FUNDAMENTOS LÓGICOS DE LA INFORMÁTICA

Tercera sesión de prácticas

PRIMER CURSO DEL GRADO DE INGENIERÍA INFORMÁTICA

PRIMER CUATRIMESTRE

2020-21

Índice del documento

El Problema SAT

- **Problema de la Satisfacibilidad booleana (en adelante, SAT), se puede definir como:** dada una oración lógica (f.b.f. en L0) formada por una serie de proposiciones atómicas (variables lógicas o booleanas), ¿existe una combinación de atribuciones de valores de verdad a las proposiciones atómicas, es decir, una interpretación, en la que la evaluación de la oración resulte verdadera?
- El problema tiene su complementario, **el Problema UNSAT**: ¿cuál es la combinación de atribuciones de valores de las proposiciones atómicas que hacen a la oración falsa?
- **Es un problema de decision!!!!** SAT es un problema fundamental en lógica formal, teoría de computación, inferencia, razonamiento automático, ingeniería VLSI, entre otros. Stephen Cook, en 1971, demostró que el problema SAT era un problema NP-COMPLETO.
- **Realmente un problema de la clase NP-COMPLETO es un problema intratable**. No hay un algoritmo mejor que realizar una búsqueda exhaustiva de todas las posibilidades en la búsqueda de una solución al problema. Por lo tanto, buenos algoritmos para SAT son indudablemente de gran utilidad, dado que es poco probable que se pueda encontrar un algoritmo de tiempo polinómico para resolver el problema NP-COMPLETO.
- **Un caso particular del problema SAT es el 3-SAT** en el que el objetivo del problema es determinar si una oración lógica (f.b.f. en L0) en forma normal conjuntiva con exactamente tres literales por cláusula, es satisfacible.

Las Técnicas SAT

- **En L0 el problema SAT y el problema UNSAT son decidibles**.
- Algoritmos completos (1): Los que proporcionando una interpretación, ésta es un modelo (interpretación que hace VERDADERO la evaluación, o un contramodelo (interpretación que hace FALSO la evaluación).
  - Tablas de verdad.
  - Algoritmo DP (Davis; Martin; Putnam; y Hillary. 1960)
  - Algoritmo DPLL (Davis; Putnam; Logemann; y Loveland. 1962)
  - Árboles semánticos.
  - Tableaux semánticos.
  - ...
- Algoritmos no Completos (2):
  - GSAT (Greedy SAT)
  - WalkSat (GSAT con probabilidades)
  - SA-SAT (Simulating Annealing SAT)
  - ...

(1) Determinan tanto la satisfacibilidad como la insatisfacibilidad.

(2) Puede que no encuentre un modelo, aún existiendo.

Tipos de Técnicas SAT (I)

- Las Técnicas que vamos a estudiar para resolver el problema SAT son aquellas que implementan aproximaciones fértiles, generalmente, mediante búsquedas en profundidad, a la pregunta (decisión) de si para una fórmula proposicional cualquiera existirá una interpretación que la haga verdadero (o, falso).

  - Si una interpretación determinada hace a la fórmula verdadero, diremos que la oración es SATISFACIBLE.
  - No sabremos que una oración será TAUTOLOGÍA, en general, si no podemos asegurar que en todas las interpretaciones la fórmula es evaluada como verdadero. En todo caso, no será TAUTOLOGÍA si es FALSEABLE.
  - Si una interpretación determinada hace a la fórmula falso, diremos que la oración es FALSEABLE.
  - No sabremos que una oración es una CONTRADICCIÓN (Insatisfacible), en general, si no podemos garantizar que será falso (FALSEABLE) en todas las interpretaciones.

Tipos de Técnicas SAT (y II)

- Vamos a clasificar las Técnicas SAT en función del tipo de oración de partida al aplicar la técnica. Básicamente, tenemos dos tipos de técnicas: el primero, aquellas que parten de oraciones expresadas en su FORMA CLAUSAL (previa transformación en FNC); y el segundo, aquellas que parten de la fórmula original sin necesidad de transformar en FNC.

  - Las Técnicas SAT aplicables a partir de FORMAS CLAUSALES:
    - Algoritmo DPLL
    - Algoritmo de Resolución
  - Las Técnicas SAT aplicables a fórmulas cualesquiera:
    - Tablas de Verdad
    - Árboles Semánticos
    - Tableaux Semánticos

Formas Normales Conjuntivas y Cláusulas (I)

Definiciones

- Un **literal** es una proposición atómica o su negada
- Una **cláusula** es la disyunción de literales.
- Una cláusula es un conjunto finito de literales.
- Una cláusula con sólo un literal se llama cláusula unitaria.
- Una cláusula sin literales se llama **cláusula vacía** y se denota por ☐.
- Una fórmula, **α**, está en forma normal conjuntiva (FNC) si responde a la expresión **ξ** $= (\wedge_i (\vee_j P_{ij}))$ donde P_{ij} son literales (de L0).
- Un conjunto clausal o clausulado es un conjunto de cláusulas expresadas como conjuntos de literales. Una fórmula está en forma clausal si se expresa como un conjunto clausal.
- El conjunto clausal **{{**¬p, q**}**; **{** r**}}** se corresponde con la fórmula $((p \rightarrow q) \wedge r)$
- La forma clausal de la FNC: $((p \vee q) \wedge \neg p)$, es **{{**p, q**}**; **{**¬p**}}.**
- **La cláusula vacía ☐, la disyunción de ningún literal, es una oración insatisfacible. POR DEFINICIÓN!!!!**
- **El conjunto vacío de oraciones {Ø} (ya sea o no de cláusulas), es un conjunto satisfacible. YA LO VIMOS AL HABLAR DE CONJUNTOS DE ORACIONES!!!!**

Formas Normales Conjuntivas y Cláusulas (II)

Construcción de una FNC aplicando transformaciones por equivalencia lógica. Vamos a transformar ahora una oración no Tautológica, es decir, una oración cuyo conjunto clausulado o Forma Clausal no es el conjunto vacío de cláusulas.

- Tengamos ahora la siguiente oración: $((\neg p \vee \neg q \rightarrow p \wedge r) \wedge (r \rightarrow \neg q))$

Formas Normales Conjuntivas y Cláusulas (III)

Construcción de una FNC aplicando transformaciones por equivalencia lógica. Vamos a transformar ahora una oración no Tautológica, es decir, una oración cuyo conjunto clausulado o Forma Clausal no es el conjunto vacío de cláusulas.

- Tengamos ahora la siguiente oración: $((\neg p \vee \neg q \rightarrow p \wedge r) \wedge (r \rightarrow \neg q))$
    - Aplicando la interdefinición: $((\neg(\neg p \vee \neg q) \vee (p \wedge r)) \wedge (\neg r \vee \neg q))$

Formas Normales Conjuntivas y Cláusulas (IV)

Construcción de una FNC aplicando transformaciones por equivalencia lógica. Vamos a transformar ahora una oración no Tautológica, es decir, una oración cuyo conjunto clausulado o Forma Clausal no es el conjunto vacío de cláusulas.

- Tengamos ahora la siguiente oración: $\mathbf{((\neg p \vee \neg q \rightarrow p \wedge r) \wedge (r \rightarrow \neg q))}$
  - Aplicando la interdefinición: $((\neg(\neg p \vee \neg q) \vee (p \wedge r)) \wedge (\neg r \vee \neg q))$
  - Aplicando la ley de D'Morgan: $(((\neg\neg p \wedge \neg\neg q) \vee (p \wedge r)) \wedge (\neg r \vee \neg q))$

Formas Normales Conjuntivas y Cláusulas (V)

Construcción de una FNC aplicando transformaciones por equivalencia lógica. Vamos a transformar ahora una oración no Tautológica, es decir, una oración cuyo conjunto clausulado o Forma Clausal no es el conjunto vacío de cláusulas.

- Tengamos ahora la siguiente oración: $((\neg p \lor \neg q \to p \land r) \land (r \to \neg q))$
  - Aplicando la interdefinición: $((\neg(\neg p \lor \neg q) \lor (p \land r)) \land (\neg r \lor \neg q))$
  - Aplicando la ley de D'Morgan: $(((\neg\neg p \land \neg\neg q) \lor (p \land r)) \land (\neg r \lor \neg q))$
  - Aplicando la idempotencia: $(((p \land q) \lor (p \land r)) \land (\neg r \lor \neg q))$

Formas Normales Conjuntivas y Cláusulas (VI)

Construcción de una FNC aplicando transformaciones por equivalencia lógica. Vamos a transformar ahora una oración no Tautológica, es decir, una oración cuyo conjunto clausulado o Forma Clausal no es el conjunto vacío de cláusulas.

- Tengamos ahora la siguiente oración: $\mathbf{((\neg p \lor \neg q \rightarrow p \land r) \land (r \rightarrow \neg q))}$
  - Aplicando la interdefinición: $((\neg(\neg p \lor \neg q) \lor (p \land r)) \land (\neg r \lor \neg q))$
  - Aplicando la ley de D'Morgan: $(((\neg\neg p \land \neg\neg q) \lor (p \land r)) \land (\neg r \lor \neg q))$
  - Aplicando la idempotencia: $(((p \land q) \lor (p \land r)) \land (\neg r \lor \neg q))$
  - Aplicando la propiedad distributiva de la disyunción con respecto a la conjunción en el primer paréntesis: $(((p \lor p) \land (p \lor r) \land (q \lor p) \land (q \lor r)) \land (\neg r \lor \neg q))$

Formas Normales Conjuntivas y Cláusulas (VII)

Construcción de una FNC aplicando transformaciones por equivalencia lógica. Vamos a transformar ahora una oración no Tautológica, es decir, una oración cuyo conjunto clausulado o Forma Clausal no es el conjunto vacío de cláusulas.

- Tengamos ahora la siguiente oración: $\mathbf{((\neg p \lor \neg q \rightarrow p \land r) \land (r \rightarrow \neg q))}$
  - Aplicando la interdefinición: $((\neg(\neg p \lor \neg q) \lor (p \land r)) \land (\neg r \lor \neg q))$
  - Aplicando la ley de D'Morgan: $(((\neg\neg p \land \neg\neg q) \lor (p \land r)) \land (\neg r \lor \neg q))$
  - Aplicando la idempotencia: $(((p \land q) \lor (p \land r)) \land (\neg r \lor \neg q))$
  - Aplicando la propiedad distributiva de la disyunción con respecto a la conjunción en el primer paréntesis: $(((p \lor p) \land (p \lor r) \land (q \lor p) \land (q \lor r)) \land (\neg r \lor \neg q))$
  - Simplificando paréntesis por asociatividad: $((p \lor p) \land (p \lor r) \land (q \lor p) \land (q \lor r) \land (\neg r \lor \neg q))$

Formas Normales Conjuntivas y Cláusulas (VIII)

Construcción de una FNC aplicando transformaciones por equivalencia lógica. Vamos a transformar ahora una oración no Tautológica, es decir, una oración cuyo conjunto clausulado o Forma Clausal no es el conjunto vacío de cláusulas.

- Tengamos ahora la siguiente oración: $\mathbf{((\neg p \lor \neg q \rightarrow p \land r) \land (r \rightarrow \neg q))}$
  - Aplicando la interdefinición: $\mathbf{((}\neg(\neg p \lor \neg q) \lor (p \land r)\mathbf{))} \land (\neg r \lor \neg q\mathbf{))}$
  - Aplicando la ley de D'Morgan: $\mathbf{(((}\neg\neg p \land \neg\neg q) \lor (p \land r)\mathbf{))} \land (\neg r \lor \neg q\mathbf{))}$
  - Aplicando la idempotencia: $\mathbf{(((}p \land q) \lor (p \land r)\mathbf{))} \land (\neg r \lor \neg q\mathbf{))}$
  - Aplicando la propiedad distributiva de la disyunción con respecto a la conjunción en el primer paréntesis: $\mathbf{(((}p \lor p) \land (p \lor r) \land (q \lor p) \land (q \lor r)\mathbf{))} \land (\neg r \lor \neg q\mathbf{))}$
  - Simplificando paréntesis por asociatividad: $\mathbf{((}p \lor p) \land (p \lor r) \land (q \lor p) \land (q \lor r) \land (\neg r \lor \neg q\mathbf{))}$
  - Simplificando al eliminar literales iguales: $((p) \land (p \lor r) \land (q \lor p) \land (q \lor r) \land (\neg r \lor \neg q))$

Formas Normales Conjuntivas y Cláusulas (IX)

Construcción de una FNC aplicando transformaciones por equivalencia lógica. Vamos a transformar ahora una oración no Tautológica, es decir, una oración cuyo conjunto clausulado o Forma Clausal no es el conjunto vacío de cláusulas.

- Tengamos ahora la siguiente oración: $\mathbf{((\neg p \vee \neg q \rightarrow p \wedge r) \wedge (r \rightarrow \neg q\,))}$
  - Aplicando la interdefinición: $\mathbf{((}\neg(\neg p \vee \neg q) \vee (p \wedge r)\mathbf{)} \wedge (\neg\, r \vee \neg q\,)\mathbf{)}$
  - Aplicando la ley de D'Morgan: $\mathbf{(((}\neg\neg p \wedge \neg\neg q) \vee (p \wedge r)\mathbf{)} \wedge (\neg\, r \vee \neg q\,)\mathbf{)}$
  - Aplicando la idempotencia: $\mathbf{(((}p \wedge q) \vee (p \wedge r)\mathbf{)} \wedge (\neg\, r \vee \neg q\,)\mathbf{)}$
  - Aplicando la propiedad distributiva de la disyunción con respecto a la conjunción en el primer paréntesis: $\mathbf{(((}p \vee p) \wedge (p \vee r) \wedge (q \vee p) \wedge (q \vee r)\mathbf{)} \wedge (\neg\, r \vee \neg q\,)\mathbf{)}$
  - Simplificando paréntesis por asociatividad: $\mathbf{((}p \vee p) \wedge (p \vee r) \wedge (q \vee p) \wedge (q \vee r) \wedge (\neg\, r \vee \neg q\,)\mathbf{)}$
  - Simplificando al eliminar literales iguales: $\mathbf{((}p) \wedge (p \vee r) \wedge (q \vee p) \wedge (q \vee r) \wedge (\neg\, r \vee \neg q\,)\mathbf{)}$
  - Y, finalmente, subsumiendo cláusulas: $\mathbf{((p) \wedge (q \vee r) \wedge (\neg\, r \vee \neg q\,))}$

Formas Normales Conjuntivas y Cláusulas (y X)

Construcción de una FNC aplicando transformaciones por equivalencia lógica. Vamos a transformar ahora una oración no Tautológica, es decir, una oración cuyo conjunto clausulado o Forma Clausal no es el conjunto vacío de cláusulas.

- Tengamos ahora la siguiente oración: $\mathbf{((\neg p \vee \neg q \rightarrow p \wedge r) \wedge (r \rightarrow \neg q))}$
  - Aplicando la interdefinición: $((\neg(\neg p \vee \neg q) \vee (p \wedge r)) \wedge (\neg r \vee \neg q))$
  - Aplicando la ley de D'Morgan: $(((\neg\neg p \wedge \neg\neg q) \vee (p \wedge r)) \wedge (\neg r \vee \neg q))$
  - Aplicando la idempotencia: $(((p \wedge q) \vee (p \wedge r)) \wedge (\neg r \vee \neg q))$
  - Aplicando la propiedad distributiva de la disyunción con respecto a la conjunción en el primer paréntesis: $(((p \vee p) \wedge (p \vee r) \wedge (q \vee p) \wedge (q \vee r)) \wedge (\neg r \vee \neg q))$
  - Simplificando paréntesis por asociatividad: $((p \vee p) \wedge (p \vee r) \wedge (q \vee p) \wedge (q \vee r) \wedge (\neg r \vee \neg q))$
  - Simplificando al eliminar literales iguales: $((p) \wedge (p \vee r) \wedge (q \vee p) \wedge (q \vee r) \wedge (\neg r \vee \neg q))$
  - Y, finalmente, subsumiendo cláusulas: $\mathbf{((p) \wedge (q \vee r) \wedge (\neg r \vee \neg q))}$
  - **Esta es la FNC**, con la que podremos resolver la satisfacibilidad de la oración de origen. Sabemos que la **oración de partida NO es una Tautología. La asignación v(p) = F, la hace Falseable. Veremos si es Satisfacible!!!**
- **Su Forma Clausal es: $\{\{p\}; \{q, r\}; \{\neg r, \neg q\}\}$**

La Propagación de Literales (I)

- Cuando tenemos una fórmula en su expresión FNC = **ξ**, la acción de **propagar un determinado literal (ℓ), consiste en simplificar** dicha expresión teniendo en cuenta la asignación a dicho literal del valor VERDADERO. La nueva FNC la denotaremos por **ξ (ℓ).**
    - Así, las cláusulas en las que aparezca dicho literal tendremos la disyunción de la constante **V** con el resto de los literales, lo que es equivalente lógicamente a VERDAD. Al ser la constante **V** el elemento neutro de la conjunción, dichas cláusulas pueden cancelarse o suprimirse de la nueva FNC = **ξ (ℓ).**
    - Al mismo tiempo, en las cláusulas en las que aparezca dicho literal negado, lo sustituiremos por la constante **F**, con lo que nos aparecerá una disyunción de dicha constante con el resto de la cláusula. Dado que la constante **F** es el elemento neutro para la disyunción, la ocurrencia de dicho literal negado puede suprimirse de las cláusulas en las que aparezca, culminando la formación de la nueva FNC = **ξ (ℓ).**

La Propagación de Literales (II)

Apliquemos la Propagación de Literales a una oración de la que conocemos su FNC

- Si tenemos la oración: $\boldsymbol{\alpha = ((\neg p \vee \neg q \rightarrow p \wedge r) \wedge (r \rightarrow \neg q))}$
- Hemos visto que su FNC es: $\boldsymbol{\xi = ((p) \wedge (q \vee r) \wedge (\neg r \vee \neg q))}$

La Propagación de Literales (III)

Apliquemos la Propagación de Literales a una oración de la que conocemos su FNC

- Si tenemos la oración: **$\alpha = ((\neg p \vee \neg q \rightarrow p \wedge r) \wedge (r \rightarrow \neg q))$**
- Hemos visto que su FNC es: **$\xi = ((p) \wedge (q \vee r) \wedge (\neg r \vee \neg q))$**
  - Si propagamos el literal **$(\ell) = p$**, la nueva FNC $\xi(p)$ resultante es: **$\xi(p)$ = (**$(q \vee r) \wedge (\neg r \vee \neg q)$**)**

La Propagación de Literales (IV)

Apliquemos la Propagación de Literales a una oración de la que conocemos su FNC

- Si tenemos la oración: $\boldsymbol{\alpha = ((\neg p \vee \neg q \to p \wedge r) \wedge (r \to \neg q))}$
- Hemos visto que su FNC es: $\boldsymbol{\xi = ((p) \wedge (q \vee r) \wedge (\neg r \vee \neg q))}$
  - Si propagamos el literal $\boldsymbol{(\ell) = p}$, la nueva FNC $\xi(p)$ resultante es: $\boldsymbol{\xi(p)}$ **=** $\boldsymbol{(}(q \vee r) \wedge (\neg r \vee \neg q)\boldsymbol{)}$
  - Si en esta FNC $\boldsymbol{\xi(p)}$, propagamos el literal $\boldsymbol{(\ell) = q}$, la nueva resultante es: $\boldsymbol{\xi(p, q)}$ **=** $\boldsymbol{(}\neg r\boldsymbol{)}$

La Propagación de Literales (V)

Apliquemos la Propagación de Literales a una oración de la que conocemos su FNC

- Si tenemos la oración: $\boldsymbol{\alpha = ((\neg p \vee \neg q \rightarrow p \wedge r) \wedge (r \rightarrow \neg q))}$
- Hemos visto que su FNC es: $\boldsymbol{\xi = ((p) \wedge (q \vee r) \wedge (\neg r \vee \neg q))}$
  - Si propagamos el literal $\boldsymbol{(\ell) = p}$, la nueva FNC $\xi(p)$ resultante es: $\boldsymbol{\xi(p)}$ $\boldsymbol{=}$ $((q \vee r) \wedge (\neg r \vee \neg q))$
  - Si en esta FNC $\boldsymbol{\xi(p)}$, propagamos el literal $\boldsymbol{(\ell) = q}$, la nueva resultante es**:** $\boldsymbol{\xi(p, q)}$ $\boldsymbol{= (}\neg r\boldsymbol{)}$
  - Si, finalmente, propagamos el literal $\boldsymbol{(\ell) = \neg r}$, la nueva resultante es el conjunto vacío de cláusulas: $\boldsymbol{\xi(p, q, \neg r) = \{\emptyset\}}$

La Propagación de Literales (VI)

Apliquemos la Propagación de Literales a una oración de la que conocemos su FNC

- Si tenemos la oración: $\boldsymbol{\alpha = ((\neg p \lor \neg q \to p \land r) \land (r \to \neg q))}$
- Hemos visto que su FNC es: $\boldsymbol{\xi = ((p) \land (q \lor r) \land (\neg r \lor \neg q))}$
    - Si propagamos el literal **(ℓ) = p**, la nueva FNC ξ(p) resultante es: **ξ(p) = (**$(q \lor r) \land (\neg r \lor \neg q)$**)**
    - Si en esta FNC **ξ(p)**, propagamos el literal **(ℓ) = q**, la nueva resultante es**: ξ(p, q) = (**$\neg r$**)**
    - Si, finalmente, propagamos el literal **(ℓ) = ¬r**, la nueva resultante es el conjunto vacío de cláusulas: **ξ(p, q, ¬r) = {Ø}**

❖ Si en un diferente intento de propagación, desde la FNC original propagamos inicialmente el literal **(ℓ) = ¬q**, la nueva resultante es: **ξ(¬q) = (**$(p) \land (r)$**)**

La Propagación de Literales (VII)

Apliquemos la Propagación de Literales a una oración de la que conocemos su FNC

- Si tenemos la oración: $\boldsymbol{\alpha = ((\neg p \vee \neg q \rightarrow p \wedge r) \wedge (r \rightarrow \neg q))}$
- Hemos visto que su FNC es: $\boldsymbol{\xi = ((p) \wedge (q \vee r) \wedge (\neg r \vee \neg q))}$
    - Si propagamos el literal $\boldsymbol{(\ell) = p}$, la nueva FNC $\xi(p)$ resultante es: $\boldsymbol{\xi(p)} = ((q \vee r) \wedge (\neg r \vee \neg q))$
    - Si en esta FNC $\boldsymbol{\xi(p)}$, propagamos el literal $\boldsymbol{(\ell) = q}$, la nueva resultante es: $\boldsymbol{\xi(p, q)} = (\neg r)$
    - Si, finalmente, propagamos el literal $\boldsymbol{(\ell) = \neg r}$, la nueva resultante es el conjunto vacío de cláusulas: $\boldsymbol{\xi(p, q, \neg r) = \{\emptyset\}}$
    - Si en un diferente intento de propagación, desde la FNC original propagamos inicialmente el literal $\boldsymbol{(\ell) = \neg q}$, la nueva resultante es: $\boldsymbol{\xi(\neg q)} = ((p) \wedge (r))$
    - Si propagamos ahora $\boldsymbol{(\ell) = p}$, nos quedaría $\boldsymbol{\xi(\neg q, p) = (r)}$. Y si, finalmente, propagamos $\boldsymbol{(\ell) = r}$, nos quedaría como antes que la nueva FNC es: $\boldsymbol{\xi(\neg q, p, r) = \{\emptyset\}}$

La Propagación de Literales (y VIII)

Apliquemos la Propagación de Literales a una oración de la que conocemos su FNC

- Si tenemos la oración: $\boldsymbol{\alpha = ((\neg p \vee \neg q \rightarrow p \wedge r) \wedge (r \rightarrow \neg q))}$
- Hemos visto que su FNC es: $\boldsymbol{\xi = ((p) \wedge (q \vee r) \wedge (\neg r \vee \neg q))}$
  - Si propagamos el literal $(\ell) = \mathbf{p}$, la nueva FNC $\xi(p)$ resultante es: $\boldsymbol{\xi(p)} = ((q \vee r) \wedge (\neg r \vee \neg q))$
  - Si en esta FNC $\boldsymbol{\xi(p)}$, propagamos el literal $(\ell) = \mathbf{q}$, la nueva resultante es: $\boldsymbol{\xi(p, q)} = (\neg r)$
  - Si, finalmente, propagamos el literal $(\ell) = \neg \mathbf{r}$, la nueva resultante es el conjunto vacío de cláusulas: $\boldsymbol{\xi(p, q, \neg r) = \{\emptyset\}}$
  - ❖Si en un diferente intento de propagación, desde la FNC original propagamos inicialmente el literal $(\ell) = \neg \mathbf{q}$, la nueva resultante es: $\boldsymbol{\xi(\neg q)} = ((p) \wedge (r))$
  - ❖Si propagamos ahora $(\ell) = \mathbf{p}$, nos quedaría $\boldsymbol{\xi(\neg q, p) = (r)}$. Y si, finalmente, propagamos $(\ell) = \mathbf{r}$, nos quedaría como antes que la nueva FNC es: $\boldsymbol{\xi(\neg q, p, r) = \{\emptyset\}}$
- Por tanto, la oración de partida y su FNC son **SATISFACIBLES** para las interpretaciones:
  - ➢a) **v(p) = V, v(q) = V, v(r) = F**
  - ➢b) **v(p) = V, v(q) = F y v(r) = V**

El Algoritmo DPLL (I)

- Para estudiar la satisfacibilidad de **α** en su FNC = **ξ** (expresada, en su FORMA CLAUSAL) se aplica un algoritmo que consiste en realizar transformaciones sucesivas que manteniendo la satisfacibilidad, sean equivalentes en los términos explicados antes a la fórmula original.
- El método considera cinco reglas:
  - **Regla de la tautología:** Eliminar las cláusulas que contengan un par de literales complementarios.
  - **Regla de la inclusión:** (o de eliminación de cláusulas subsumidas): Si en **ξ** existen conjuntos clausales C1 y C2 tales que C1 es un subconjunto de C2, eliminar C2 de **ξ**. De ahí que, si existe una cláusula C1 cuyos literales están también en otra cláusula C2, eliminar C2 de **ξ**.
  - **Regla de propagación unitaria:** Si existe una cláusula unitaria formada por un único literal (ℓ), propagar el literal (ℓ).
  - **Regla del literal puro:** Si existe un literal puro (ℓ) (para el que no existe su complementario en otras cláusulas), propagar el literal (ℓ).
  - **Regla de ramificación:** Si no existe una cláusula unitaria y no existen literales puros, considerar un literal (ℓ) de alguna cláusula (de forma no determinística) y reducir el problema de satisfacibilidad a resolver uno de estos dos problemas:
    - o bien, la satisfacibilidad de $\xi \cup \{\ell\}$ (propagando (ℓ))
    - o bien, la satisfacibilidad de $\xi \cup \{\neg \ell\}$ (propagando ($\neg \ell$))

El Algoritmo DPLL (y II)

Un ejemplo de aplicación del algoritmo de DPLL

- **ϕ = {{**¬ p**}**; **{**p, q**}**; **{**p, ¬q**}}**
- No podemos aplicar la regla de la Tautología, tampoco la regla de la inclusión. Tenemos una cláusula unitaria cuyo único literal el (¬p); por tanto, propagamos dicho literal:
- **ϕ** (¬p) = **{{**q**}**; **{**¬q**}}**
- Ahora tenemos dos cláusulas unitarias, propagamos cualquiera de los dos literales:
- **ϕ** (¬p, q) = **{ □ }**
- Al resultar la cláusula vacía, se ha FALSEADO la oración, por lo que debemos hacer backtraking y propagar (¬q):
- **ϕ** (¬p, ¬q) = **{ □ }**
- Como en las dos ramas el resultado es la cláusula vacía, esto quiere decir que el conjunto clausulado es **INSATISFACIBLE**, y la FNC es una oración INSATISFACIBLE. Hay que advertir que si volvemos al primer paso y en vez de propagar (¬p), propagásemos su literal complementario (p), entonces la cláusula unitaria de partida se transforma en la cláusula vacía, es decir, asimismo interpretación falseable.

Ejercicio 6: Satisfacibilidad de conjuntos con DPLL (I)

- **Determinar si los siguientes conjuntos de oraciones son satisfacibles utilizando el algoritmo DPLL:**
- **$\{(\neg p \vee q), (\neg p \vee \neg r), (p \vee q), (q \vee r), (\neg q \vee r)\}$**
- **$\{(\neg p), (p \vee s), (q \vee \neg s), (\neg q \vee r), (\neg r)\}$**
- Los dos conjuntos de oraciones son, en realidad, sendos conjuntos de cláusulas, ya que todas ellas son disyunciones de literales. Por tanto, podemos convertir directamente el conjunto en un conjunto clausal y comprobar si es satisfacible con DPLL.

Ejercicio 6: Satisfacibilidad de conjuntos con DPLL (II)

- **Determinar si los siguientes conjuntos de oraciones son satisfacibles utilizando el algoritmo DPLL:**
- **$\{(\neg p \lor q), (\neg p \lor \neg r), (p \lor q), (q \lor r), (\neg q \lor r)\}$**
- **$\{(\neg p), (p \lor s), (q \lor \neg s), (\neg q \lor r), (\neg r)\}$**
- Los dos conjuntos de oraciones son, en realidad, sendos conjuntos de cláusulas, ya que todas ellas son disyunciones de literales. Por tanto, podemos convertir directamente el conjunto en un conjunto clausal y comprobar si es satisfacible con DPLL.
- El primer conjunto en su forma clausal, es: **$\phi = \{\{\neg p, q\}; \{\neg p, \neg r\}; \{p, q\}; \{q, r\}; \{\neg q, r\}\}$**
- No tenemos cláusulas triviales, ni se puede hacer inclusión de cláusulas. No tenemos cláusulas unitarias, ni literales puros. Así que no nos queda más que aplicar la regla de ramificación.

Ejercicio 6: Satisfacibilidad de conjuntos con DPLL (III)

- **Determinar si los siguientes conjuntos de oraciones son satisfacibles utilizando el algoritmo DPLL:**
- **$\{(\neg p \vee q), (\neg p \vee \neg r), (p \vee q), (q \vee r), (\neg q \vee r)\}$**
- **$\{(\neg p), (p \vee s), (q \vee \neg s), (\neg q \vee r), (\neg r)\}$**
- Los dos conjuntos de oraciones son, en realidad, sendos conjuntos de cláusulas, ya que todas ellas son disyunciones de literales. Por tanto, podemos convertir directamente el conjunto en un conjunto clausal y comprobar si es satisfacible con DPLL.
- El primer conjunto en su forma clausal, es: **$\phi = \{\{\neg p, q\}; \{\neg p, \neg r\}; \{p, q\}; \{q, r\}; \{\neg q, r\}\}$**
- No tenemos cláusulas triviales, ni se puede hacer inclusión de cláusulas. No tenemos cláusulas unitarias, ni literales puros. Así que no nos queda más que aplicar la regla de ramificación.
- Propagamos el literal (**$\neg p$**), con ello nos queda **$\phi(\neg p) = \{\{q\}; \{q, r\}; \{\neg q, r\}\}$**

Ejercicio 6: Satisfacibilidad de conjuntos con DPLL (IV)

- **Determinar si los siguientes conjuntos de oraciones son satisfacibles utilizando el algoritmo DPLL:**
- **$\{(\neg p \vee q), (\neg p \vee \neg r), (p \vee q), (q \vee r), (\neg q \vee r)\}$**
- **$\{(\neg p), (p \vee s), (q \vee \neg s), (\neg q \vee r), (\neg r)\}$**
- Los dos conjuntos de oraciones son, en realidad, sendos conjuntos de cláusulas, ya que todas ellas son disyunciones de literales. Por tanto, podemos convertir directamente el conjunto en un conjunto clausal y comprobar si es satisfacible con DPLL.
- El primer conjunto en su forma clausal, es: **$\phi = \{\{\neg p, q\}; \{\neg p, \neg r\}; \{p, q\}; \{q, r\}; \{\neg q, r\}\}$**
- No tenemos cláusulas triviales, ni se puede hacer inclusión de cláusulas. No tenemos cláusulas unitarias, ni literales puros. Así que no nos queda más que aplicar la regla de ramificación.
- Propagamos el literal (**¬p**), con ello nos queda **$\phi(\neg p) = \{\{q\}; \{q, r\}; \{\neg q, r\}\}$**
- Propagamos ahora el literal (**q**), con ello nos queda **$\phi(\neg p, q) = \{\{r\}\}$**

Ejercicio 6: Satisfacibilidad de conjuntos con DPLL (V)

- **Determinar si los siguientes conjuntos de oraciones son satisfacibles utilizando el algoritmo DPLL:**
- **$\{(\neg p \vee q), (\neg p \vee \neg r), (p \vee q), (q \vee r), (\neg q \vee r)\}$**
- **$\{(\neg p), (p \vee s), (q \vee \neg s), (\neg q \vee r), (\neg r)\}$**
- Los dos conjuntos de oraciones son, en realidad, sendos conjuntos de cláusulas, ya que todas ellas son disyunciones de literales. Por tanto, podemos convertir directamente el conjunto en un conjunto clausal y comprobar si es satisfacible con DPLL.
- El primer conjunto en su forma clausal, es: **$\phi = \{\{\neg p, q\}; \{\neg p, \neg r\}; \{p, q\}; \{q, r\}; \{\neg q, r\}\}$**
- No tenemos cláusulas triviales, ni se puede hacer inclusión de cláusulas. No tenemos cláusulas unitarias, ni literales puros. Así que no nos queda más que aplicar la regla de ramificación.
- Propagamos el literal (**$\neg p$**), con ello nos queda **$\phi(\neg p) = \{\{q\}; \{q, r\}; \{\neg q, r\}\}$**
- Propagamos ahora el literal (**q**), con ello nos queda **$\phi(\neg p, q) = \{\{r\}\}$**
- Propagamos, finalmente, el literal (**r**), con lo que nos queda **$\phi(\neg p, q, r) = \{\emptyset\}$**

Ejercicio 6: Satisfacibilidad de conjuntos con DPLL (VI)

- **Determinar si los siguientes conjuntos de oraciones son satisfacibles utilizando el algoritmo DPLL:**
- **$\{(\neg p \vee q), (\neg p \vee \neg r), (p \vee q), (q \vee r), (\neg q \vee r)\}$**
- **$\{(\neg p), (p \vee s), (q \vee \neg s), (\neg q \vee r), (\neg r)\}$**
- Los dos conjuntos de oraciones son, en realidad, sendos conjuntos de cláusulas, ya que todas ellas son disyunciones de literales. Por tanto, podemos convertir directamente el conjunto en un conjunto clausal y comprobar si es satisfacible con DPLL.
- El primer conjunto en su forma clausal, es: **$\phi = \{\{\neg p, q\}; \{\neg p, \neg r\}; \{p, q\}; \{q, r\}; \{\neg q, r\}\}$**
- No tenemos cláusulas triviales, ni se puede hacer inclusión de cláusulas. No tenemos cláusulas unitarias, ni literales puros. Así que no nos queda más que aplicar la regla de ramificación.
- Propagamos el literal (**$\neg p$**), con ello nos queda **$\phi(\neg p) = \{\{q\}; \{q, r\}; \{\neg q, r\}\}$**
- Propagamos ahora el literal (**q**), con ello nos queda **$\phi(\neg p, q) = \{\{r\}\}$**
- Propagamos, finalmente, el literal (**r**), con lo que nos queda **$\phi(\neg p, q, r) = \{\emptyset\}$**
- Es decir, el conjunto vacío. **SATISFACIBLE**. El conjunto dado es SATISFACIBLE para el modelo dado por la interpretación **v(p) = F, v(q) = V, v(r) = V**

Ejercicio 6: Satisfacibilidad de conjuntos con DPLL (VII)

- **Determinar si los siguientes conjuntos de oraciones son satisfacibles utilizando el algoritmo DPLL:**
- **$\{(\neg p \vee q), (\neg p \vee \neg r), (p \vee q), (q \vee r), (\neg q \vee r)\}$**
- **$\{(\neg p), (p \vee s), (q \vee \neg s), (\neg q \vee r), (\neg r)\}$**
- Los dos conjuntos de oraciones son, en realidad, sendos conjuntos de cláusulas, ya que todas ellas son disyunciones de literales. Por tanto, podemos convertir directamente el conjunto en un conjunto clausal y comprobar si es satisfacible con DPLL.
- El primer conjunto en su forma clausal, es: **$\phi = \{\{\neg p, q\}; \{\neg p, \neg r\}; \{p, q\}; \{q, r\}; \{\neg q, r\}\}$**
- No tenemos cláusulas triviales, ni se puede hacer inclusión de cláusulas. No tenemos cláusulas unitarias, ni literales puros. Así que no nos queda más que aplicar la regla de ramificación.
- Propagamos el literal (**$\neg p$**), con ello nos queda **$\phi(\neg p) = \{\{q\}; \{q, r\}; \{\neg q, r\}\}$**
- Propagamos ahora el literal (**q**), con ello nos queda **$\phi(\neg p, q) = \{\{r\}\}$**
- Propagamos, finalmente, el literal (**r**), con lo que nos queda **$\phi(\neg p, q, r) = \{\emptyset\}$**
- Es decir, el conjunto vacío. **SATISFACIBLE**. El conjunto dado es SATISFACIBLE para el modelo dado por la interpretación **$v(p) = F, v(q) = V, v(r) = V$**
- Podríamos tener más modelos?????

Ejercicio 6: Satisfacibilidad de conjuntos con DPLL (VIII)

- **Determinar si los siguientes conjuntos de oraciones son satisfacibles utilizando el algoritmo DPLL:**
- **$\{(\neg p \vee q), (\neg p \vee \neg r), (p \vee q), (q \vee r), (\neg q \vee r)\}$**
- **$\{(\neg p), (p \vee s), (q \vee \neg s), (\neg q \vee r), (\neg r)\}$**
- Los dos conjuntos de oraciones son, en realidad, sendos conjuntos de cláusulas, ya que todas ellas son disyunciones de literales. Por tanto, podemos convertir directamente el conjunto en un conjunto clausal y comprobar si es satisfacible con DPLL.
- El segundo conjunto en su forma clausal, es: **$\phi = \{\{\neg p\}; \{p, s\}; \{q, \neg s\}; \{\neg q, r\}; \{\neg r\}\}$**

Facultad de Informática

Ejercicio 6: Satisfacibilidad de conjuntos con DPLL (IX)

- **Determinar si los siguientes conjuntos de oraciones son satisfacibles utilizando el algoritmo DPLL:**
- **{(¬p ∨ q), (¬p ∨ ¬r), (p ∨ q), (q ∨ r), (¬q ∨ r)}**
- **{(¬p), (p ∨ s), (q ∨ ¬s), (¬q ∨ r), (¬r)}**
- Los dos conjuntos de oraciones son, en realidad, sendos conjuntos de cláusulas, ya que todas ellas son disyunciones de literales. Por tanto, podemos convertir directamente el conjunto en un conjunto clausal y comprobar si es satisfacible con DPLL.
- El segundo conjunto en su forma clausal, es: **ϕ = {{¬p}; {p, s}; {q, ¬s}; {¬q, r}; {¬r}}**
- No tenemos cláusulas triviales, ni se puede hacer inclusión de cláusulas. No tenemos literales puros, pero si tenemos dos cláusulas unitarias **{¬p}** y **{¬r}**.

Ejercicio 6: Satisfacibilidad de conjuntos con DPLL (X)

- **Determinar si los siguientes conjuntos de oraciones son satisfacibles utilizando el algoritmo DPLL:**
- **{(¬p ∨ q), (¬p ∨ ¬r), (p ∨ q), (q ∨ r), (¬q ∨ r)}**
- **{(¬p), (p ∨ s), (q ∨ ¬s), (¬q ∨ r), (¬r)}**
- Los dos conjuntos de oraciones son, en realidad, sendos conjuntos de cláusulas, ya que todas ellas son disyunciones de literales. Por tanto, podemos convertir directamente el conjunto en un conjunto clausal y comprobar si es satisfacible con DPLL.
- El segundo conjunto en su forma clausal, es: **ϕ = {{¬p}; {p, s}; {q, ¬s}; {¬q, r}; {¬r}}**
- No tenemos cláusulas triviales, ni se puede hacer inclusión de cláusulas. No tenemos literales puros, pero si tenemos dos cláusulas unitarias **{¬p}** y **{¬r}**.
- Propagamos, por ejemplo, el literal (**¬r**), con ello nos queda **ϕ(¬r) = {{¬p}; {p, s}; {q, ¬s}; {¬q}}**

Ejercicio 6: Satisfacibilidad de conjuntos con DPLL (XI)

- **Determinar si los siguientes conjuntos de oraciones son satisfacibles utilizando el algoritmo DPLL:**
- **{(¬p ∨ q), (¬p ∨ ¬r), (p ∨ q), (q ∨ r), (¬q ∨ r)}**
- **{(¬p), (p ∨ s), (q ∨ ¬s), (¬q ∨ r), (¬r)}**
- Los dos conjuntos de oraciones son, en realidad, sendos conjuntos de cláusulas, ya que todas ellas son disyunciones de literales. Por tanto, podemos convertir directamente el conjunto en un conjunto clausal y comprobar si es satisfacible con DPLL.
- El segundo conjunto en su forma clausal, es: **ϕ = {{¬p}; {p, s}; {q, ¬s}; {¬q, r}; {¬r}}**
- No tenemos cláusulas triviales, ni se puede hacer inclusión de cláusulas. No tenemos literales puros, pero si tenemos dos cláusulas unitarias **{¬p}** y **{¬r}**.
- Propagamos, por ejemplo, el literal (**¬r**), con ello nos queda **ϕ(¬r) = {{¬p}; {p, s}; {q, ¬s}; {¬q}}**

Ejercicio 6: Satisfacibilidad de conjuntos con DPLL (XII)

- **Determinar si los siguientes conjuntos de oraciones son satisfacibles utilizando el algoritmo DPLL:**
- **$\{(\neg p \vee q), (\neg p \vee \neg r), (p \vee q), (q \vee r), (\neg q \vee r)\}$**
- **$\{(\neg p), (p \vee s), (q \vee \neg s), (\neg q \vee r), (\neg r)\}$**
- Los dos conjuntos de oraciones son, en realidad, sendos conjuntos de cláusulas, ya que todas ellas son disyunciones de literales. Por tanto, podemos convertir directamente el conjunto en un conjunto clausal y comprobar si es satisfacible con DPLL.
- El segundo conjunto en su forma clausal, es: **$\phi = \{\{\neg p\}; \{p, s\}; \{q, \neg s\}; \{\neg q, r\}; \{\neg r\}\}$**
- No tenemos cláusulas triviales, ni se puede hacer inclusión de cláusulas. No tenemos literales puros, pero si tenemos dos cláusulas unitarias **$\{\neg p\}$** y **$\{\neg r\}$**.
- Propagamos, por ejemplo, el literal (**$\neg r$**), con ello nos queda **$\phi(\neg r) = \{\{\neg p\}; \{p, s\}; \{q, \neg s\}; \{\neg q\}\}$**
- Propagamos ahora el literal (**$\neg q$**), con ello nos queda **$\phi(\neg r, \neg q) = \{\{\neg p\}; \{p, s\}; \{\neg s\}\}$**

Ejercicio 6: Satisfacibilidad de conjuntos con DPLL (XIII)

- **Determinar si los siguientes conjuntos de oraciones son satisfacibles utilizando el algoritmo DPLL:**
- **$\{(\neg p \vee q), (\neg p \vee \neg r), (p \vee q), (q \vee r), (\neg q \vee r)\}$**
- **$\{(\neg p), (p \vee s), (q \vee \neg s), (\neg q \vee r), (\neg r)\}$**
- Los dos conjuntos de oraciones son, en realidad, sendos conjuntos de cláusulas, ya que todas ellas son disyunciones de literales. Por tanto, podemos convertir directamente el conjunto en un conjunto clausal y comprobar si es satisfacible con DPLL.
- El segundo conjunto en su forma clausal, es: **$\phi = \{\{\neg p\}; \{p, s\}; \{q, \neg s\}; \{\neg q, r\}; \{\neg r\}\}$**
- No tenemos cláusulas triviales, ni se puede hacer inclusión de cláusulas. No tenemos literales puros, pero si tenemos dos cláusulas unitarias **$\{\neg p\}$** y **$\{\neg r\}$**.
- Propagamos, por ejemplo, el literal (**$\neg r$**), con ello nos queda **$\phi(\neg r) = \{\{\neg p\}; \{p, s\}; \{q, \neg s\}; \{\neg q\}\}$**
- Propagamos ahora el literal (**$\neg q$**), con ello nos queda **$\phi(\neg r, \neg q) = \{\{\neg p\}; \{p, s\}; \{\neg s\}\}$**
- Propagamos el literal (**$\neg p$**), con lo que nos queda **$\phi(\neg r, \neg q, \neg p) = \{\{s\}; \{\neg s\}\}$**

Ejercicio 6: Satisfacibilidad de conjuntos con DPLL (XIV)

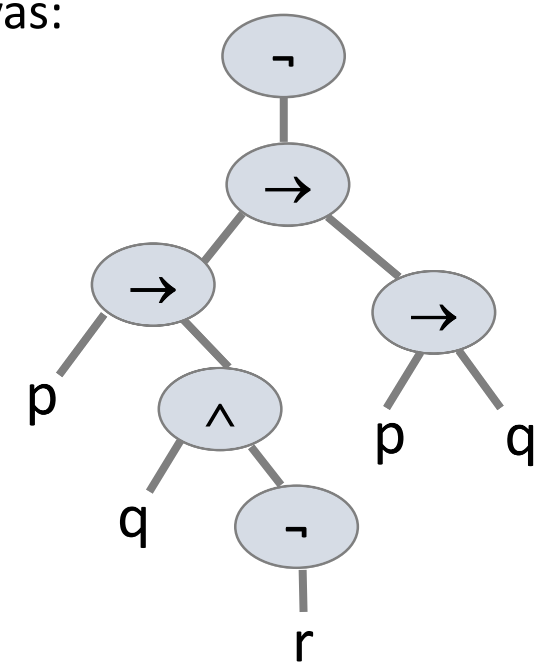
- **Determinar si los siguientes conjuntos de oraciones son satisfacibles utilizando el algoritmo DPLL:**
- **$\{(\neg p \vee q), (\neg p \vee \neg r), (p \vee q), (q \vee r), (\neg q \vee r)\}$**
- **$\{(\neg p), (p \vee s), (q \vee \neg s), (\neg q \vee r), (\neg r)\}$**
- Los dos conjuntos de oraciones son, en realidad, sendos conjuntos de cláusulas, ya que todas ellas son disyunciones de literales. Por tanto, podemos convertir directamente el conjunto en un conjunto clausal y comprobar si es satisfacible con DPLL.
- El segundo conjunto en su forma clausal, es: **$\phi = \{\{\neg p\}; \{p, s\}; \{q, \neg s\}; \{\neg q, r\}; \{\neg r\}\}$**
- No tenemos cláusulas triviales, ni se puede hacer inclusión de cláusulas. No tenemos literales puros, pero si tenemos dos cláusulas unitarias **$\{\neg p\}$** y **$\{\neg r\}$**.
- Propagamos, por ejemplo, el literal (**$\neg r$**), con ello nos queda **$\phi(\neg r) = \{\{\neg p\}; \{p, s\}; \{q, \neg s\}; \{\neg q\}\}$**
- Propagamos ahora el literal (**$\neg q$**), con ello nos queda **$\phi(\neg r, \neg q) = \{\{\neg p\}; \{p, s\}; \{\neg s\}\}$**
- Propagamos el literal (**$\neg p$**), con lo que nos queda **$\phi(\neg r, \neg q, \neg p) =\{\{s\}; \{\neg s\}\}$**
- Finalmente, si propagamos el literal (**s**), o bien, el literal (**$\neg s$**), nos queda unas formas clausales que en ambos caso nos aportan la cláusula vacía. **$\phi(\neg r, \neg q, \neg p, s) = \phi(\neg r, \neg q, \neg p, \neg s) = \{ \square \}$**

Ejercicio 6: Satisfacibilidad de conjuntos con DPLL (y XV)

- **Determinar si los siguientes conjuntos de oraciones son satisfacibles utilizando el algoritmo DPLL:**
- **$\{(\neg p \vee q), (\neg p \vee \neg r), (p \vee q), (q \vee r), (\neg q \vee r)\}$**
- **$\{(\neg p), (p \vee s), (q \vee \neg s), (\neg q \vee r), (\neg r)\}$**
- Los dos conjuntos de oraciones son, en realidad, sendos conjuntos de cláusulas, ya que todas ellas son disyunciones de literales. Por tanto, podemos convertir directamente el conjunto en un conjunto clausal y comprobar si es satisfacible con DPLL.
- El segundo conjunto en su forma clausal, es: **$\phi = \{\{\neg p\}; \{p, s\}; \{q, \neg s\}; \{\neg q, r\}; \{\neg r\}\}$**
- No tenemos cláusulas triviales, ni se puede hacer inclusión de cláusulas. No tenemos literales puros, pero si tenemos dos cláusulas unitarias **$\{\neg p\}$** y **$\{\neg r\}$**.
- Propagamos, por ejemplo, el literal (**$\neg r$**), con ello nos queda **$\phi(\neg r) = \{\{\neg p\}; \{p, s\}; \{q, \neg s\}; \{\neg q\}\}$**
- Propagamos ahora el literal (**$\neg q$**), con ello nos queda **$\phi(\neg r, \neg q) = \{\{\neg p\}; \{p, s\}; \{\neg s\}\}$**
- Propagamos el literal (**$\neg p$**), con lo que nos queda **$\phi(\neg r, \neg q, \neg p) =\{\{s\}; \{\neg s\}\}$**
- Finalmente, si propagamos el literal (**s**)**,** o bien, el literal (**$\neg$s**), nos queda unas formas clausales que en ambos caso nos aportan la cláusula vacía. **$\phi(\neg r, \neg q, \neg p, s) = \phi(\neg r, \neg q, \neg p, \neg s) = \{ \square \}$**
- Las dos ramas resultan obtener la cláusula vacía, y como hemos propagado siempre literales de cláusulas unitarias, no hace falta hacer backtraking. **Así que, el conjunto que nos dan es INSATISFACIBLE**

Ejercicio 7: FNC y satisfacibilidad con DPLL (I)

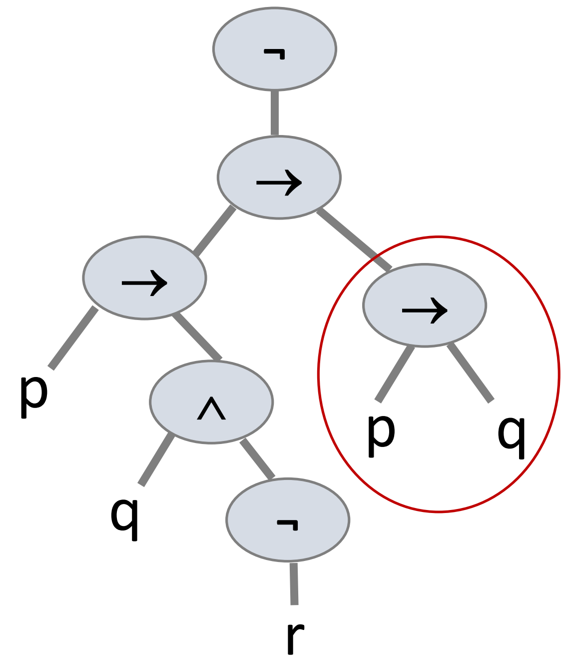
- **Convertir en Forma Normal Conjuntiva (FNC) la siguiente fórmula proposicional. ¿Es satisfacible? Utilizamos el algoritmo DPLL:**
- **$\neg(p \rightarrow (q \wedge \neg r) \rightarrow (p \rightarrow q))$**
- Antes de nada, hagamos el árbol sintáctico de esta formula para no equivocarnos en la precedencia de conectivas:



- Recorramos el árbol de abajo hacia arriba para, aplicando equivalencias lógicas, transformar la fórmula dada en su correspondiente FNC.

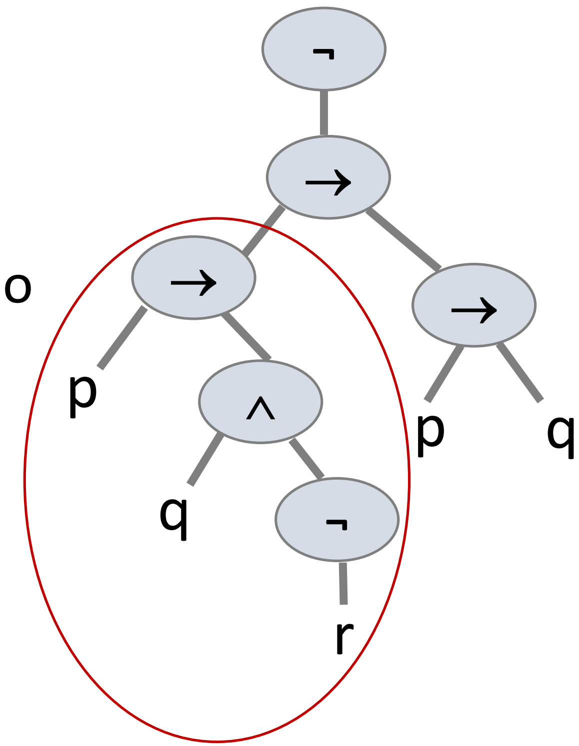
Ejercicio 7: FNC y satisfacibilidad con DPLL (II)

- **Convertir en Forma Normal Conjuntiva (FNC) la siguiente fórmula proposicional. ¿Es satisfacible? Utilizamos el algoritmo DPLL:**
- $\neg(p \rightarrow (q \wedge \neg r) \rightarrow (p \rightarrow q))$
- Aplicamos la interdefinición a la implicación que está en el círculo
- $\neg(p \rightarrow (q \wedge \neg r) \rightarrow (\neg p \vee q))$



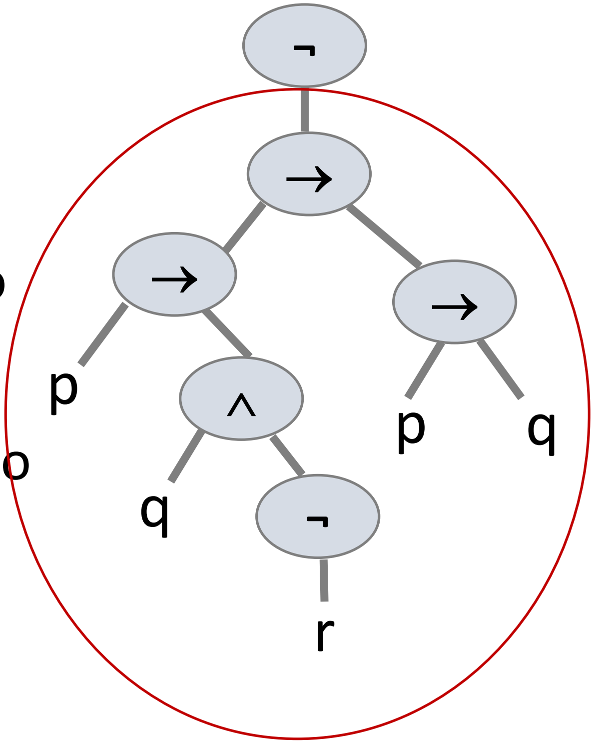
Ejercicio 7: FNC y satisfacibilidad con DPLL (III)

- **Convertir en Forma Normal Conjuntiva (FNC) la siguiente fórmula proposicional. ¿Es satisfacible? Utilizamos el algoritmo DPLL:**
- **$\neg(p \to (q \wedge \neg r) \to (p \to q))$**
- Aplicamos la interdefinición a la implicación que está en el círculo
- **$\neg(p \to (q \wedge \neg r) \to (\neg p \vee q))$**
- Aplicamos de Nuevo la interdefinición a la implicación que está en círculo
- **$\neg(\neg p \vee (q \wedge \neg r) \to (\neg p \vee q))$**



Ejercicio 7: FNC y satisfacibilidad con DPLL (IV)

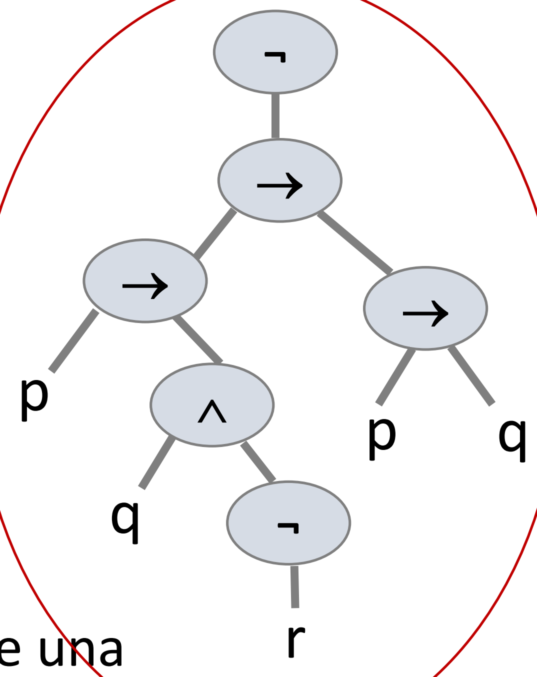
- **Convertir en Forma Normal Conjuntiva (FNC) la siguiente fórmula proposicional. ¿Es satisfacible? Utilizamos el algoritmo DPLL:**
- $\neg(p \to (q \wedge \neg r) \to (p \to q))$
- Aplicamos la interdefinición a la implicación que está en el círculo
- $\neg(p \to (q \wedge \neg r) \to (\neg p \vee q))$
- Aplicamos de nuevo la interdefinición a la implicación que está en círculo
- $\neg(\neg p \vee (q \wedge \neg r) \to (\neg p \vee q))$
- Aplicamos otra vez la interdefinición a la implicación que está en el círculo
- $\neg(\neg(\neg p \vee (q \wedge \neg r)) \vee (\neg p \vee q))$



Facultad de Informática

Ejercicio 7: FNC y satisfacibilidad con DPLL (V)

- **Convertir en Forma Normal Conjuntiva (FNC) la siguiente fórmula proposicional. ¿Es satisfacible? Utilizamos el algoritmo DPLL:**
- $\neg(p \to (q \land \neg r) \to (p \to q))$
- Aplicamos la interdefinición a la implicación que está en el círculo
- $\neg(p \to (q \land \neg r) \to (\neg p \lor q))$
- Aplicamos de nuevo la interdefinición a la implicación que está en círculo
- $\neg(\neg p \lor (q \land \neg r) \to (\neg p \lor q))$
- Aplicamos otra vez la interdefinición a la implicación que está en el círculo
- $\neg(\neg(\neg p \lor (q \land \neg r)) \lor (\neg p \lor q))$
- Aplicamos la equivalencia De Morgan a la negación exterior, la negación de una disyunción
- $(\neg\neg(\neg p \lor (q \land \neg r)) \land \neg(\neg p \lor q))$



Ejercicio 7: FNC y satisfacibilidad con DPLL (VI)

- **Convertir en Forma Normal Conjuntiva (FNC) la siguiente fórmula proposicional. ¿Es satisfacible? Utilizamos el algoritmo DPLL:**
- $\neg(p \rightarrow (q \wedge \neg r) \rightarrow (p \rightarrow q))$
- Aplicamos la idempotencia
- $((\neg p \vee (q \wedge \neg r)) \wedge \neg(\neg p \vee q))$
- Aplicamos en el primer paréntesis la propiedad distributive de la disyunción con respect a la conjunción
- $(((\neg p \vee q) \wedge (\neg p \vee \neg r)) \wedge \neg(\neg p \vee q))$
- Aplicamos la equivalencia De Morgan al ultimo paréntesis, la idempotencia a la doble negación y la asociatividad de la conjunción
- $((\neg p \vee q) \wedge (\neg p \vee \neg r) \wedge p \wedge \neg q)$

Ejercicio 7: FNC y satisfacibilidad con DPLL (VII)

- **Convertir en Forma Normal Conjuntiva (FNC) la siguiente fórmula proposicional. ¿Es satisfacible? Utilizamos el algoritmo DPLL:**
- $\neg(p \rightarrow (q \wedge \neg r) \rightarrow (p \rightarrow q))$
- Aplicamos la idempotencia
- $((\neg p \vee (q \wedge \neg r)) \wedge \neg(\neg p \vee q))$
- Aplicamos en el primer paréntesis la propiedad distributive de la disyunción con respect a la conjunción
- $(((\neg p \vee q) \wedge (\neg p \vee \neg r)) \wedge \neg(\neg p \vee q))$
- Aplicamos la equivalencia De Morgan al ultimo paréntesis, la idempotencia a la doble negación que resulta y la asociatividad de la conjunción
- $((\neg p \vee q) \wedge (\neg p \vee \neg r) \wedge p \wedge \neg q)$
- **Y ESTA ES LA FNC QUE BUSCAMOS!!!! Es una FNC con cuatro cláusulas, dos de ellas unitarias.**

Facultad de Informática

Ejercicio 7: FNC y satisfacibilidad con DPLL (y VIII)

- **Convertir en Forma Normal Conjuntiva (FNC) la siguiente fórmula proposicional. ¿Es satisfacible? Utilizamos el algoritmo DPLL:**
- **$\neg(p \rightarrow (q \wedge \neg r) \rightarrow (p \rightarrow q))$**
- **En su FNC: $((\neg p \vee q) \wedge (\neg p \vee \neg r) \wedge p \wedge \neg q)$**
- Ahora comprobaremos si es una oración **SATISFACIBLE**, utilizando DPLL. Para lo cual pasemos la expression a su Forma Clausal: **$\phi = \{\{\neg p, q\}; \{\neg p, \neg r\}; \{p\}; \{\neg q\}\}$**
- Tenemos dos cláusulas unitarias, por tanto, propagamos el literal de una de ellas
- **$\phi(p) = \{\{q\}; \{\neg r\}; \{\neg q\}\}$**
- Si propagamos el literal de cualquiera de éstas, por ejemplo (q)
- **$\phi(p, q) = \{\{\neg r\}; \{\square\}\}$**
- Si hacemos backtraking y propagamos (¬q), el resultado es: **$\phi(p, \neg q) = \{\{\square\}; \{\neg r\}\}$**
- **Por tanto, obtenemos en las dos ramas la cláusula vacía, con lo que la oración es INSATISFACIBLE. LO COMPRUEBAN USTEDES CON TABLAS DE VERDAD!!!!!**