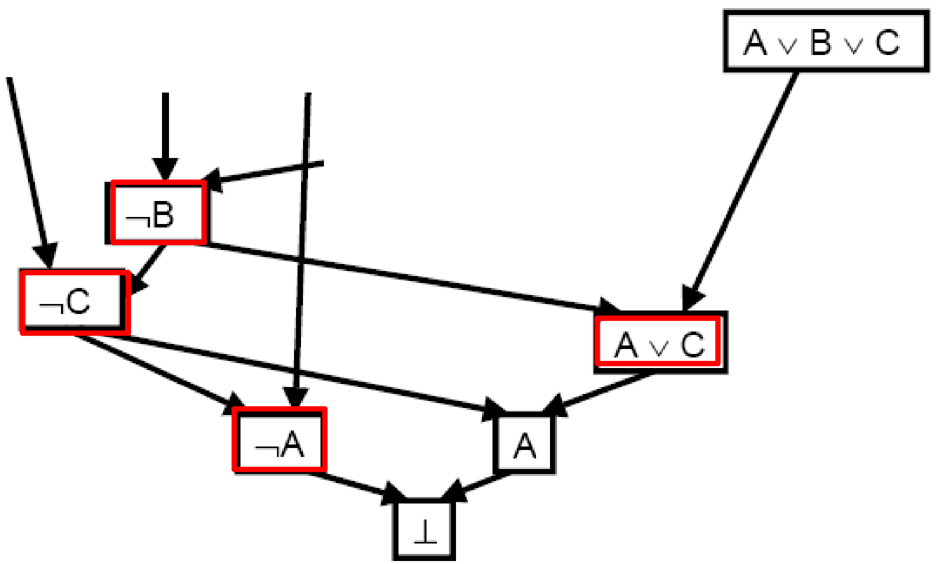
Eliminación de Cláusulas

- El resultado del proceso de resolución no cambia si eliminamos cláusulas no minimales. Ejemplo:



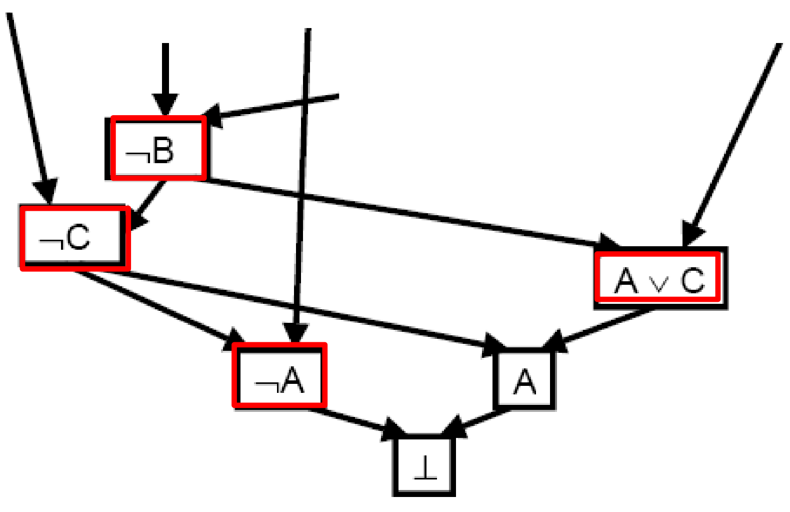
Eliminación de Cláusulas

- El resultado del proceso de resolución no cambia si eliminamos cláusulas no minimales. Ejemplo:



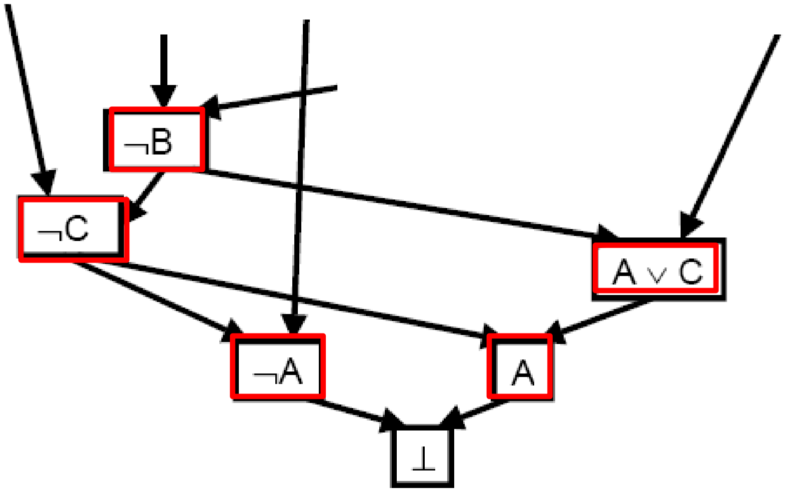
Eliminación de Cláusulas

- El resultado del proceso de resolución no cambia si eliminamos cláusulas no minimales. Ejemplo:



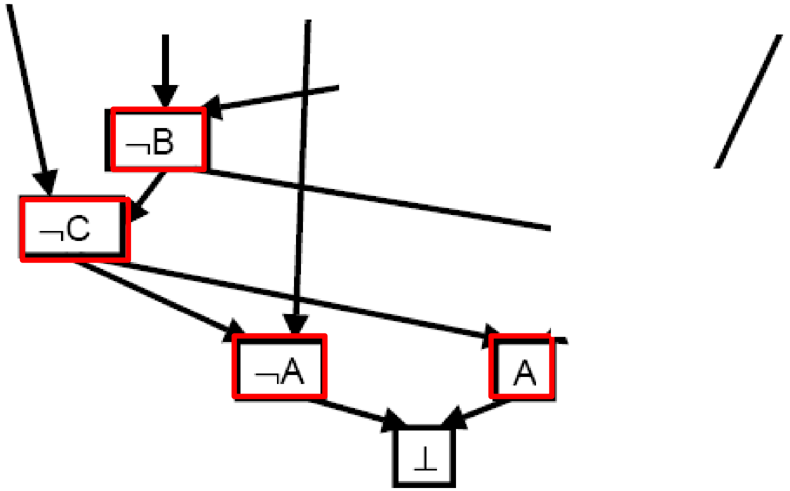
Eliminación de Cláusulas

- El resultado del proceso de resolución no cambia si eliminamos cláusulas no minimales. Ejemplo:



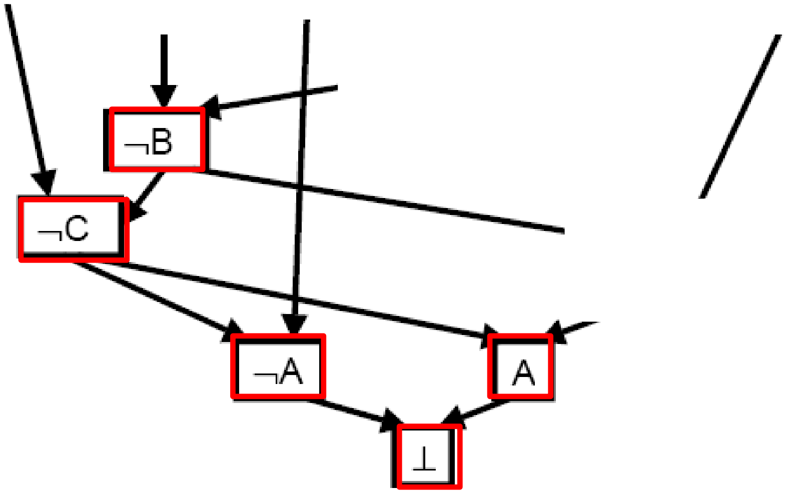
Eliminación de Cláusulas

- El resultado del proceso de resolución no cambia si eliminamos cláusulas no minimales. Ejemplo:



Eliminación de Cláusulas

- El resultado del proceso de resolución no cambia si eliminamos cláusulas no minimales. Ejemplo:



Estrategia del Conjunto Soporte

Recordar que debemos verificar si KB $\models \alpha$:

- Si KB es satisfacible entonces toda refutación para (KB $\wedge \neg\alpha$) debe contener alguna cláusula de $\neg\alpha$
- Idea: guiar la búsqueda de la cláusula vacía usando ($\neg\alpha$).

Conjunto Soporte

- Conjunto soporte:
  - Una cláusula U es descendiente de una cláusula W si
    - (i) U es resolvente de W y otra clásula o
    - (ii) U es resolvente de una cláusula descendiente de W y otra cláusula
  - Conjunto soporte: cláusulas en $(\neg\alpha)$ o descendientes de cláusulas en $(\neg\alpha)$.
- Estrategia Conjunto Soporte: al resolver tomar en cuenta al menos una cláusula en conjunto soporte.

Estrategia del Conjunto Soporte: Ejemplo

Sea KB= {(B), (¬R),(¬B,¬O,R)}; y α = ¬O.

{(B), (¬R),(¬B,¬O,R),(O)}

(¬B,R)

(¬B)

()

Resolución Lineal

- Sólo permite resoluciones que usan al menos una cláusula en $(KB \wedge \neg\alpha)$
- Ejemplo (es el mismo que el anterior):

$$\{(B), (\neg R),(\neg B,\neg O,R),(O)\}$$

$$(\neg B,R)$$

$$(\neg B)$$

$$()$$

Resolución Lineal

- Problema: refutación basada en resolución lineal no es completa

$\{ (P, Q)\ (P, \neg Q)\ (\neg P, Q)\ (\neg P, \neg Q) \}$

(P) $\quad\quad$ $(\neg P)$

$()$

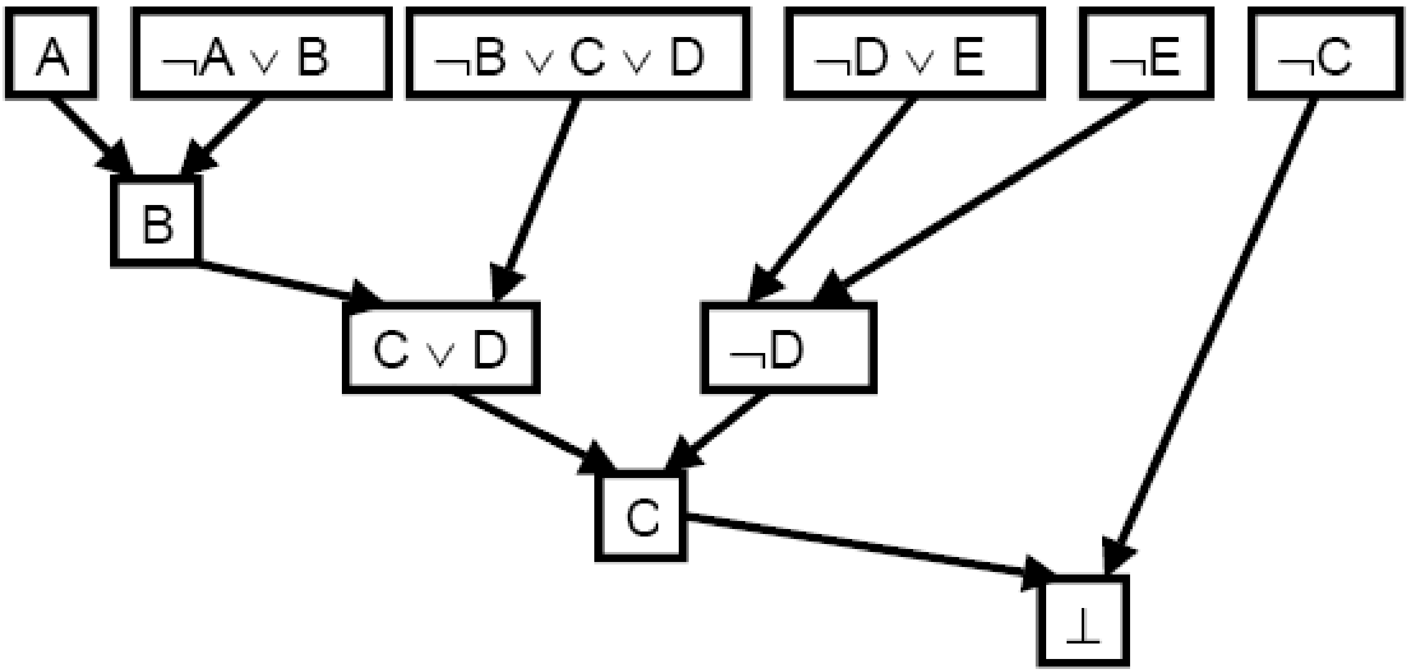
Ejercicio: complejidad de resolución lineal

- Resolución lineal es más eficiente que resolución.
- Investigar la complejidad de resolución lineal (en muchos artículos resolución lineal se denomina “input resolution”).

Resolución Unitaria

- Sólo permite resoluciones donde al menos una de las cláusulas tiene un literal (cláusula unitaria)
- En este caso decimos que la refutación (si la hay) es unitaria

Ejemplo: Resolución Unitaria



Complejidad de Resolución Unitaria

- En una resolución C1,U→C3, dode U es la cláusula unitaria, siempre podemos eliminar C1 del conjunto de cláusulas (¿Por qué?)
- Dado un conjunto de cláusulas S, sea Largo(S) su número de literales (incluyendo repetidos).
- Ejemplo, si S es { (P, Q) (P, ¬Q) (¬P, Q) (¬P, ¬Q) }, Largo(S)=8.
- Luego en un proceso de resolución unitaria + eliminación de cláusulas no tenemos más de Largo(S) resoluciones.
  - Recordar que en un proceso de resolución arbitrario podemos generar del orden de $2^{\text{Largo(S)}}$ resoluciones.

Resolución Unitaria vs. Resolución Lineal

- Teorema: Un conjunto de cláusulas tiene una refutación unitaria ssi tiene una refutación lineal
- Resolución lineal no es completa.
- Por lo tanto resolución unitaria tampoco es completa

Ejemplo: Refutación Lineal para el ejemplo anterior