Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

IIC2213 — Lógica para ciencia de la computación — 1' 2023

TAREA 7

Publicación: Martes 20 de junio.
Entrega: **Viernes 30 de junio hasta las 23:59 horas**.

Indicaciones

- Cada pregunta tiene 6 puntos (+1 base) y la nota de la tarea es el promedio de las preguntas.
- La solución debe estar escrita en LaTeX. No se aceptarán tareas escritas de otra forma.
- La tarea es individual, pudiendo discutirla con sus pares. Toda referencia externa debe citarse.

Objetivos

- Aplicar el concepto de consecuencia lógica en lógica proposicional.
- Modelar problemas usando lógica proposicional.
- Aplicar el concepto de NP-completitud a problemas generales.
- Construir y demostrar correctitud de reducciones.

Pregunta 1: Lógica proposicional

(a) Sea P un conjunto de variables proposicionales. Dado $\Sigma \subseteq \mathcal{L}(P)$ y fórmulas $\alpha, \beta \in \mathcal{L}(P)$ demuestre que

$$\Sigma \models (\alpha \to \beta) \quad \text{si y solo si} \quad \Sigma \cup \{\alpha\} \models \beta$$

(b) Sea $\mathcal{L} = \{R\}$ vocabulario con R símbolo de relación n-aria y sea $\mathfrak{A}$ una $\mathcal{L}$-estructura con dominio finito. Construya una fórmula proposicional φ tal que φ sea satisfacible si y solo si existe un automorfismo no trivial en $\mathfrak{A}$. Demuestre la correctitud de su construcción. *Aclaración:* un automorfismo no trivial es un isomorfismo de $\mathfrak{A}$ en $\mathfrak{A}$, tal que es distinto de la función $f(a) = a$.

Solución P1.

Parte (a) - 3 pts.

($\Rightarrow$) Supongamos que tenemos $\Sigma \models (\alpha \to \beta)$, entonces para toda valuación tal que $\sigma(\Sigma) = 1$, tenemos que $\sigma(\alpha \to \beta) = 1$, por la definición de implicancia, sabemos que eso significa que cuando $\sigma(\alpha) = 1$, entonces $\sigma(\beta) = 1$. Ahora notemos que si $\sigma(\Sigma \cup \{\alpha\}) = 1$ entonces $\sigma(\Sigma) = 1$ y $\sigma(\alpha) = 1$, entonces $\sigma(\beta) = 1$. Es decir, $\Sigma \cup \{\alpha\} \models \beta$.

($\Leftarrow$) Supongamos que $\Sigma \cup \{\alpha\} \models \beta$. Sea σ una valuación tal que $\sigma(\Sigma) = 1$, queremos probar que $\sigma(\alpha \to \beta) = 1$. Notemos que tenemos dos casos:

1. $\sigma(\alpha) = 0$: En este caso por definición de implicancia, tenemos que $\sigma(\alpha \to \beta) = 1$.

2. $\sigma(\alpha) = 1$: En este caso, tenemos que $\sigma(\Sigma \cup \{\alpha\}) = 1$, entonces opr definición de consecuencia lógica tenemos que $\sigma(\beta) = 1$, por lo tanto, $\sigma(\alpha \to \beta) = 1$.

En conclusión $\sigma(\Sigma) = 1$ implica que $\sigma(\alpha \to \beta) = 1$, entonces $\Sigma \models (\alpha \to \beta)$.

Otorgar la mitad del puntaje por cada implicancia.

Parte (b) - 3 pts.

Supongamos que tenemos la estructura $\mathfrak{A} = \langle D, R \rangle$, queremos encontrar una función entre D y D que preserva aristas, por lo tanto definiremos las variables proposicionales

$$p_{ij} := \text{el elemento } i \in D \text{ se mapea a } j \in D$$

Con esto, $P = \{p_{ij} : i, j \in D\}$.

Ahora construiremos fórmulas para imponer las restricciones

- $\varphi_{\text{función}}$: La función está bien definida, es decir todo elemento tiene exactamente una imagen.
- $\varphi_{\text{biyección}}$: La función es una biyección.
- φ_R: La función preserva la relación
- φ_{triv}: La función no es trivial

Primero, definimos $\varphi_{\text{función}} := \varphi_{\text{imagen}} \wedge \varphi_{\text{unicidad}}$ donde

$$\varphi_{\text{imagen}} = \bigwedge_{i \in D} \bigvee_{j \in D} p_{ij}$$

$$\varphi_{\text{unicidad}} = \bigwedge_{i \in D} \bigwedge_{j \in D} \left(p_{ij} \to \bigwedge_{\substack{k \in D \\ k \neq j}} \neg p_{ik} \right)$$

La fórmula φ_{imagen} asegura que cada elemento tenga una imagen y $\varphi_{\text{unicidad}}$ asegura que esta imagen es única.

Luego, definimos $\varphi_{\text{biyección}} := \varphi_{\text{iny}} \wedge \varphi_{\text{sobre}}$ donde

$$\varphi_{\text{iny}} = \bigwedge_{i \in D} \bigwedge_{j \in D} \left(p_{ij} \to \bigwedge_{\substack{k \in D \\ k \neq i}} \neg p_{kj} \right)$$

$$\varphi_{\text{sobre}} = \bigwedge_{j \in D} \bigvee_{i \in D} p_{ij}$$

La fórmula φ_{iny} asegura que la función sea inyectiva y $\varphi_{\text{unicidad}}$ asegura que es sobreyectiva.

Por otro lado, definimos $\varphi_{\text{aristas}} := \varphi_1 \wedge \varphi_2$

$$\varphi_1 = \bigwedge_{\{i_1,\ldots,i_n\}\in R} \bigvee_{\{j_1,\ldots,j_n\}\in R} \bigwedge_{k\in[n]} (p_{i_k j_k})$$
$$\varphi_2 = \bigwedge_{\{i_1,\ldots,i_n\}\in R} \bigvee_{\{j_1,\ldots,j_n\}\in R} \bigwedge_{k\in[n]} (p_{j_k i_k})$$

Finalmente, definimos

$$\varphi_{\text{triv}} = \bigvee_{i\in D} \neg p_{ii}$$

Con esto, podemos definir la fórmula pedida como

$$\varphi_{\text{aut}} := \varphi_{\text{función}} \wedge \varphi_{\text{biyección}} \wedge \varphi_R \wedge \varphi_{\text{triv}}$$

Ahora probemos que efectivamente φ_{aut} es satisfacible si y solo si existe un automorfismo no trivial en $\mathfrak{A}$.

($\Rightarrow$) Supongamos que existe una valuación σ tal que satisface φ_{aut}, entonces definimos la función $f : D \to D$ tal que $f(i) = j$ cuando $\sigma(p_{ij}) = 1$. En primer lugar, tenemos que esta función está bien definida por $\varphi_{\text{función}}$, ya que la fórmula imagen asegura que por cada $i \in D$ debe haber algún $\sigma(p_{ij}) = 1$, sino tendríamos que $\varphi_{\text{imagen}} = 0$. Además, al estar satisfaciéndose $\varphi_{\text{unicidad}}$ nos aseguramos que cada elemento tiene solo una imagen.
Por otro lado, podemos asegurar que la función es biyectiva porque se está satisfaciendo $\varphi_{\text{biyección}}$ esto se obtiene del hecho que φ_{iny} asegura que por cada $j \in V_2$ solo hay un $i \in V_1$ con $\sigma(p_{ij}) = 1$ y del hecho que φ_{sobre} asegura que por cada $j \in V_2$ hay por lo menos un $i \in V_1$ tal que $\sigma(p_{ij}) = 1$. Es decir, para todo $j \in V_2$ existe un único $i \in V_1$ tal que $f(i) = j$.
Por último el isomorfismo no trivial se obtiene de las fórmulas φ_R y φ_{triv}, ya que φ_1 asegura que para toda n-tupla en R tiene como imagen una tupla en R. Análogamente se tiene por φ_2 que para toda n-tupla en R tiene como preimagen una tupla en R. Además, por φ_{triv} tenemos que hay algún elemento que no es igual a la imagen, por lo tanto el automotfismo no es trivial.

($\Leftarrow$) Supongamos que tenemos un automorfismo no trivial $f : D \to D$, entonces podemos definir la valuación σ tal que

$$\sigma(p_{ij}) = \begin{cases} 1 \text{ si } f(i) = j \\ 0 \text{ en otro caso} \end{cases}$$

Veamos que esto satisface todas las fórmulas.
En primer lugar, como f es función, todo elemento tiene exactamente una imagen. Por lo tanto, se cumplen las fórmulas de $\varphi_{\text{función}}$. Además, al ser un isomorfismo, en particular se tiene que es una biyección de lo que se concluye que las fórmulas φ_{iny} y φ_{sobre} se satisfacen. Finalmente, al tener la correspondencia de tuplas en R se concluye que se satisfacen las fórmulas de φ_{aristas} y al ser no trivial se cumple φ_{triv}.

Otorgar 0,5 pts. por hacer que la oración cumpla cada uno de los siguientes: que la función esté bien definida, que sea biyección, que preserve la relación y que no sea trivial. Finalmente dar un punto por la argumentación de por qué la oración cumple lo pedido.

Pregunta 2: Complejidad

(a) Sean L_1, L_2 dos lenguajes NP-completos. ¿Es $L_1 \cap L_2$ NP-completo? Demuestre su respuesta.

(b) Sea $\mathcal{L} = \{E\}$ el vocabulario usual para grafos, i.e. E es símbolo de relación binaria. Una $\mathcal{L}$-oración φ se dice existencial si es de la forma $\varphi = \exists x_1 \exists x_2 \cdots \exists x_k \psi(x_1, x_2, \ldots, x_k)$, donce ψ es una $\mathcal{L}$-fórmula sin cuantificadores y k un entero positivo. Demuestre que el siguiente lenguaje es NP-hard

$$L = \{(\mathfrak{A}, \varphi) \mid \varphi \text{ es } \mathcal{L}\text{-oración existencial y } \mathfrak{A} \text{ es } \mathcal{L}\text{-estructura finita tal que } \mathfrak{A} \models \varphi\}$$

Sugerencia: reduzca desde 3-COL. *Sugerencia 2:* dado un grafo, construya $\mathfrak{A} = \langle A, E^{\mathfrak{A}} \rangle$ para codificar la asignación de colores a pares de nodos vecinos, de forma que A es el conjunto de colores y $E^{\mathfrak{A}}$ son pares de colores legales. ¿Cuántos elementos tiene A? ¿Qué pares de colores son ilegales en una 3-coloración y no están dentro de $E^{\mathfrak{A}}$? La oración existencial debe intentar hacer match con pares de colores para cada arista del grafo original.

Solución P2.

Parte (a) - 3 pts.

No, por ejemplo podríamos tomar el lenguaje 3-COL tal que las codificaciones de los grafos son con símbolos $\{0, 1\}$ y el mismo lenguaje pero con codificación en $\{2, 3\}$. Claramente los dos son lenguajes NP-completos con abecedario $\{0, 1, 2, 3\}$, sin embargo la intersección es vacía, lo que se puede hacer linealmente simplemente construyendo una máquina que rechaza todo input.

> Otorgar 1 pt. por decir que no se tiene que $L_1 \cap L_2$ es NP-completo y los otros 2 puntos por el contraejemplo.

Parte (b) - 3 pts.

Definamos la función f tal que $f(w) = 0$ si w no es codificación de un grafo y si w es la codificación de un grafo $G = (V, E^G)$ se define $f(w) = (\mathfrak{A}, \varphi)$ con $\mathfrak{A}$ la estructura con dominio un conjunto tres colores y $E^{\mathfrak{A}}$ la relación binaria que contiene pares de colores distintos. Por otro lado, si n es la cantidad de vértices del grafo entonces definimos la oración

$$\varphi = \exists x_1 \ldots \exists x_n \bigwedge_{(i,j) \in E^G} E(x_i, x_j)$$

Probemos que w es 3-coloreable si y solo si $f(w) \in L$.

($\Rightarrow$) Supongamos w es un grafo 3-coloreable, entonces podemos tomar x_i como el color del cual lo pintamos un la 3-coloración. Al ser esto una coloración, para todas las aristas (i, j) se tiene que los vértices en sus extremos son de colores diferentes entonces se tiene que $(x_i, x_j) \in E^{\mathfrak{A}}$, por lo tanto $\mathfrak{A} \models \varphi$.

($\Leftarrow$) Sea $f(w) \in L$, entonces tenemos que w es la codificación de un grafo. Sean $x_1, \ldots, x_n$ tales que $\bigwedge_{(i,j) \in E^G} E^{\mathfrak{A}}(x_i, x_j)$, entonces notemos que esto es una coloración del grafo porque para cada arista, sus vértices extremos son de colores diferentes.

Además esta reducción es claramente computable en tiempo polinomial. Por lo tanto, como 3-COL es NP-completo, tenemos que L es NP-hard.

> Mitad del puntaje por construir correctamente la reducción y la otra mitad del puntaje por demostrar la doble implicancia.