Transformación de Tseitin

Semana 13

Edgar Andrade, PhD

Última revisión: Abril de 2022

Departmento de Matemáticas Aplicadas y Ciencias de la Computación

Presentación

Forma Normal Conjuntiva: Motivación

1. Queremos construir un algoritmo eficiente para determinar si una fórmula es satisfacible.
2. Los algoritmos eficientes que resuelven este problema trabajan sobre fórmulas en FNC.
3. Debemos mostrar que el rango de estos algoritmos no se reduce a fórmulas en FNC, sino que podemos considerar cualquier fórmula arbitraria.

Forma Normal Conjuntiva

Definiciones:

Una cláusula es una disyunción de literales.

Forma Normal Conjuntiva

Definiciones:

Una cláusula es una disyunción de literales.

Una fórmula está en *Forma Normal Conjuntiva* (FNC) si es una conjunción de cláusulas.

Forma Normal Conjuntiva

Definiciones:

- Una cláusula es una disyunción de literales.
- Una fórmula está en *Forma Normal Conjuntiva* (FNC) si es una conjunción de cláusulas.
- Ejemplo: $(p \vee q) \wedge r$ es una fórmula en FNC.

Forma Normal Conjuntiva

Definiciones:

- Una cláusula es una disyunción de literales.
- Una fórmula está en *Forma Normal Conjuntiva* (FNC) si es una conjunción de cláusulas.
- Ejemplo: $(p \vee q) \wedge r$ es una fórmula en FNC.

Teorema:

Para toda fórmula A, existe una fórmula A' en forma normal conjuntiva tal que $A \equiv A'$.

Forma Clausal

La forma clausal es una variación notacional para las fórmulas en FNC:

Forma Clausal

La forma clausal es una variación notacional para las fórmulas en FNC:

Los literales negativos, digamos $\neg p$, se denotan como $\bar{p}$.

Forma Clausal

La forma clausal es una variación notacional para las fórmulas en FNC:

- Los literales negativos, digamos $\neg p$, se denotan como $\bar{p}$.
- Las cláusulas, digamos $p \vee \neg q \vee r$, se denotan como secuencias de literales, es decir, $p\bar{q}r$.

Forma Clausal

La forma clausal es una variación notacional para las fórmulas en FNC:

- Los literales negativos, digamos $\neg p$, se denotan como $\bar{p}$.
- Las cláusulas, digamos $p \vee \neg q \vee r$, se denotan como secuencias de literales, es decir, $p\bar{q}r$.
- Convención: La cláusula vacía (es decir, la secuencia vacía de literales) se denota por $\square$.

Forma Clausal

La forma clausal es una variación notacional para las fórmulas en FNC:

- Los literales negativos, digamos $\neg p$, se denotan como $\bar{p}$.
- Las cláusulas, digamos $p \vee \neg q \vee r$, se denotan como secuencias de literales, es decir, $p\bar{q}r$.
- Convención: La cláusula vacía (es decir, la secuencia vacía de literales) se denota por $\Box$.
- Una conjunción de cláusulas, digamos $(p \vee q) \wedge (r \vee \neg p)$ se denota como un conjunto de cláusulas, es decir, $\{pq, r\bar{p}\}$.

Forma Clausal

La forma clausal es una variación notacional para las fórmulas en FNC:

- Los literales negativos, digamos $\neg p$, se denotan como $\bar{p}$.
- Las cláusulas, digamos $p \vee \neg q \vee r$, se denotan como secuencias de literales, es decir, $p\bar{q}r$.
- Convención: La cláusula vacía (es decir, la secuencia vacía de literales) se denota por $\square$.
- Una conjunción de cláusulas, digamos $(p \vee q) \wedge (r \vee \neg p)$ se denota como un conjunto de cláusulas, es decir, $\{pq, r\bar{p}\}$.
- Convención: El conjunto vacío de cláusulas se denota por $\emptyset$ y es distinto de $\square$, (pues podemos considerar $\{\square\}$).

Datos importantes

Equivalencias útiles en Forma Normal Conjuntiva:

$$\begin{aligned}
p \leftrightarrow \neg q &\equiv (\neg p \vee \neg q) \wedge (p \vee q) \\
p \leftrightarrow (q \wedge r) &\equiv (q \vee \neg p) \wedge (r \vee \neg p) \wedge (\neg q \vee \neg r \vee p) \\
p \leftrightarrow (q \vee r) &\equiv (\neg q \vee p) \wedge (\neg r \vee p) \wedge (q \vee r \vee \neg p) \\
p \leftrightarrow (q \rightarrow r) &\equiv (q \vee p) \wedge (\neg r \vee p) \wedge (\neg q \vee r \vee \neg p)
\end{aligned}$$

Datos importantes

Equivalencias útiles en Forma Clausal:

$$
\begin{array}{ccc}
p \leftrightarrow \neg q & \equiv & \{\bar{p}\bar{q}, pq\} \\
p \leftrightarrow (q \wedge r) & \equiv & \{q\bar{p}, r\bar{p}, \bar{q}\bar{r}p\} \\
p \leftrightarrow (q \vee r) & \equiv & \{\bar{q}p, \bar{r}p, qr\bar{p}\} \\
p \leftrightarrow (q \rightarrow r) & \equiv & \{qp, \bar{r}p, \bar{q}r\bar{p}\}
\end{array}
$$

Datos importantes

Crecimiento exponencial de la transformación a FNC:

Fórmula Inicial	FNC equivalente	Núm. Inicial de $\wedge$s	Núm. Cláusulas de la FNC
$p \wedge q$	$p \wedge q$	1	2
$(p \wedge q) \vee (r \wedge s)$			

Datos importantes

Crecimiento exponencial de la transformación a FNC:

Fórmula Inicial	FNC equivalente	Núm. Inicial de $\wedge$s	Núm. Cláusulas de la FNC
$p \wedge q$	$p \wedge q$	1	2
$(p \wedge q) \vee (r \wedge s)$	$(p \vee r) \wedge (p \vee s)$ $\wedge (q \vee r) \wedge (q \vee s)$	2	4

Datos importantes

Crecimiento exponencial de la transformación a FNC:

Fórmula Inicial	FNC equivalente	Núm. Inicial de $\wedge$s	Núm. Cláusulas de la FNC
$p \wedge q$	$p \wedge q$	1	2
$(p \wedge q) \vee (r \wedge s)$	$(p \vee r) \wedge (p \vee s)$ $\wedge (q \vee r) \wedge (q \vee s)$	2	4
$(p \wedge q) \vee (r \wedge s) \vee (t \wedge u)$	$(p \vee r \vee t) \wedge (p \vee r \vee u)$ $\wedge (p \vee s \vee t) \wedge (p \vee s \vee u)$ $\wedge (q \vee r \vee t) \wedge (q \vee r \vee u)$ $\wedge (q \vee s \vee t) \wedge (q \vee s \vee u)$	3	8

Datos importantes

Crecimiento exponencial en la transformación a FNC:

Requerimos un procedimiento eficiente tal que, dada una fórmula A, nos proporcione una fórmula A' que esté en FNC y sea equivalente a A.

Datos importantes

Crecimiento exponencial en la transformación a FNC:

Requerimos un procedimiento eficiente tal que, dada una fórmula A, nos proporcione una fórmula A' que esté en FNC y sea equivalente a A.

☞ No tenemos tal procedimiento eficiente (hasta ahora).

Datos importantes

Crecimiento exponencial en la transformación a FNC:

Requerimos un procedimiento eficiente tal que, dada una fórmula A, nos proporcione una fórmula A' que esté en FNC y sea equivalente a A.

☞ No tenemos tal procedimiento eficiente (hasta ahora).

☞ Tenemos una alternativa: la transformación de Tseitin.

Presentación

Idea general

- ☞ Dada una fórmula A, necesitamos una fórmula B en FNC la cual, aunque no sea equivalente a A, sea "igualmente buena" que A.

Idea general

- ☞ Dada una fórmula A, necesitamos una fórmula B en FNC la cual, aunque no sea equivalente a A, sea “igualmente buena” que A.
- ☞ La relación que buscamos es: Si I es modelo de B, entonces I es modelo de A.

Idea general

- ☞ Dada una fórmula A, necesitamos una fórmula B en FNC la cual, aunque no sea equivalente a A, sea "igualmente buena" que A.
- ☞ La relación que buscamos es: Si I es modelo de B, entonces I es modelo de A.
- ☞ Recorderis: Un modelo de una fórmula A es una interpretación I tal que $A.valor(I) = \texttt{True}$.

Idea general

- ☞ Dada una fórmula A, necesitamos una fórmula B en FNC la cual, aunque no sea equivalente a A, sea "igualmente buena" que A.
- ☞ La relación que buscamos es: Si I es modelo de B, entonces I es modelo de A.
- ☞ Recorderis: Un modelo de una fórmula A es una interpretación I tal que $A.valor(I) = \texttt{True}$.
- ☞ Es decir, solucionar el problema B me permite solucionar el problema A.

Idea general

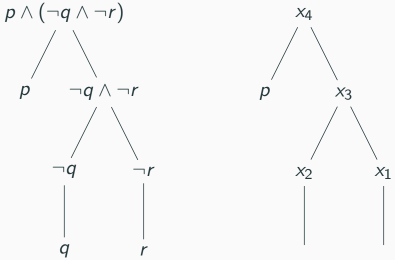
- ☞ Dada una fórmula A, necesitamos una fórmula B en FNC la cual, aunque no sea equivalente a A, sea "igualmente buena" que A.
- ☞ La relación que buscamos es: Si I es modelo de B, entonces I es modelo de A.
- ☞ Recorderis: Un modelo de una fórmula A es una interpretación I tal que $A.valor(I) = \texttt{True}$.
- ☞ Es decir, solucionar el problema B me permite solucionar el problema A.
- ☞ Ejemplo: Todo modelo de $\big(r \to (p \vee q)\big) \wedge (p \leftrightarrow q)$ es modelo de $p \leftrightarrow q$ [Pero no viceversa].

Transformación de Tseitin

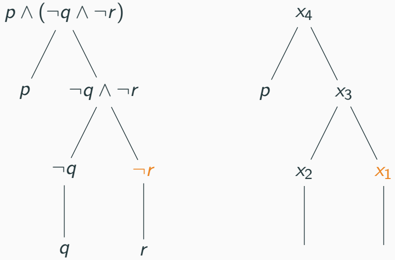
Transformación de Tseitin:

Sea A una fórmula. G.S. Tseitin demostró en 1968 que A puede transformarse eficientemente en una fórmula B en FNC de tal manera que si I es un modelo para B, entonces I es un modelo para A.

Pasos de la transformación —Ejemplo—

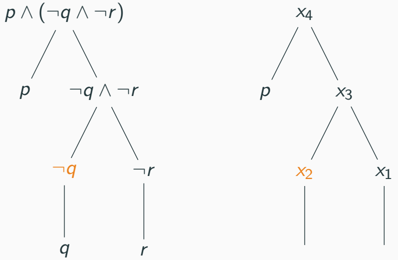


Pasos de la transformación —Ejemplo—



$$x_1 \leftrightarrow \neg r$$

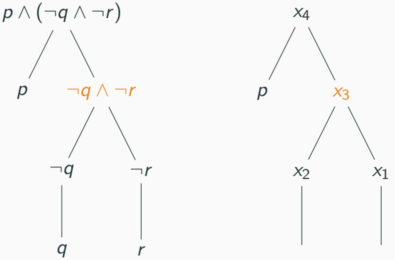
Pasos de la transformación —Ejemplo—



$$x_2 \leftrightarrow \neg q$$

$$x_1 \leftrightarrow \neg r$$

Pasos de la transformación —Ejemplo—

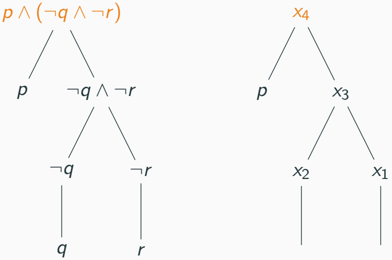


$x_3 \leftrightarrow x_2 \wedge x_1$

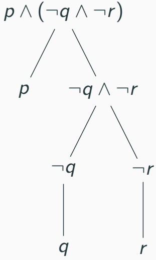
$x_2 \leftrightarrow \neg q$

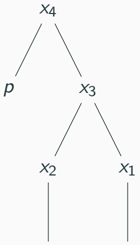
$x_1 \leftrightarrow \neg r$

Pasos de la transformación —Ejemplo—



$$x_4 \leftrightarrow p \wedge x_3$$
$$x_3 \leftrightarrow x_2 \wedge x_1$$
$$x_2 \leftrightarrow \neg q$$
$$x_1 \leftrightarrow \neg r$$





$$x_4 \leftrightarrow p \wedge x_3$$
$$x_3 \leftrightarrow x_2 \wedge x_1$$
$$x_2 \leftrightarrow \neg q$$
$$x_1 \leftrightarrow \neg r$$

La fórmula
$x_4 \wedge (x_4 \leftrightarrow (p \wedge x_3)) \wedge (x_3 \leftrightarrow (x_2 \wedge x_1)) \wedge (x_2 \leftrightarrow \neg q) \wedge (x_1 \leftrightarrow \neg r)$
es tal que si I es uno de sus modelos, entonces I también es un modelo para $p \wedge (\neg q \wedge \neg r)$.

La transformación funciona (1/4)

La fórmula $x_4 \wedge (x_4 \leftrightarrow (p \wedge x_3)) \wedge (x_3 \leftrightarrow (x_2 \wedge x_1)) \wedge (x_2 \leftrightarrow \neg q) \wedge (x_1 \leftrightarrow \neg r)$ es tal que si I es uno de sus modelos, entonces I también es un modelo para $p \wedge (\neg q \wedge \neg r)$. En efecto,

1. Supongamos que I es un modelo para la primera.

La transformación funciona (1/4)

La fórmula $x_4 \wedge (x_4 \leftrightarrow (p \wedge x_3)) \wedge (x_3 \leftrightarrow (x_2 \wedge x_1)) \wedge (x_2 \leftrightarrow \neg q) \wedge (x_1 \leftrightarrow \neg r)$ es tal que si I es uno de sus modelos, entonces I también es un modelo para $p \wedge (\neg q \wedge \neg r)$. En efecto,

1. Supongamos que I es un modelo para la primera.
2. Entonces todos los lados de las conjunciones son verdaderas.

La transformación funciona (1/4)

La fórmula $x_4 \wedge (x_4 \leftrightarrow (p \wedge x_3)) \wedge (x_3 \leftrightarrow (x_2 \wedge x_1)) \wedge (x_2 \leftrightarrow \neg q) \wedge (x_1 \leftrightarrow \neg r)$ es tal que si I es uno de sus modelos, entonces I también es un modelo para $p \wedge (\neg q \wedge \neg r)$. En efecto,

1. Supongamos que I es un modelo para la primera.
2. Entonces todos los lados de las conjunciones son verdaderas.
3. En particular, $x_4.valor(I) = \texttt{True}$ y $(x_4 \leftrightarrow (p \wedge x_3)).valor(I) = \texttt{True}$.

La transformación funciona (2/4)

4. Entonces $(p \wedge x_3).valor(I) = \texttt{True}$, luego $p.valor(I) = \texttt{True}$ y $x_3.valor(I) = \texttt{True}$.

La transformación funciona (2/4)

4. Entonces $(p \wedge x_3).valor(I) = \texttt{True}$, luego $p.valor(I) = \texttt{True}$ y $x_3.valor(I) = \texttt{True}$.
5. Como $(x_3 \leftrightarrow (x_2 \wedge x_1)).valor(I) = \texttt{True}$ y $x_3.valor(I) = \texttt{True}$, entonces $(x_2 \wedge x_1).valor(I) = \texttt{True}$ y luego $x_2.valor(I) = \texttt{True}$ y $x_1.valor(I) = \texttt{True}$.

La transformación funciona (2/4)

4. Entonces $(p \wedge x_3).valor(I) = \texttt{True}$, luego $p.valor(I) = \texttt{True}$ y $x_3.valor(I) = \texttt{True}$.
5. Como $(x_3 \leftrightarrow (x_2 \wedge x_1)).valor(I) = \texttt{True}$ y $x_3.valor(I) = \texttt{True}$, entonces $(x_2 \wedge x_1).valor(I) = \texttt{True}$ y luego $x_2.valor(I) = \texttt{True}$ y $x_1.valor(I) = \texttt{True}$.
6. Por lo tanto $(\neg q).valor(I) = \texttt{True}$ y $(\neg r).valor(I) = \texttt{True}$.

La transformación funciona (2/4)

4. Entonces $(p \wedge x_3).valor(I) = \texttt{True}$, luego $p.valor(I) = \texttt{True}$ y $x_3.valor(I) = \texttt{True}$.
5. Como $(x_3 \leftrightarrow (x_2 \wedge x_1)).valor(I) = \texttt{True}$ y $x_3.valor(I) = \texttt{True}$, entonces $(x_2 \wedge x_1).valor(I) = \texttt{True}$ y luego $x_2.valor(I) = \texttt{True}$ y $x_1.valor(I) = \texttt{True}$.
6. Por lo tanto $(\neg q).valor(I) = \texttt{True}$ y $(\neg r).valor(I) = \texttt{True}$.

La transformación funciona (2/4)

4. Entonces $(p \wedge x_3).valor(I) = \texttt{True}$, luego $p.valor(I) = \texttt{True}$ y $x_3.valor(I) = \texttt{True}$.
5. Como $(x_3 \leftrightarrow (x_2 \wedge x_1)).valor(I) = \texttt{True}$ y $x_3.valor(I) = \texttt{True}$, entonces $(x_2 \wedge x_1).valor(I) = \texttt{True}$ y luego $x_2.valor(I) = \texttt{True}$ y $x_1.valor(I) = \texttt{True}$.
6. Por lo tanto $(\neg q).valor(I) = \texttt{True}$ y $(\neg r).valor(I) = \texttt{True}$.
7. En consecuencia, $(p \wedge (\neg q \wedge \neg r)).valor(I) = \texttt{True}$.

La transformación funciona (3/4)

Por otro lado, observe que si I es una interpretación tal que $I(p) = \texttt{True}$ e $I(q) = I(r) = I(x_4) = \texttt{False}$, entonces I es un modelo para $p \wedge (\neg q \wedge \neg r)$, pero no es un modelo para $x_4 \wedge (x_4 \leftrightarrow (p \wedge x_3)) \wedge (x_3 \leftrightarrow (x_2 \wedge x_1)) \wedge (x_2 \leftrightarrow \neg q) \wedge (x_1 \leftrightarrow \neg r)$.

La transformación funciona (4/4)

Paso final: reemplazar por FNC equivalente:

$$\begin{aligned} x_4 \wedge & (x_4 \leftrightarrow (p \wedge x_3)) \\ & \wedge (x_3 \leftrightarrow (x_2 \wedge x_1)) \\ & \wedge (x_2 \leftrightarrow \neg q) \\ & \wedge (x_1 \leftrightarrow \neg r) \end{aligned}$$

La transformación funciona (4/4)

Paso final: reemplazar por FNC equivalente:

$$
\begin{aligned}
& x_4 \wedge (p \vee \neg x_4) \wedge (x_3 \vee \neg x_4) \wedge (\neg p \vee \neg x_3 \vee x_4) \\
& \quad \wedge (x_3 \leftrightarrow (x_2 \wedge x_1)) \\
& \quad \wedge (x_2 \leftrightarrow \neg q) \\
& \quad \wedge (x_1 \leftrightarrow \neg r)
\end{aligned}
$$

La transformación funciona (4/4)

Paso final: reemplazar por FNC equivalente:

$$
\begin{aligned}
& x_4 \wedge (p \vee \neg x_4) \wedge (x_3 \vee \neg x_4) \wedge (\neg p \vee \neg x_3 \vee x_4) \\
& \quad \wedge (x_2 \vee \neg x_3) \wedge (x_1 \vee \neg x_3) \wedge (\neg x_2 \vee \neg x_1 \vee x_3) \\
& \quad \wedge (x_2 \leftrightarrow \neg q) \\
& \quad \wedge (x_1 \leftrightarrow \neg r)
\end{aligned}
$$

La transformación funciona (4/4)

Paso final: reemplazar por FNC equivalente:

$$
\begin{aligned}
& x_4 \wedge (p \vee \neg x_4) \wedge (x_3 \vee \neg x_4) \wedge (\neg p \vee \neg x_3 \vee x_4) \\
& \quad \wedge (x_2 \vee \neg x_3) \wedge (x_1 \vee \neg x_3) \wedge (\neg x_2 \vee \neg x_1 \vee x_3) \\
& \quad \wedge (\neg x_2 \vee \neg q) \wedge (x_2 \vee q) \\
& \quad \wedge (x_1 \leftrightarrow \neg r)
\end{aligned}
$$

La transformación funciona (4/4)

Paso final: reemplazar por FNC equivalente:

$$
\begin{aligned}
x_4 &\land (p \lor \neg x_4) \land (x_3 \lor \neg x_4) \land (\neg p \lor \neg x_3 \lor x_4) \\
&\land (x_2 \lor \neg x_3) \land (x_1 \lor \neg x_3) \land (\neg x_2 \lor \neg x_1 \lor x_3) \\
&\land (\neg x_2 \lor \neg q) \land (x_2 \lor q) \\
&\land (\neg x_1 \lor \neg r) \land (x_1 \lor r)
\end{aligned}
$$

Algoritmo transformación de Tseitin (1/9)

Suponga que A es una fórmula tal que:

Algoritmo transformación de Tseitin (1/9)

Suponga que A es una fórmula tal que:

☞ no tiene dobles negaciones,

Algoritmo transformación de Tseitin (1/9)

Suponga que A es una fórmula tal que:

- ☞ no tiene dobles negaciones,
- ☞ la consideramos como una cadena de símbolos en notación Inorder,

Algoritmo transformación de Tseitin (1/9)

Suponga que A es una fórmula tal que:

- ☞ no tiene dobles negaciones,
- ☞ la consideramos como una cadena de símbolos en notación Inorder,
- ☞ sea `letras_tseitin`$=[x_1, x_2, \ldots, x_n]$

Algoritmo transformación de Tseitin (1/9)

Suponga que A es una fórmula tal que:

- no tiene dobles negaciones,
- la consideramos como una cadena de símbolos en notación Inorder,
- sea `letras_tseitin`$=[x_1, x_2, \ldots, x_n]$
- suponga que las letras proposicionales de A no tiene elementos en común con la lista `letras_tseitin`,

Algoritmo transformación de Tseitin (1/9)

Suponga que A es una fórmula tal que:

- ☞ no tiene dobles negaciones,
- ☞ la consideramos como una cadena de símbolos en notación Inorder,
- ☞ sea `letras_tseitin`$=[x_1, x_2, \ldots, x_n]$
- ☞ suponga que las letras proposicionales de A no tiene elementos en común con la lista `letras_tseitin`,
- ☞ La unión de ambos conjuntos se llama `Letrasp`

Algoritmo transformación de Tseitin (1/9)

Suponga que A es una fórmula tal que:

- ☞ no tiene dobles negaciones,
- ☞ la consideramos como una cadena de símbolos en notación Inorder,
- ☞ sea `letras_tseitin`$=[x_1, x_2, \ldots, x_n]$
- ☞ suponga que las letras proposicionales de A no tiene elementos en común con la lista `letras_tseitin`,
- ☞ La unión de ambos conjuntos se llama `Letrasp`
- ☞ Sea s el símbolo de la cadena actualmente procesado.

Algoritmo transformación de Tseitin (2/9)

```
L = [] # Inicializamos lista de conjunciones
Pila = [] # Inicializamos pila
i = -1 # Inicializamos contador de variables nuevas
s = A[0] # Inicializamos símbolo de trabajo
Mientras len(A) > 0:
        Si s es un atomo y Pila no vacía y Pila[-1] = '¬':
              i += 1
              atomo = letras_tseitin[i]
              Pila = Pila[:-1]
              Pila.append(atomo)
              L.append(atomo ↔ ¬s)
              A = A[1:]
              si len(A) > 0:
                s = A[0]
        Si no, si s = ')':
             w = Pila[-1]
             u = Pila[-2]
             v = Pila[-3]
             Pila = Pila[:len(Pila)-4]
             i += 1
             atomo = letras_tseitin[i]
             L.append(atomo ↔(vuw))
             s = atomo
```

```
.
.
.
       Si no:
             Pila.append(s)
             A = A[1:]
             si len(A) > 0:
               s = A[0]
Si i < 0:
   atomo = Pila[-1]
Si no:
   atomo = letras_tseitin[i]
B = [[atomo]] + [a_clausal(X) for X in L]
retornar B
```

Algoritmo transformación de Tseitin (3/9)

Ejemplo: Transformar $\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$

☞ Caso 1: $s \in$ `Letrasp` y `Pila[-1]`$=\neg$

Algoritmo transformación de Tseitin (3/9)

Ejemplo: Transformar $\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)$

☞ Caso 1: $s \in$ `Letrasp` y `Pila[-1]`$=\neg$

$$\begin{array}{l}\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big) \\ \quad\;\uparrow\end{array}$$

`s`$=p$

`Pila`$=[(, (, \neg]$

`L`$=[]$

Algoritmo transformación de Tseitin (3/9)

Ejemplo: Transformar $\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$

☞ Caso 1: $s \in$ `Letrasp` y `Pila[-1]`$=\neg$

$$\begin{array}{l}\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)\\ \quad\ \ \uparrow\end{array}$$

`s`$=p$

`Pila`$=[(, (, x_1]$

`L`$=[x_1 \leftrightarrow \neg p]$

Algoritmo transformación de Tseitin (4/9)

Ejemplo: Transformar $\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

Algoritmo transformación de Tseitin (4/9)

Ejemplo: Transformar $\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

$$\begin{array}{l}\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big) \\ \qquad\quad\ \uparrow\end{array}$$

s=)

`Pila`=[(, (, x_1, $\vee$, q]

`L`=[$x_1 \leftrightarrow \neg p$]

Algoritmo transformación de Tseitin (4/9)

Ejemplo: Transformar $\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

$$\begin{array}{l}\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big) \\ \uparrow\end{array}$$

s=)

Pila=[(, (, x_1, $\vee$, p]

L=[$x_1 \leftrightarrow \neg p$]

w=x_1 o=$\vee$ v=q

Algoritmo transformación de Tseitin (4/9)

Ejemplo: Transformar $\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

$$\begin{array}{l}\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big) \\ \uparrow\end{array}$$

s=)

Pila=[(, (, x_1, $\vee$, p]

L=[$x_1 \leftrightarrow \neg p$, $x_2 \leftrightarrow x_1 \vee q$]

w=x_1 o=$\vee$ v=q

Algoritmo transformación de Tseitin (4/9)

Ejemplo: Transformar $\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

$$\begin{array}{l}\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)\\ \qquad\quad\;\uparrow\end{array}$$

s=)

`Pila`=[(, (, x_1, $\vee$, p]

`L`=[$x_1 \leftrightarrow \neg p$, $x_2 \leftrightarrow x_1 \vee q$]

Algoritmo transformación de Tseitin (4/9)

Ejemplo: Transformar $\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

$$\begin{array}{l}\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big) \\ \uparrow\end{array}$$

s=x_2

`Pila`=[(]

`L`=[$x_1 \leftrightarrow \neg p$, $x_2 \leftrightarrow x_1 \vee q$]

Algoritmo transformación de Tseitin (5/9)

Ejemplo: Transformar $\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$

☞ Caso 1: $s \in$ `Letrasp` y `Pila[-1]`$=\neg$

Algoritmo transformación de Tseitin (5/9)

Ejemplo: Transformar $\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$

☞ Caso 1: $s \in$ `Letrasp` y `Pila[-1]`$=\neg$

$$\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$$

$$\uparrow$$

`s`$=t$

`Pila`$=[(, x_2, \to, \neg, (, r, \wedge, \neg]$

`L`$=[x_1 \leftrightarrow \neg p, x_2 \leftrightarrow x_1 \vee q]$

Algoritmo transformación de Tseitin (5/9)

Ejemplo: Transformar $\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)$

☞ Caso 1: $s \in$ `Letrasp` y `Pila[-1]`$=\neg$

$$\begin{array}{l}\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)\\ \hphantom{\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg}\uparrow\end{array}$$

`s`$=t$

`Pila`$=[(, x_2, \rightarrow, \neg, (, r, \wedge, x_3]$

`L`$=[x_1 \leftrightarrow \neg p, x_2 \leftrightarrow x_1 \vee q, x_3 \leftrightarrow \neg t]$

Algoritmo transformación de Tseitin (6/9)

Ejemplo: Transformar $\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

Algoritmo transformación de Tseitin (6/9)

Ejemplo: Transformar $\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

$$\begin{array}{c}\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big) \\ \hphantom{\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t}\uparrow\hphantom{)}\end{array}$$

s=)

`Pila`=[(, x_2, $\rightarrow$, $\neg$, (, r, $\wedge$, x_3]

`L`=[$x_1 \leftrightarrow \neg p$, $x_2 \leftrightarrow x_1 \vee q$, $x_3 \leftrightarrow \neg t$]

Algoritmo transformación de Tseitin (6/9)

Ejemplo: Transformar $\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

$$\begin{array}{l}\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big) \\ \qquad\qquad\qquad\qquad\qquad\quad\ \uparrow\end{array}$$

s=)

Pila=[(, x_2, $\rightarrow$, $\neg$, (, r, $\wedge$, x_3]

L=[$x_1 \leftrightarrow \neg p$, $x_2 \leftrightarrow x_1 \vee q$, $x_3 \leftrightarrow \neg t$]

w=r o=$\wedge$ v=x_3

Algoritmo transformación de Tseitin (6/9)

Ejemplo: Transformar $\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

$$\begin{array}{l}\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)\\ \hphantom{\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t}\uparrow\end{array}$$

s=)

Pila=[(, x_2, $\rightarrow$, $\neg$, (, r, $\wedge$, x_3]

L=[$x_1 \leftrightarrow \neg p$, $x_2 \leftrightarrow x_1 \vee q$, $x_3 \leftrightarrow \neg t$, $x_4 \leftrightarrow r \wedge x_3$]

w=r o=$\wedge$ v=x_3

Algoritmo transformación de Tseitin (6/9)

Ejemplo: Transformar $\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

$$\begin{array}{l}\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big) \\ \uparrow\end{array}$$

s=)

`Pila`=[(, x_2, $\rightarrow$, $\neg$, (, r, $\wedge$, x_3]

`L`=[$x_1 \leftrightarrow \neg p$, $x_2 \leftrightarrow x_1 \vee q$, $x_3 \leftrightarrow \neg t$, $x_4 \leftrightarrow r \wedge x_3$]

Algoritmo transformación de Tseitin (6/9)

Ejemplo: Transformar $\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

$$\begin{array}{c}\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big) \\ \hphantom{\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t}\uparrow\hphantom{)\big)}\end{array}$$

s=x_4

Pila=[(, x_2, $\rightarrow$, $\neg$]

L=[$x_1 \leftrightarrow \neg p$, $x_2 \leftrightarrow x_1 \vee q$, $x_3 \leftrightarrow \neg t$, $x_4 \leftrightarrow r \wedge x_3$]

Ejemplo: Transformar $\big((\neg p \lor q) \to \neg(r \land \neg t)\big)$

☞ Caso 1: $s \in$ `Letrasp` y `Pila[-1]`$=\neg$

Algoritmo transformación de Tseitin (7/9)

Ejemplo: Transformar $\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$

☞ Caso 1: $s \in$ `Letrasp` y `Pila[-1]`$=\neg$

$$\big((\neg p \vee q) \to \neg(r \wedge \neg \underset{\uparrow}{t})\big)$$

s=x_4

`Pila`$=[(,\ x_2,\ \to,\ \neg]$

`L`$=[x_1 \leftrightarrow \neg p,\ x_2 \leftrightarrow x_1 \vee q,\ x_3 \leftrightarrow \neg t,\ x_4 \leftrightarrow r \wedge x_3]$

Algoritmo transformación de Tseitin (7/9)

Ejemplo: Transformar $\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big)$

☞ Caso 1: $s \in$ `Letrasp` y `Pila[-1]`$=\neg$

$$\begin{array}{l}\big((\neg p \vee q) \rightarrow \neg(r \wedge \neg t)\big) \\ \uparrow\end{array}$$

$s=x_4$

`Pila`$=[(, x_2, \rightarrow, x_5]$

`L`$=[x_1 \leftrightarrow \neg p, x_2 \leftrightarrow x_1 \vee q, x_3 \leftrightarrow \neg t, x_4 \leftrightarrow r \wedge x_3, x_5 \leftrightarrow \neg x_4]$

Algoritmo transformación de Tseitin (8/9)

Ejemplo: Transformar $\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

Algoritmo transformación de Tseitin (8/9)

Ejemplo: Transformar $\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

$$\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$$
$$\uparrow$$

s=)

`Pila`=[(, x_2, $\to$, x_5]

`L`=[$x_1 \leftrightarrow \neg p$, $x_2 \leftrightarrow x_1 \vee q$, $x_3 \leftrightarrow \neg t$, $x_4 \leftrightarrow r \wedge x_3$, $x_5 \leftrightarrow \neg x_4$]

Algoritmo transformación de Tseitin (8/9)

Ejemplo: Transformar $\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

$$\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$$

$$\uparrow$$

s=)

`Pila`=[(, x_2, $\to$, x_5]

`L`=[$x_1 \leftrightarrow \neg p$, $x_2 \leftrightarrow x_1 \vee q$, $x_3 \leftrightarrow \neg t$, $x_4 \leftrightarrow r \wedge x_3$, $x_5 \leftrightarrow \neg x_4$]

`w`=x_2 `o`=$\to$ `v`=x_5

Algoritmo transformación de Tseitin (8/9)

Ejemplo: Transformar $\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

$$\begin{array}{l}\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big) \\ \hspace{16.5em}\uparrow\end{array}$$

s=)

`Pila`=[(, x_2, $\to$, x_5]

`L`=[$x_1 \leftrightarrow \neg p$, $x_2 \leftrightarrow x_1 \vee q$, $x_3 \leftrightarrow \neg t$, $x_4 \leftrightarrow r \wedge x_3$, $x_5 \leftrightarrow \neg x_4$,
$x_6 \leftrightarrow x_2 \to x_5$]

`w`=x_2 `o`=$\to$ `v`=x_5

Algoritmo transformación de Tseitin (8/9)

Ejemplo: Transformar $\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$

☞ Caso 2: $s{=})$

$$\begin{array}{l}\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)\\ \hphantom{\big((\neg p \vee q) \to \neg(r \wedge \neg t)}\uparrow\end{array}$$

$s{=})$

Pila$=[($, x_2, $\to$, $x_5]$

L$=[x_1 \leftrightarrow \neg p$, $x_2 \leftrightarrow x_1 \vee q$, $x_3 \leftrightarrow \neg t$, $x_4 \leftrightarrow r \wedge x_3$, $x_5 \leftrightarrow \neg x_4$, $x_6 \leftrightarrow x_2 \to x_5]$

Algoritmo transformación de Tseitin (8/9)

Ejemplo: Transformar $\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$

☞ Caso 2: s=)

$$\big((\neg p \vee q) \to \neg(r \wedge \neg t)\big)$$
$$\uparrow$$

s=)

`Pila`=[]

`L`=[$x_1 \leftrightarrow \neg p$, $x_2 \leftrightarrow x_1 \vee q$, $x_3 \leftrightarrow \neg t$, $x_4 \leftrightarrow r \wedge x_3$, $x_5 \leftrightarrow \neg x_4$, $x_6 \leftrightarrow x_2 \to x_5$]

☞ len(A)=0

Algoritmo transformación de Tseitin (9/9)

```
L = [] # Inicializamos lista de conjunciones
Pila = [] # Inicializamos pila
i = -1 # Inicializamos contador de variables nuevas
s = A[0] # Inicializamos símbolo de trabajo
Mientras len(A) > 0:
    Si s es un atomo y Pila no vacía y Pila[-1] = '¬':
        i += 1
        atomo = letras_tseitin[i]
        Pila = Pila[:-1]
        Pila.append(atomo)
        L.append(atomo ↔ ¬s)
        A = A[1:]
        si len(A) > 0:
          s = A[0]
    Si no, si s = ')':
        w = Pila[-1]
        u = Pila[-2]
        v = Pila[-3]
        Pila = Pila[:len(Pila)-4]
        i += 1
        atomo = letras_tseitin[i]
        L.append(atomo ↔(vuw))
        s = atomo
```

```
⋮
    Si no:
        Pila.append(s)
        A = A[1:]
        si len(A) > 0:
          s = A[0]
Si i < 0:
   atomo = Pila[-1]
Si no:
   atomo = letras_tseitin[i]
B = [[atomo]] + [a_clausal(X) for X in L]
retornar B
```

Datos importantes

Comparación transformación a FNC y transformación de Tseitin:

Fórmula Inicial	Núm. Inicial de $\wedge$s	Núm. Cláusulas de la FNC	Núm. Cláusulas Tseitin
$p \wedge q$	1	2	4
$(p \wedge q) \vee (r \wedge s)$	2	4	10
$(p \wedge q) \vee (r \wedge s) \vee (t \wedge u)$	3	8	16
$(p \wedge q) \vee (r \wedge s) \vee (t \wedge u) \vee (a \wedge b)$	4	16	22
$(p \wedge q) \vee (r \wedge s) \vee (t \wedge u) \vee (a \wedge b) \vee (c \wedge d)$	5	32	28
$(p \wedge q) \vee (r \wedge s) \vee (t \wedge u) \vee (a \wedge b) \vee (c \wedge d) \vee (e \wedge f)$	6	64	34

Tareas faltantes...

☞ Correr el algoritmo con algunas fórmulas sencillas.

☞ Verificar que el algoritmo devuelve una fórmula B tal que:

- B está en FNC.
- si encontramos un modelo para B, entonces encontramos un modelo para A.
- la longitud de B está en proporción aritmética respecto a la longitud de A.

☞ Verificar que el algoritmo es eficiente (es decir, pertenece a $O(N)$).

Fin de la sesión 25

En esta sesión usted ha aprendido a:

1. Dada una fórmula arbitraria A, encontrar de manera eficiente una fórmula B en forma clausal que sea 'igualmente buena'.
2. Una fórmula B es 'igualmente buena' para A si todo modelo de B es un modelo de A.
3. El procedimiento de la transformación de Tseitin.