Lógica informática (2005–06)

Tema 3: Formas normales

José A. Alonso Jiménez

Grupo de Lógica Computacional
Dpto. Ciencias de la Computación e Inteligencia Artificial
Universidad de Sevilla

Equivalencia lógica: Fórmulas equivalentes.

- Def.: F y G son equivalentes si $v(F) = v(G)$ para toda valoración v.
  Representación: $F \equiv G$.
- Ejemplos de equivalencias notables:
  1. Idempotencia: $F \vee F \equiv F$; $F \wedge F \equiv F$.
  2. Conmutatividad: $F \vee G \equiv G \vee F$; $F \wedge G \equiv G \wedge F$.
  3. Asociatividad: $F \vee (G \vee H) \equiv (F \vee G) \vee H$; $F \wedge (G \wedge H) \equiv (F \wedge G) \wedge H$.
  4. Absorción: $F \wedge (F \vee G) \equiv F$; $F \vee (F \wedge G) \equiv F$.
  5. Distributividad: $F \wedge (G \vee H) \equiv (F \wedge G) \vee (F \wedge H)$;
     $F \vee (G \wedge H) \equiv (F \vee G) \wedge (F \vee H)$.
  6. Doble negación: $\neg\neg F \equiv F$.
  7. Leyes de De Morgan: $\neg(F \wedge G) \equiv \neg F \vee \neg G$; $\neg(F \vee G) \equiv \neg F \wedge \neg G$.
  8. Leyes de tautologías: Si F es una tautología, $F \wedge G \equiv G$; $F \vee G \equiv F$.
  9. Leyes de contradicciones: Si F es una contradicción $F \wedge G \equiv F$; $F \vee G \equiv G$.

Equivalencia lógica: propiedades

- Relación entre equivalencia y bicondicional:
  - $F \equiv G$ syss $\models F \leftrightarrow G$ syss $\vdash F \leftrightarrow G$.
- Propiedades básicas de la equivalencia lógica:
  - Reflexiva: $F \equiv F$.
  - Simétrica: Si $F \equiv G$, entonces $G \equiv F$.
  - Transitiva: Si $F \equiv G$ y $G \equiv H$, entonces $F \equiv H$.
- Principio de sustitución de fórmulas equivalentes:
  - Prop.: Si en la fórmula F se sustituye una de sus subfórmulas G por una fórmula G' lógicamente equivalente a G, entonces la fórmula obtenida, F', es lógicamente equivalente a F.
  - Ejemplo:
$$\begin{aligned} F &= \neg(p \wedge q) \rightarrow \neg(p \wedge \neg\neg r) \\ G &= \neg(p \wedge q) \\ G' &= \neg p \vee \neg q \\ F' &= (\neg p \vee \neg q) \rightarrow \neg(p \wedge \neg\neg r) \end{aligned}$$

Formas normales: Forma normal conjuntiva

- Átomos y literales:
  - Def.: Un átomo es un variable proposicional (p.e. $p, q, \dots$).
  - Def.: Un literal es un átomo o su negación (p.e. $p, \neg p, q, \neg q, \dots$).
  - Notación: $L, L_1, L_2, \dots$ representarán literales.
- Forma normal conjuntiva:
  - Def.: Una fórmula está en forma normal conjuntiva (FNC) si es una conjunción de disyunciones de literales; es decir, es de la forma $(L_{1,1} \vee \dots \vee L_{1,n_1}) \wedge \dots \wedge (L_{m,1} \vee \dots \vee L_{m,n_m})$.
  - Ejemplos: $(\neg p \vee q) \wedge (\neg q \vee p)$ está en FNC.
    $(\neg p \vee q) \wedge (q \rightarrow p)$ no está en FNC.
  - Def.: Una fórmula G es una forma normal conjuntiva (FNC) de la fórmula F si G está en forma normal conjuntiva y es equivalente a F.
  - Ejemplo: Una FNC de $\neg(p \wedge (q \rightarrow r))$ es $(\neg p \vee q) \wedge (\neg p \vee \neg r)$.

Formas normales: Cálculo de forma normal conjuntiva

- Algoritmo de cálculo de forma normal conjuntiva:
  - Algoritmo: Aplicando a una fórmula F los siguientes pasos se obtiene una forma normal conjuntiva de F, FNC(F):
    1. Eliminar los bicondicionales usando la equivalencia
    $$A \leftrightarrow B \equiv (A \rightarrow B) \wedge (B \rightarrow A) \quad (1)$$
    2. Eliminar los condicionales usando la equivalencia
    $$A \rightarrow B \equiv \neg A \vee B \quad (2)$$
    3. Interiorizar las negaciones usando las equivalencias
    $$\neg(A \wedge B) \equiv \neg A \vee \neg B \quad (3)$$
    $$\neg(A \vee B) \equiv \neg A \wedge \neg B \quad (4)$$
    $$\neg\neg A \equiv A \quad (5)$$
    4. Interiorizar las disyunciones usando las equivalencias
    $$A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C) \quad (6)$$
    $$(A \wedge B) \vee C \equiv (A \vee C) \wedge (B \vee C) \quad (7)$$

Formas normales: Cálculo de forma normal conjuntiva

- Ejemplo de cálculo de una FNC de $\neg(p \wedge (q \rightarrow r))$:

$$
\begin{array}{lll}
 & \neg(p \wedge (q \rightarrow r)) & \\
\equiv & \neg(p \wedge (\neg q \vee r)) & \text{[por (2)]} \\
\equiv & \neg p \vee \neg(\neg q \vee r) & \text{[por (3)]} \\
\equiv & \neg p \vee (\neg\neg q \wedge \neg r) & \text{[por (4)]} \\
\equiv & \neg p \vee (q \wedge \neg r) & \text{[por (5)]} \\
\equiv & (\neg p \vee q) \wedge (\neg p \vee \neg r) & \text{[por (6)]}
\end{array}
$$

- Ejemplo de cálculo de una FNC de $(p \rightarrow q) \vee (q \rightarrow p)$:

$$
\begin{array}{lll}
 & (p \rightarrow q) \vee (q \rightarrow p) & \\
\equiv & (\neg p \vee q) \vee (\neg q \vee p) & \text{[por (2)]} \\
\equiv & \neg p \vee q \vee \neg q \vee p &
\end{array}
$$

Formas normales: Cálculo de forma normal conjuntiva

- Ejemplo de cálculo de una FNC de $(p \leftrightarrow q) \rightarrow r$:

$$
\begin{array}{rll}
 & (p \leftrightarrow q) \rightarrow r & \\
\equiv & (p \rightarrow q) \wedge (q \rightarrow p) \rightarrow r & [(1)] \\
\equiv & \neg((p \rightarrow q) \wedge (q \rightarrow p)) \vee r & [(2)] \\
\equiv & \neg((\neg p \vee q) \wedge (\neg q \vee p)) \vee r & [(2)] \\
\equiv & (\neg(\neg p \vee q) \vee \neg(\neg q \vee p)) \vee r & [(3)] \\
\equiv & ((\neg\neg p \wedge \neg q) \vee (\neg\neg q \wedge \neg p)) \vee r & [(4)] \\
\equiv & ((p \wedge \neg q) \vee (q \wedge \neg p)) \vee r & [(5)] \\
\equiv & (((p \wedge \neg q) \vee q) \wedge ((p \wedge \neg q) \vee \neg p)) \vee r & [(6)] \\
\equiv & (((p \vee q) \wedge (\neg q \vee q)) \wedge ((p \vee \neg p) \wedge (\neg q \vee \neg p))) \vee r & [(7)] \\
\equiv & (((p \vee q) \wedge (\neg q \vee q)) \vee r) \wedge (((p \vee \neg p) \wedge (\neg q \vee \neg p)) \vee r) & [(7)] \\
\equiv & (((p \vee q) \vee r) \wedge ((\neg q \vee q) \vee r)) \wedge (((p \vee \neg p) \vee r) \wedge ((\neg q \vee \neg p) \vee r)) & [(7)] \\
\equiv & (p \vee q \vee r) \wedge (\neg q \vee q \vee r) \wedge (p \vee \neg p \vee r) \wedge (\neg q \vee \neg p \vee r) & \\
\equiv & (p \vee q \vee r) \wedge (\neg q \vee \neg p \vee r) &
\end{array}
$$

Formas normales: Forma normal disyuntiva

- Forma normal disyuntiva:
  - Def.: Una fórmula está en forma normal disyuntiva (FND) si es una disyunción de conjunciones de literales; es decir, es de la forma
    $(L_{1,1} \wedge \ldots \wedge L_{1,n_1}) \vee \ldots \vee (L_{m,1} \wedge \ldots \wedge L_{m,n_m})$.
  - Ejemplos: $(\neg p \wedge q) \vee (\neg q \wedge p)$ está en FND.
    $(\neg p \wedge q) \vee (q \to p)$ no está en FND.
  - Def.: Una fórmula G es una forma normal disyuntiva (FND) de la fórmula F si G está en forma normal disyuntiva y es equivalente a F.
  - Ejemplo: Una FND de $\neg(p \wedge (q \to r))$ es $\neg p \vee (q \wedge \neg r)$.

Formas normales: Cálculo de forma normal disyuntiva

- Algoritmo de cálculo de forma normal disyuntiva:
  - Algoritmo: Aplicando a una fórmula F los siguientes pasos se obtiene una forma normal disyuntiva de F, FND(F):
    1. Eliminar los bicondicionales usando la equivalencia
       $$A \leftrightarrow B \equiv (A \to B) \land (B \to A) \tag{1}$$
    2. Eliminar los condicionales usando la equivalencia
       $$A \to B \equiv \neg A \lor B \tag{2}$$
    3. Interiorizar las negaciones usando las equivalencias
       $$\neg(A \land B) \equiv \neg A \lor \neg B \tag{3}$$
       $$\neg(A \lor B) \equiv \neg A \land \neg B \tag{4}$$
       $$\neg\neg A \equiv A \tag{5}$$
    4. Interiorizar las conjunciones usando las equivalencias
       $$A \land (B \lor C) \equiv (A \land B) \lor (A \land C) \tag{6}$$
       $$(A \lor B) \land C \equiv (A \land C) \lor (B \land C) \tag{7}$$

Formas normales: Cálculo de forma normal disyuntiva

- Ejemplo de cálculo de una FND de $\neg(p \wedge (q \rightarrow r))$:

$$
\begin{array}{lll}
 & \neg(p \wedge (q \rightarrow r)) & \\
\equiv & \neg(p \wedge (\neg q \vee r)) & \text{[por (2)]} \\
\equiv & \neg p \vee \neg(\neg q \vee r) & \text{[por (3)]} \\
\equiv & \neg p \vee (\neg\neg q \wedge \neg r) & \text{[por (4)]} \\
\equiv & \neg p \vee (q \wedge \neg r) & \text{[por (5)]}
\end{array}
$$

- Ejemplo de cálculo de una FND de $\neg(\neg p \vee \neg q \rightarrow \neg(p \wedge q))$:

$$
\begin{array}{lll}
 & \neg(\neg p \vee \neg q \rightarrow \neg(p \wedge q)) & \\
\equiv & \neg(\neg(\neg p \vee \neg q) \vee \neg(p \wedge q)) & \text{[por (2)]} \\
\equiv & \neg\neg(\neg p \vee \neg q) \wedge \neg\neg(p \wedge q) & \text{[por (4)]} \\
\equiv & (\neg p \vee \neg q) \wedge (p \wedge q) & \text{[por (5)]} \\
\equiv & (\neg p \wedge (p \wedge q)) \vee (\neg q \wedge (p \wedge q)) & \text{[por (7)]} \\
\equiv & (\neg p \wedge p \wedge q) \vee (\neg q \wedge p \wedge q) &
\end{array}
$$

Decisión de validez mediante FNC

- Literales complementarios:
  - El complementario de un literal L es $L^c = \begin{cases} \neg p & \text{si } L = p; \\ p & \text{si } L = \neg p. \end{cases}$
- Propiedades de reducción de tautologías:
  - $F_1 \wedge \ldots \wedge F_n$ es una tautología syss $F_1, \ldots, F_n$ lo son.
  - $L_1 \vee \ldots \vee L_n$ es una tautología syss $\{L_1, \ldots, L_n\}$ contiene algún par de literales complementarios (i.e. existen i, j tales que $L_i = L_j^c$).
- Algoritmo de decisión de tautologías mediante FNC
  - Entrada: Una fórmula F.
  - Procedimiento:
    1. Calcular una FNC de F.
    2. Decidir si cada una de las conjunciones de la FNC tiene algún par de literales complementarios.

Decisión de validez mediante FNC

- Ejemplos de decisión de tautologías mediante FNC
  - $\neg(p \land (q \to r))$ no es tautología:
    $\text{FNC}(\neg(p \land (q \to r))) = (\neg p \lor q) \land (\neg p \lor \neg r)$
    Contramodelos de $\neg(p \land (q \to r))$:
    v_1 tal que $v_1(p) = 1$ y $v_1(q) = 0$
    v_2 tal que $v_2(p) = 1$ y $v_2(r) = 1$
  - $(p \to q) \lor (q \to p)$ es tautología:
    $\text{FNC}((p \to q) \lor (q \to p)) = \neg p \lor q \lor \neg q \lor p$
  - $(p \leftrightarrow q) \to r$ no es tautología:
    $\text{FNC}((p \leftrightarrow q) \to r) = (p \lor q \lor r) \land (\neg q \lor \neg p \lor r)$
    Contramodelos de $(p \leftrightarrow q) \to r$:
    v_1 tal que $v_1(p) = 0, v_1(q) = 0$ y $v_1(r) = 0$
    v_2 tal que $v_2(p) = 1, v_2(q) = 1$ y $v_2(r) = 0$

Decisión de satisfacibilidad mediante FND

- Propiedades de reducción de satisfacibilidad:
  - $F_1 \vee \ldots \vee F_n$ es satisfacible syss alguna de las fórmulas $F_1, \ldots, F_n$ lo es.
  - $L_1 \wedge \ldots \wedge L_n$ es satisfacible syss $\{L_1, \ldots, L_n\}$ no contiene ningún par de literales complementarios.
- Algoritmo de decisión de satisfacibilidad mediante FND:
  - Entrada: Una fórmula F.
  - Procedimiento:
    1. Calcular una FND de F.
    2. Decidir si alguna de las disyunciones de la FND no tiene un par de literales complementarios.

Decisión de satisfacibilidad mediante FND

- Ejemplos de decisión de satisfacibilidad mediante FND:
  - $\neg(p \land (q \to r))$ es satisfacible:

    $\text{FND}(\neg(p \land (q \to r))) = \neg p \lor (q \land \neg r)$

    Modelos de $\neg(p \land (q \to r))$:

    v_1 tal que $v_1(p) = 0$

    v_2 tal que $v_2(q) = 1$ y $v_2(r) = 0$
  - $\neg(\neg p \lor \neg q \to \neg(p \land q))$ es insatisfacible:

    $\text{FND}(\neg(\neg p \lor \neg q \to \neg(p \land q))) = (\neg p \land p \land q) \lor (\neg q \land p \land q)$

Bibliografía

1. C. Badesa, I. Jané y R. Jansana *Elementos de lógica formal.* (Ariel, 2000)

   Cap. 8 (Equivalencia lógica) y 10 (Formas normales).

2. M. Ben–Ari, *Mathematical logic for computer science (2nd ed.).* (Springer, 2001)

   Cap. 2 (Propositional calculus: formulas, models, tableaux).

3. J.A. Díez *Iniciación a la Lógica*, (Ariel, 2002)

   Cap. 3 (Semántica formal. Consecuencia lógica).

4. M. Huth y M. Ryan *Logic in computer science: modelling and reasoning about systems.* (Cambridge University Press, 2000)

   Cap. 1 (Propositional logic).

5. E. Paniagua, J.L. Sánchez y F. Martín *Lógica computacional* (Thomson, 2003)

   Cap. 4.4 (Formas normales).