

Introducción

En nuestra experiencia, los buenos algoritmos son descubiertos al hacer un análisis formal y/o matemático del problema a resolver. En este caso nos proponemos el desarrollo de algoritmos para resolver un problema básico de la teoría de modelos finitos: decidir definibilidad de relaciones en el lenguaje de primer orden. La motivación inicial es la explotación de resultados teóricos recientes que caracterizan la definibilidad en términos semánticos presentados en Campercholi y Vaggione 2015.

Teoremas de definibilidad

En esta sección presentamos resultados que caracterizan la definibilidad, en diferentes fragmentos de primer orden, de relaciones sobre familias de estructuras. Estas caracterizaciones reducen la pregunta de si una relación es definible a la pregunta de si dicha relación es preservada por una colección de morfismos que depende del fragmento de primer orden considerado. Por ejemplo, (cf. referencia cruzada)...

Estos resultados, expuestos originalmente en Campercholi y Vaggione 2015, constituyen el punto de partida para nuestro estudio del chequeo computacional de la definibilidad. Los algoritmos desarrollados pueden encontrarse en las secciones siguientes.

2.1. Definibilidad por fórmulas abiertas

A continuación presentamos el teorema de caracterización para definibilidad por fórmulas abiertas. Antes necesitaremos algunos lemas.

TEOREMA 1 (Los embeddings preservan fórmulas abiertas). *Sean \mathbf{A}, \mathbf{B} estructuras y $\gamma : \mathbf{A} \rightarrow \mathbf{B}$ una función. Son equivalentes:*

1. *γ es un embedding de A en B .*
2. *Para toda fórmula abierta $\varphi(\bar{x})$ y para cada \bar{a} de A vale que:*

$$\mathbf{A} \models \varphi[\bar{a}] \Leftrightarrow \mathbf{B} \models \varphi[\gamma(\bar{a})].$$

DEMOSTRACIÓN. $1 \Rightarrow 2$)

Sea γ un embedding de A en B , sea $\varphi(x_1, \dots, x_n)$ una fórmula abierta y $\bar{a} \in A^n$, el caso base sale directo ya que los homomorfismos preservan términos. Veamos los casos inductivos:

Sea $\varphi(\bar{x}) = \neg \varphi_1(\bar{x})$ con $\varphi_1 \in F_k^T$:

$$\mathbf{A} \models \neg \varphi_1[\bar{a}] \text{ sii } \mathbf{A} \not\models \varphi_1[\bar{a}] \text{ sii } \mathbf{B} \not\models \varphi_1[\gamma(\bar{a})] \text{ sii } \mathbf{B} \models \neg \varphi_1[\gamma(\bar{a})]$$

Sea $\varphi(\bar{x}) = (\varphi_1 \eta \varphi_2)(\bar{x})$ con $\varphi_1, \varphi_2 \in F_k^T$:

$$\mathbf{A} \models (\varphi_1 \eta \varphi_2)[\bar{a}] \text{ sii } \mathbf{A} \models \varphi_1[\bar{a}] \text{ " } \eta \text{ " } \mathbf{A} \models \varphi_2[\bar{a}] \text{ sii } \mathbf{A} \models \varphi_1[\gamma(\bar{a})] \text{ " } \eta \text{ " } \mathbf{B} \models \varphi_2[\gamma(\bar{a})] \text{ sii } \mathbf{B} \models (\varphi_1 \eta \varphi_2)[\gamma(\bar{a})]$$

$2 \Rightarrow 1$)

Supongamos que para toda fórmula abierta $\varphi = \varphi(\bar{x})$ y cada $\bar{a} \in A^m$ vale que:

$$\mathbf{A} \models \varphi[\bar{a}] \Leftrightarrow \mathbf{B} \models \varphi[\gamma(\bar{a})]$$

■ Veamos que γ es inyectiva:

$$\begin{aligned} \gamma(a) &= \gamma(a') \\ \mathbf{B} &\models (x_1 \equiv x_2)[\gamma(a), \gamma(a')] \\ \mathbf{A} &\models (x_1 \equiv x_2)[a, a'] \\ a &= a' \end{aligned}$$

- Veamos que γ es un homomorfismo:
Sea $c \in \mathcal{C}$

$$\begin{aligned}\mathbf{A} &\models (c \equiv x_1)[c^{\mathbf{A}}] \\ \mathbf{B} &\models (c \equiv x_1)[\gamma(c^{\mathbf{A}})] \\ c^{\mathbf{B}} &= \gamma(c^{\mathbf{A}})\end{aligned}$$

Sea $r \in \mathcal{R}_n$

$$\begin{aligned}(a_1, \dots, a_n) &\in r^{\mathbf{A}} \\ \mathbf{A} &\models r(x_1, \dots, x_n)[a_1, \dots, a_n] \\ \mathbf{B} &\models r(x_1, \dots, x_n)[\gamma(a_1), \dots, \gamma(a_n)] \\ (\gamma(a_1), \dots, \gamma(a_n)) &\in r^{\mathbf{B}}\end{aligned}$$

Sea $f \in \mathcal{F}_n$

$$\begin{aligned}\mathbf{A} &\models (f(x_1, \dots, x_n) \equiv x_{n+1})[a_1, \dots, a_n, f^{\mathbf{A}}(a_1, \dots, a_n)] \\ \mathbf{B} &\models (f(x_1, \dots, x_n) \equiv x_{n+1})[\gamma(a_1), \dots, \gamma(a_n), \gamma(f^{\mathbf{A}}(a_1, \dots, a_n))] \\ f^{\mathbf{B}}(\gamma(a_1), \dots, \gamma(a_n)) &= \gamma(f^{\mathbf{A}}(a_1, \dots, a_n))\end{aligned}$$

- Veamos que γ es embedding:
Sea $r \in \mathcal{R}_n$

$$\begin{aligned}(\gamma(a_1), \dots, \gamma(a_n)) &\in r^{\mathbf{B}} \\ \mathbf{B} &\models r(x_1, \dots, x_n)[\gamma(a_1), \dots, \gamma(a_n)] \\ \mathbf{A} &\models r(x_1, \dots, x_n)[a_1, \dots, a_n] \\ (a_1, \dots, a_n) &\in r^{\mathbf{A}}\end{aligned}$$

□

TEOREMA 2. Si \mathbf{A} es una subestructura de \mathbf{B} y $\varphi(\bar{x})$ es una fórmula abierta, entonces para cada $\bar{a} \in A^n$ vale que

$$\mathbf{A} \models \varphi[\bar{a}] \Leftrightarrow \mathbf{B} \models \varphi[\bar{a}]$$

DEMOSTRACIÓN. Sea \mathbf{A} una subestructura de \mathbf{B} y $\varphi(\bar{x})$ una fórmula abierta. Como \mathbf{A} es subestructura es cerrada sobre r, f y $c^{\mathbf{B}} \in A$

Sea $\gamma : \mathbf{A} \rightarrow \mathbf{B}$ tal que para todo $x \in A$, $\gamma(x) = x$. Como $\mathbf{A} \leq \mathbf{B}$, es directo que γ es un embedding. Entonces por el Teorema 1, $\mathbf{A} \models \varphi[\bar{a}] \Leftrightarrow \mathbf{B} \models \varphi[\gamma(\bar{a})]$

Finalmente, por definición de γ :

$$\mathbf{A} \models \varphi[\bar{a}] \Leftrightarrow \mathbf{B} \models \varphi[\bar{a}]$$

□

NOTACIÓN 3. Dado un conjunto de fórmulas Δ , escribiremos $\Delta(\bar{x})$ para anunciar que cada una de las fórmulas en Δ tiene sus variables libres contenidas en la tupla \bar{x} , y que consideramos cada $\delta \in \Delta(\bar{x})$ declarada $\delta = \delta(\bar{x})$. Si \mathbf{A} es una estructura y \bar{a} es una tupla de elementos de A , escribiremos $\mathbf{A} \models \Delta[\bar{a}]$ cuando $\mathbf{A} \models \delta[\bar{a}]$ para cada $\delta \in \Delta(\bar{x})$.

DEFINICIÓN 4. Sea \mathbf{A} una estructura y sean $a_1, \dots, a_n \in A$. Definimos el diagrama abierto para a_1, \dots, a_n en \mathbf{A} como:

$$\Delta_{\mathbf{A}, \bar{a}}(x_1, \dots, x_n) := \{\alpha \mid \alpha \text{ es abierta y } \mathbf{A} \models \alpha[\bar{a}]\}$$

TEOREMA 5. Sea \mathbf{A} una estructura y $b_1, \dots, b_n \in B$, son equivalentes:

1. $\mathbf{B} \models \Delta_{\mathbf{A}\bar{a}} [\bar{b}]$
2. Hay un isomorfismo γ de $\langle \bar{a} \rangle^{\mathbf{A}}$ en $\langle \bar{b} \rangle^{\mathbf{B}}$ tal que $\gamma(\bar{a}) = \bar{b}$

DEMOSTRACIÓN. Veamos $1 \Rightarrow 2$:

Supongamos que $\mathbf{B} \models \Delta_{\mathbf{A}\bar{a}} [\bar{b}]$

Si α fórmula abierta y $\mathbf{A} \models \alpha [\bar{a}]$ entonces $\mathbf{B} \models \alpha [\bar{b}]$

Tomo $\langle \bar{a} \rangle^{\mathbf{A}} = \{t^{\mathbf{A}}[a_1, \dots, a_n] : t \in T^r, a_1, \dots, a_n \in \bar{a}\}$ y $\langle \bar{b} \rangle^{\mathbf{A}} = \{t^{\mathbf{A}}[b_1, \dots, b_n] : t \in T^r, b_1, \dots, b_n \in \bar{b}\}$

Defino

$$\begin{aligned} \gamma &: A' \rightarrow B' \\ \gamma(t^{\mathbf{A}}[a_1, \dots, a_n]) &= t^{\mathbf{B}}[b_1, \dots, b_n] \end{aligned}$$

Es claro que γ es un homomorfismo.

- Veamos que γ es inyectivo

Sean $a'_1, a'_2 \in A'$ tales que $a'_1 \neq a'_2$

$$\begin{aligned} a'_1 &\neq a'_2 \\ t_1^{\mathbf{A}}[\bar{a}] &\neq t_2^{\mathbf{A}}[\bar{a}] \\ \mathbf{A} &\models \neg(t_1 \equiv t_2)[\bar{a}] \end{aligned} \tag{2.1.1}$$

$$\begin{aligned} \mathbf{B} &\models \neg(t_1 \equiv t_2)[\bar{b}] \\ t_1^{\mathbf{B}}[\bar{b}] &\neq t_2^{\mathbf{B}}[\bar{b}] \\ \gamma(t_1^{\mathbf{A}}[\bar{a}]) &\neq \gamma(t_2^{\mathbf{A}}[\bar{a}]) \end{aligned} \tag{2.1.2}$$

- Veamos que γ es sobreyectivo

Sea $b' \in B'$

$$\begin{aligned} b' &\in B' \\ t^{\mathbf{B}}[\bar{b}] &\in B' \end{aligned}$$

Tomo $t^{\mathbf{A}}[\bar{a}] = a'$

$$\gamma(t^{\mathbf{A}}[\bar{a}]) = t^{\mathbf{B}}[\bar{b}]$$

Por lo tanto γ es un isomorfismo y $\gamma(a_j) = \gamma(x_j^{\mathbf{A}}[a_1, \dots, a_n]) = x_j^{\mathbf{B}}[b_1, \dots, b_n] = b_j$.

Ahora veamos $2 \Rightarrow 1$:

Supongamos que hay un isomorfismo γ de $\langle \bar{a} \rangle^{\mathbf{A}}$ en $\langle \bar{b} \rangle^{\mathbf{B}}$ tal que $\gamma(\bar{a}) = \bar{b}$

Entonces, es claro que γ es un embedding de $\langle \bar{a} \rangle^{\mathbf{A}}$ en $\langle \bar{b} \rangle^{\mathbf{B}}$.

Por el Teorema 1 para toda fórmula abierta $\varphi(\bar{x})$ y cada $\bar{a} \in A'$

$$\langle \bar{a} \rangle^{\mathbf{A}} \models \varphi[\bar{a}] \Leftrightarrow \langle \bar{b} \rangle^{\mathbf{B}} \models \varphi[\gamma(\bar{a})]$$

Por el Teorema 2 como $\langle \bar{a} \rangle^{\mathbf{A}}$ es subestructura de \mathbf{A} , $\langle \bar{b} \rangle^{\mathbf{B}}$ lo es de \mathbf{B} y φ es abierta

$$\mathbf{A} \models \varphi[\bar{a}] \Leftrightarrow \mathbf{B} \models \varphi[\gamma(\bar{a})]$$

$$\mathbf{A} \models \varphi[\bar{a}] \Leftrightarrow \mathbf{B} \models \varphi[\bar{b}]$$

Por lo tanto si α es abierta y $\mathbf{A} \models \alpha[\bar{a}]$ entonces $\mathbf{B} \models \alpha[\bar{b}]$

Finalmente, $\mathbf{B} \models \Delta_{\mathbf{A}\bar{a}} [\bar{b}]$

□

DEFINICIÓN 6. Dos fórmulas $\alpha(\bar{x})$ y $\beta(\bar{x})$ se dicen *equivalentes* sobre una familia de estructuras \mathcal{K} si para cada $\mathbf{A} \in \mathcal{K}$ y cada \bar{a} de \mathbf{A} vale que

$$\mathbf{A} \models \alpha[\bar{a}] \iff \mathbf{A} \models \beta[\bar{a}].$$

TEOREMA 7 (Modulo equivalencia sobre una estructura finita, la cantidad de fórmulas en x_1, \dots, x_n es finita). Sea \mathcal{K} una clase finita de estructuras finitas, y sean x_1, \dots, x_n variables.

Hay un conjunto finito de fórmulas $\Sigma(\bar{x})$ tal que para toda fórmula $\varphi(\bar{x})$ hay $\sigma(\bar{x}) \in \Sigma(\bar{x})$ tal que $\varphi(\bar{x})$ y $\sigma(\bar{x})$ son equivalentes sobre \mathcal{K} .

DEMOSTRACIÓN. Sea $\mathcal{K} = \{\mathbf{A}_1, \dots, \mathbf{A}_m\}$. Veamos primero que las fórmulas son finitas modulo equivalencia. Sea $\varphi(\bar{x})$ defino $T_{\varphi\mathcal{K}} = \{\bar{a} \in A_1^n \mid \mathbf{A}_1 \models \varphi[\bar{a}]\} \times \dots \times \{\bar{a} \in A_m^n \mid \mathbf{A}_m \models \varphi[\bar{a}]\}$ y supongamos φ equivalente ψ en \mathcal{K} , entonces

$$(\bar{a}_1, \dots, \bar{a}_m) \in T_{\varphi\mathcal{K}} \iff (\bar{a}_1, \dots, \bar{a}_m) \in T_{\psi\mathcal{K}} \text{ para cada } (\bar{a}_1, \dots, \bar{a}_m) \in A_1^n \times \dots \times A_m^n$$

lo que significa que $T_{\varphi\mathcal{K}} = T_{\psi\mathcal{K}}$. Entonces basta con contar los subconjuntos de $A_1^n \times \dots \times A_m^n$, como $|A_1^n \times \dots \times A_m^n| = |A_1|^n \cdot \dots \cdot |A_m|^n$ y cada A_i era finito, es claro que $A_1^n \times \dots \times A_m^n$ es finito. Por lo tanto $\mathcal{P}(A^n)$ también lo es.

Ahora veamos que existe $\Sigma(\bar{x})$. Sean $T_1, \dots, T_k \subseteq A_1^n \times \dots \times A_m^n$ tales que existe $\varphi_i(\bar{x})$ tal que para cada $\mathbf{A} \in \mathcal{K}$, $\mathbf{A} \models \varphi_i[T_i]$ y sea $\psi(\bar{x})$, tomo $T_{\psi\mathcal{K}}$ como T_1, \dots, T_k es la sucesión de todos los subconjuntos que se definen con una fórmula, hay un j tal que $T_j = T_{\psi\mathcal{K}}$. Entonces, para cada $\mathbf{A} \in \mathcal{K}$, $\mathbf{A} \models \psi[\bar{a}] \iff \mathbf{A} \models \varphi_j[\bar{a}]$. \square

TEOREMA 8. Sean \mathbf{A}, \mathbf{B} estructuras finitas y $a_1, \dots, a_n \in A$, entonces existe una fórmula abierta $\varphi(\bar{x})$ tal que para todo $\bar{b} \in B^n$ son equivalentes:

1. $\mathbf{B} \models \varphi[\bar{b}]$
2. Hay un isomorfismo γ de $\langle \bar{a} \rangle^{\mathbf{A}}$ en $\langle \bar{b} \rangle^{\mathbf{B}}$ tal que $\gamma(\bar{a}) = \bar{b}$

DEMOSTRACIÓN. Tomo $\varphi = \bigwedge_{\alpha \in \Delta_{\mathbf{A}\bar{a}}} \alpha(\bar{x})$ que es una fórmula, por el Teorema 7.

$$\text{Es claro que } \mathbf{B} \models \bigwedge_{\alpha \in \Delta_{\mathbf{A}\bar{a}}} \alpha(\bar{x})[\bar{b}] \iff \mathbf{B} \models \Delta_{\mathbf{A}\bar{a}}[\bar{b}]$$

Entonces por Teorema 5, hay un isomorfismo γ de $\langle \bar{a} \rangle^{\mathbf{A}}$ en $\langle \bar{b} \rangle^{\mathbf{B}}$ tal que $\gamma(\bar{a}) = \bar{b}$. \square

DEFINICIÓN 9. Sean $\mathcal{L} \subseteq \mathcal{L}'$ lenguajes de primer orden y $R \in \mathcal{L}' - \mathcal{L}$ un símbolo de relación n-ario. Diremos que R es *definible* en una familia \mathcal{K} de \mathcal{L}' -estructuras cuando exista una \mathcal{L} -fórmula $\varphi(x_1, \dots, x_n)$ tal que para todo $\mathbf{A} \in \mathcal{K}$ y todas $a_1, \dots, a_n \in A$

$$(a_1, \dots, a_n) \in R^{\mathbf{A}} \iff \varphi(a_1, \dots, a_n)$$

Si $\mathcal{L} \subseteq \mathcal{L}'$ son lenguajes de primer orden, para una \mathcal{L}' -estructura \mathbf{A} , usaremos $\mathbf{A}_{\mathcal{L}}$ para indicar el reducto de \mathbf{A} al lenguaje \mathcal{L} . Si \mathbf{A}, \mathbf{B} son \mathcal{L} -estructuras, usaremos $\mathbf{A} \leq \mathbf{B}$ para expresar que \mathbf{A} es subestructura de \mathbf{B} .

Sean:

$$Fo(\mathcal{L}) = \{\varphi : \varphi \text{ es una } \mathcal{L}\text{-fórmula}\}$$

$$E(\mathcal{L}) = \{\varphi : \varphi \text{ es una } \mathcal{L}\text{-fórmula existencial}\}$$

$$E^+(\mathcal{L}) = \{\varphi : \varphi \text{ es una } \mathcal{L}\text{-fórmula existencial positiva}\}$$

$$Op(\mathcal{L}) = \{\varphi : \varphi \text{ es una } \mathcal{L}\text{-fórmula abierta}\}$$

$$Op^+(\mathcal{L}) = \{\varphi : \varphi \text{ es una } \mathcal{L}\text{-fórmula abierta positiva}\}$$

$$\pm At(\mathcal{L}) = At(\mathcal{L}) \cup \{\neg\alpha : \alpha \in At(\mathcal{L})\}$$

$$At(\mathcal{L}) = \{\mathcal{L}\text{-fórmulas atómicas}\}$$

DEFINICIÓN 10. Dados A, B conjuntos, $R^A \subseteq A^n$, $R^B \subseteq B^n$, diremos que una función $\gamma : A \rightarrow B$ preserva R si para toda tupla $(a_1, \dots, a_n) \in A_0$ tenemos que si $(a_1, \dots, a_n) \in R^A$ implica que $(\sigma(a_1), \dots, \sigma(a_n)) \in R^B$.

TEOREMA 11. Sean $\mathcal{L} \subseteq \mathcal{L}'$ lenguajes de primer orden, sea $R \in \mathcal{L}' - \mathcal{L}$ un símbolo de relación n -ario. Para una clase finita \mathcal{K} de \mathcal{L}' -estructuras, los siguientes son equivalentes:

1. Hay una fórmula en $Op(\mathcal{L})$ que define R en \mathcal{K} .
2. Para todas $\mathbf{A}, \mathbf{B} \in \mathcal{K}$, todas $\mathbf{A}_0 \leq \mathbf{A}_{\mathcal{L}}, \mathbf{B}_0 \leq \mathbf{B}_{\mathcal{L}}$, todo isomorfismo $\sigma : \mathbf{A}_0 \rightarrow \mathbf{B}_0$ preserva R .

DEMOSTRACIÓN. $1 \Rightarrow 2$) Sea $\varphi(\bar{x})$ la fórmula que define R en \mathcal{K} , sean $\mathbf{A}, \mathbf{B} \in \mathcal{K}$ y sean $\mathbf{A}_0 \leq \mathbf{A}_{\mathcal{L}}$ y $\mathbf{B}_0 \leq \mathbf{B}_{\mathcal{L}}$ tales que $\sigma : \mathbf{A}_0 \rightarrow \mathbf{B}_0$ sea un isomorfismo. Sea $\bar{a} \in R^{\mathbf{A}_0} \cap A_0$ veamos que $\sigma(\bar{a}) \in R^{\mathbf{B}_0}$. Como $\bar{a} \in R^{\mathbf{A}_0} \Rightarrow \bar{a} \in R^{\mathbf{A}} \Leftrightarrow \mathbf{A} \models \varphi[\bar{a}]$, por Teorema 2, $\mathbf{A}_0 \models \varphi[\bar{a}]$ y como \mathbf{A}_0 y \mathbf{B}_0 son isomorfos por σ , $\mathbf{B}_0 \models \varphi[\sigma(\bar{a})]$. Nuevamente por Teorema 2 $\mathbf{B} \models \varphi[\sigma(\bar{a})]$ y como por hipótesis φ define a R en \mathcal{K} , $\sigma(\bar{a}) \in R^{\mathbf{B}}$.

$2 \Rightarrow 1$) Sea $\varphi = \bigvee_{\mathbf{A} \in \mathcal{K}} (\bigvee_{\bar{a} \in R^{\mathbf{A}}} (\bigwedge_{\alpha \in \Delta_{\mathbf{A}\bar{a}}} \alpha(\bar{x})))$, la cual es fórmula por el Teorema 7. Veamos que para cada $\mathbf{B} \in \mathcal{K}$, $\mathbf{B} \models \varphi[\bar{b}]$ sii $\bar{b} \in R^{\mathbf{B}}$. Sea $\mathbf{B} \in \mathcal{K}$:

\Rightarrow) Supongamos $\mathbf{B} \models \varphi[\bar{b}]$, entonces en particular para algún $\mathbf{A} \in \mathcal{K}$ y $\bar{a} \in R^{\mathbf{A}}$, $\mathbf{B} \models \bigwedge_{\alpha \in \Delta_{\mathbf{A}\bar{a}}} \alpha(\bar{x})[\bar{b}]$. Por Teorema 8, hay un isomorfismo $\gamma : \langle \bar{a} \rangle^{\mathbf{A}} \rightarrow \langle \bar{b} \rangle^{\mathbf{B}}$ tal que $\gamma(\bar{a}) = \bar{b}$. Entonces por hipótesis γ preserva R . Finalmente como $\bar{a} \in R^{\mathbf{A}}$, entonces $\gamma(\bar{a}) \in R^{\mathbf{B}}$ o sea que $\bar{b} \in R^{\mathbf{B}}$.

\Leftarrow) Supongamos $\bar{b} \in R^{\mathbf{B}}$. Entonces $\varphi = \bigvee_{\mathbf{A} \in \mathcal{K}} (\bigvee_{\bar{a} \in R^{\mathbf{A}}} (\bigwedge_{\alpha \in \Delta_{\mathbf{A}\bar{a}}} \alpha(\bar{x}))) = \dots \vee \dots \vee (\bigwedge_{\alpha \in \Delta_{\mathbf{B}\bar{b}}} \alpha(\bar{x}))$, evidentemente $\mathbf{B} \models \Delta_{\mathbf{B}\bar{b}}[\bar{b}]$, entonces $\mathbf{B} \models \varphi[\bar{b}]$. \square

2.2. Definibilidad por fórmulas abiertas positivas y conjunción de atómicas

TEOREMA 12. Sean \mathbf{A}, \mathbf{B} estructuras y $h : A \rightarrow B$ una función, son equivalentes:

1. h es un homomorfismo de \mathbf{A} en \mathbf{B} .
2. Para toda fórmula atómica $\varphi = \varphi(\bar{x})$ y para cada \bar{a} de A vale que

$$\mathbf{A} \models \varphi[\bar{a}] \implies \mathbf{B} \models \varphi[h(\bar{a})].$$

3. Para toda fórmula abierta positiva¹ $\varphi = \varphi(\bar{x})$ y para cada \bar{a} de A vale que

$$\mathbf{A} \models \varphi[\bar{a}] \implies \mathbf{B} \models \varphi[h(\bar{a})].$$

DEMOSTRACIÓN. $1 \Rightarrow 2$) Directo, ya que los homomorfismos preservan términos.

$2 \Rightarrow 1$) Supongamos $\mathbf{A} \models \varphi[\bar{a}] \Rightarrow \mathbf{B} \models \varphi[h(\bar{a})]$ con φ atómica. Veamos que h es homomorfismo.

Sea $c \in \mathcal{C}$

$$\mathbf{A} \models (c \equiv x_1) [c^{\mathbf{A}}] \Rightarrow \mathbf{B} \models (c \equiv x_1) [h(c^{\mathbf{A}})] \Leftrightarrow c^{\mathbf{B}} = h(c^{\mathbf{A}})$$

¹Una fórmulas es positiva si no tiene ocurrencias de \neg , \rightarrow , \leftrightarrow .

Sea $f \in \mathcal{F}_n$

$$\mathbf{A} \models (f(x_1, \dots, x_n) \equiv x_{n+1}) [a_1, \dots, a_n, f^{\mathbf{A}}(\bar{a})]$$

$$\mathbf{B} \models (f(x_1, \dots, x_n) \equiv x_{n+1}) [h(a_1), \dots, h(\bar{a}_n), h(f^{\mathbf{A}}(\bar{a}))]$$

$$f^{\mathbf{B}}(h(\bar{a})) = h(f^{\mathbf{A}}(\bar{a}))$$

Sea $r \in \mathcal{R}_n$

$$\bar{a} \in r^{\mathbf{A}} \Rightarrow \mathbf{A} \models r(\bar{x}) [\bar{a}] \Rightarrow \mathbf{B} \models r(\bar{x}) [h(\bar{a})] \Rightarrow h(\bar{a}) \in r^{\mathbf{B}}$$

2 \Rightarrow 3) Rutina.

3 \Rightarrow 2) Directo ya que toda atómica es abierta positiva. \square

DEFINICIÓN 13. Sea \mathbf{A} una estructura y sean $a_1, \dots, a_n \in A$. Definimos el diagrama atómico positivo de \bar{a} en \mathbf{A} como

$$\Delta_{\mathbf{A}, \bar{a}}^+(x_1, \dots, x_n) := \{\alpha \mid \alpha \text{ es atómica y } \mathbf{A} \models \alpha[\bar{a}]\}.$$

TEOREMA 14. Sean \mathbf{B} una estructura y $b_1, \dots, b_n \in B$, son equivalentes:

1. $\mathbf{B} \models \Delta_{\mathbf{A}, \bar{a}}^+[\bar{b}]$.
2. Hay un homomorfismo h de $\langle \bar{a} \rangle^{\mathbf{A}}$ en $\langle \bar{b} \rangle^{\mathbf{B}}$ tal que $h(\bar{a}) = \bar{b}$.

DEMOSTRACIÓN. 1 \Rightarrow 2) Defino $h : \langle \bar{a} \rangle^{\mathbf{A}} \rightarrow \langle \bar{b} \rangle^{\mathbf{B}}$, como

$$h(t^{\mathbf{A}}[\bar{a}]) = t^{\mathbf{B}}[\bar{b}]$$

El cual es se ve fácilmente que es un homomorfismo que cumple $h(\bar{a}) = \bar{b}$.

2 \Rightarrow 1) Supongamos que hay un homomorfismo h de $\langle \bar{a} \rangle^{\mathbf{A}}$ en $\langle \bar{b} \rangle^{\mathbf{B}}$ tal que $h(\bar{a}) = \bar{b}$, entonces como $\mathbf{A} \models \Delta_{\mathbf{A}, \bar{a}}^+[\bar{a}]$, por el Teorema 12 $\mathbf{B} \models \Delta_{\mathbf{A}, \bar{a}}^+[h(\bar{a})]$, entonces $\mathbf{B} \models \Delta_{\mathbf{A}, \bar{a}}^+[\bar{b}]$ \square

TEOREMA 15. Sean \mathbf{A}, \mathbf{B} estructuras finitas y $a_1, \dots, a_n \in A$, entonces existe una fórmula abierta $\varphi(\bar{x})$ tal que para todo $\bar{b} \in B^n$ son equivalentes:

1. $\mathbf{B} \models \varphi[\bar{b}]$
2. Hay un homomorfismo h de $\langle \bar{a} \rangle^{\mathbf{A}}$ en $\langle \bar{b} \rangle^{\mathbf{B}}$ tal que $h(\bar{a}) = \bar{b}$

DEMOSTRACIÓN. Tomo $\varphi = \bigwedge_{\alpha \in \Delta_{\mathbf{A}, \bar{a}}^+} \alpha(\bar{x})$ que es una fórmula, por el Teorema 7.

Es claro que $\mathbf{B} \models \varphi[\bar{b}] \Leftrightarrow \mathbf{B} \models \Delta_{\mathbf{A}, \bar{a}}^+[\bar{b}]$

Entonces por Teorema 14, hay un homomorfismo h de $\langle \bar{a} \rangle^{\mathbf{A}}$ en $\langle \bar{b} \rangle^{\mathbf{B}}$ tal que $h(\bar{a}) = \bar{b}$ \square

TEOREMA 16. Sean $\mathcal{L} \subseteq \mathcal{L}'$ lenguajes de primer orden, sea $R \in \mathcal{L}' - \mathcal{L}$ un símbolo de relación n -ario. Para una clase finita \mathcal{K} de \mathcal{L}' -estructuras, los siguientes son equivalentes:

1. Hay una \mathcal{L} -fórmula abierta positiva que define R en \mathcal{K} .
2. Para todas $\mathbf{A}, \mathbf{B} \in \mathcal{K}$, todas $\mathbf{A}_0 \leq \mathbf{A}_{\mathcal{L}}, \mathbf{B}_0 \leq \mathbf{B}_{\mathcal{L}}$, todo homomorfismo $h : \mathbf{A}_0 \rightarrow \mathbf{B}_0$ preserva R .

DEMOSTRACIÓN. Veamos 1 \Rightarrow 2:

Sea $\varphi(\bar{x})$ la fórmula abierta positiva que define R en \mathcal{K} , sean $\mathbf{A}, \mathbf{B} \in \mathcal{K}$ y sean $\mathbf{A}_0 \leq \mathbf{A}_{\mathcal{L}}$ y $\mathbf{B}_0 \leq \mathbf{B}_{\mathcal{L}}$ tales que $h : \mathbf{A}_0 \rightarrow \mathbf{B}_0$ sea un homomorfismo. Sea $\bar{a} \in R^{\mathbf{A}_0} \cap A_0$ veamos que $h(\bar{a}) \in R^{\mathbf{B}_0}$:

$\bar{a} \in R^{\mathbf{A}_0} \Rightarrow \bar{a} \in R^{\mathbf{A}} \Leftrightarrow \mathbf{A} \models \varphi[\bar{a}]$, por Teorema 2, $\mathbf{A}_0 \models \varphi[\bar{a}]$ y como h es un homomorfismo de \mathbf{A}_0 en \mathbf{B}_0 y φ es abierta positiva por Teorema 12, $\mathbf{B}_0 \models \varphi[h(\bar{a})]$. Nuevamente por Teorema 2 $\mathbf{B} \models \varphi[h(\bar{a})]$ y como por hipótesis φ define a R en \mathcal{K} , $h(\bar{a}) \in R^{\mathbf{B}}$.

Veamos $2 \Rightarrow 1$:

Sea $\varphi = \bigvee_{\mathbf{A} \in \mathcal{K}} \left(\bigvee_{\bar{a} \in R^{\mathbf{A}}} \left(\bigwedge_{\alpha \in \Delta_{\bar{a}}^+} \alpha(\bar{x}) \right) \right)$, la cual es fórmula por el Teorema 7. Sea $\mathbf{B} \in \mathcal{K}$:

\Rightarrow) Supongamos $\mathbf{B} \models \varphi[\bar{b}]$, entonces en particular para algún $\mathbf{A} \in \mathcal{K}$ y $\bar{a} \in R^{\mathbf{A}}$, $\mathbf{B} \models \Delta_{\bar{a}}^+[\bar{b}]$. Por Teorema 15, hay un homomorfismo $h : \langle \bar{a} \rangle^{\mathbf{A}} \rightarrow \langle \bar{b} \rangle^{\mathbf{B}}$ tal que $h(\bar{a}) = \bar{b}$. Entonces por hipótesis h preserva R . Finalmente como $\bar{a} \in R^{\mathbf{A}}$, entonces $h(\bar{a}) \in R^{\mathbf{B}}$ o sea que $\bar{b} \in R^{\mathbf{B}}$.

\Leftarrow) Supongamos $\bar{b} \in R^{\mathbf{B}}$. Entonces $\varphi = \dots \vee \dots \vee \left(\bigwedge_{\alpha \in \Delta_{\bar{b}}^+} \alpha(\bar{x}) \right)$, evidentemente $\mathbf{B} \models \Delta_{\bar{b}}^+[\bar{b}]$, entonces $\mathbf{B} \models \varphi[\bar{b}]$. \square

DEFINICIÓN 17. Sean A_1, \dots, A_m, B conjuntos, $R^{A_i} \subseteq A_i^n$ para cada $i \in [1, m]$, $R^B \subseteq B^n$ y $h : D \subseteq A_1 \times \dots \times A_m \rightarrow B$ una función. Diremos que h *preserva R* si dados

$$\bar{a}_1 = (a_{11}, \dots, a_{1m}), \dots, \bar{a}_n = (a_{n1}, \dots, a_{nm}) \in D$$

tales que

$$(a_{1i}, \dots, a_{ni}) \in R^{A_i} \text{ para cada } i \in [1, m]$$

se tiene que $(h(\bar{a}_1), \dots, h(\bar{a}_n)) \in R^B$.

LEMA 18. Sean $\mathbf{A}_1, \dots, \mathbf{A}_m$ estructuras y $\varphi = \exists \bar{y} \psi(\bar{x}, \bar{y})$ con ψ una conjunción de fórmulas atómicas. Entonces son equivalentes:

1. Para todo $i = 1, \dots, m$ se da que $\mathbf{A}_i \models \varphi[a_{1i}, \dots, a_{ni}]$
2. $\mathbf{A}_1 \times \dots \times \mathbf{A}_m \models \varphi[\bar{a}_1, \dots, \bar{a}_n]$, con $\bar{a}_j = (a_{j1}, \dots, a_{jm})$

DEMOSTRACIÓN. Rutina. \square

TEOREMA 19. Sean $\mathcal{L} \subseteq \mathcal{L}'$ lenguajes de primer orden, sea $R \in \mathcal{L}' - \mathcal{L}$ un símbolo de relación n -ario. Para una clase finita \mathcal{K} de \mathcal{L}' -estructuras, los siguientes son equivalentes:

1. Para cada $m \geq 1$, para todas $\mathbf{A}_1, \dots, \mathbf{A}_m, \mathbf{B} \in \mathcal{K}$, para cada $\mathbf{S} \leq (\mathbf{A}_1 \times \dots \times \mathbf{A}_m)_{\mathcal{L}}$, todo homomorfismo $h : \mathbf{S} \rightarrow \mathbf{B}_{\mathcal{L}}$ preserva R .
2. Hay una conjunción finita de \mathcal{L} -fórmulas atómicas que define R en \mathcal{K} .

DEMOSTRACIÓN. $2 \Rightarrow 1$) Sea $\varphi(\bar{x})$ conjunción de atómicas que define a R en \mathcal{K} . Sean $m \geq 1$, $\mathbf{A}_1, \dots, \mathbf{A}_m, \mathbf{B} \in \mathcal{K}$, $\mathbf{S} \leq (\mathbf{A}_1 \times \dots \times \mathbf{A}_m)_{\mathcal{L}}$, y $h : \mathbf{S} \rightarrow \mathbf{B}_{\mathcal{L}}$ un homomorfismo. Sean

$$\bar{s}_1 = (s_{11}, \dots, s_{1m}), \dots, \bar{s}_n = (s_{n1}, \dots, s_{nm}) \in S$$

tales que

$$(s_{1i}, \dots, s_{ni}) \in R^{A_i} \text{ para cada } i \in \{1, \dots, m\}.$$

Veamos que $(h(\bar{s}_1), \dots, h(\bar{s}_n)) \in R^B$. Como $(s_{1i}, \dots, s_{ni}) \in R^{A_i}$ y φ define a R en \mathcal{K} , tenemos que

$$\mathbf{A}_i \models \varphi[s_{1i}, \dots, s_{ni}] \text{ para } i \in \{1, \dots, m\}.$$

Luego, por el Lema 18,

$$\mathbf{A}_1 \times \dots \times \mathbf{A}_m \models \varphi[\bar{s}_1, \dots, \bar{s}_n].$$

Como φ es abierta y $\mathbf{S} \leq (\mathbf{A}_1 \times \dots \times \mathbf{A}_m)_{\mathcal{L}}$, el Teorema 2 implica que $\mathbf{S} \models \varphi[\bar{s}_1, \dots, \bar{s}_n]$, y aplicando el Teorema 12 obtenemos que $\mathbf{B} \models \varphi[h(\bar{s}_1), \dots, h(\bar{s}_n)]$. Finalmente, como φ define a R en \mathcal{K} y $\mathbf{B} \in \mathcal{K}$, concluimos que $(h(\bar{s}_1), \dots, h(\bar{s}_n)) \in R^B$.

$1 \Rightarrow 2$) Supongamos $\mathcal{K} = \{\mathbf{A}_1, \dots, \mathbf{A}_m\}$, R n -aria. Sean $\bar{x}_1, \dots, \bar{x}_n \in \mathbf{A}_1^{|R^{\mathbf{A}_1}|} \times \dots \times \mathbf{A}_m^{|R^{\mathbf{A}_m}|}$ tales que $\bar{x}_i = \left(x_{i1}^1, \dots, x_{i1}^1|_{R^{\mathbf{A}_1}|}, \dots, x_{i1}^m, \dots, x_{i1}^m|_{R^{\mathbf{A}_m}|} \right)$ para cada

$i \in \{1, \dots, m\}$ con $\bigcup R^{\mathbf{A}^j} = \left\{ \left(x_{1i}^j, \dots, x_{ni}^j \right) \text{ con } i \in \{1, |R^{\mathbf{A}^j}|\} \right\}$ para cada $i \in \{1, \dots, m\}$. Sea

$$\varphi = \bigwedge_{\alpha \in \Delta_{\mathbf{A}_1^{|\mathbf{A}^1|} \times \dots \times \mathbf{A}_m^{|\mathbf{A}^m|}, (\bar{x}_1, \dots, \bar{x}_n)}^+} \alpha(\bar{x})$$

. Supongamos $\mathbf{B} \models \varphi[b_1, \dots, b_n]$ y veamos que $(b_1, \dots, b_n) \in R^{\mathbf{B}}$. Como $\mathbf{B} \models \Delta_{\mathbf{A}_1^{|\mathbf{A}^1|} \times \dots \times \mathbf{A}_m^{|\mathbf{A}^m|}, (\bar{x}_1, \dots, \bar{x}_n)}^+[b_1, \dots, b_n]$, por Teorema 14 hay h homomorfismo de $\langle \bar{x}_1, \dots, \bar{x}_n \rangle_{\mathbf{A}_1^{|\mathbf{A}^1|} \times \dots \times \mathbf{A}_m^{|\mathbf{A}^m|}}$ en $\langle b_1, \dots, b_n \rangle^{\mathbf{B}}$ tal que $h(\bar{x}_i) = b_i$. Como por hipótesis h preserva R , entonces como $(x_{1i}^j, \dots, x_{ni}^j) \in R^{\mathbf{A}^j}$, $(h(\bar{x}_1), \dots, h(\bar{x}_n)) \in R^{\mathbf{B}}$, que es exactamente $(b_1, \dots, b_n) \in R^{\mathbf{B}}$.

Ahora supongamos $(b_1, \dots, b_n) \in R^{\mathbf{B}}$ y veamos que $\mathbf{B} \models \varphi[b_1, \dots, b_n]$, entonces hay algún j tal que $\mathbf{A}_j = \mathbf{B}$ y hay algún k y algún j tal que

$$(x_{1k}^j, \dots, x_{nk}^j) = (b_1, \dots, b_n)$$

. Entonces como $\mathbf{A}_1^{|\mathbf{A}^1|} \times \dots \times \mathbf{B}^{|\mathbf{B}|} \times \dots \times \mathbf{A}_m^{|\mathbf{A}^m|} \models \varphi[\bar{x}_1, \dots, \bar{x}_n]$ por Teorema 18 $\mathbf{B} \models \varphi[x_{1k}^j, \dots, x_{nk}^j]$, por lo tanto $\mathbf{B} \models \varphi[b_1, \dots, b_n]$. \square

TEOREMA 20. *Sea \mathbf{A} una estructura finita y $a_1, \dots, a_n \in A$. Entonces hay una fórmula existencial $\varphi(x_1, \dots, x_n)$ tal que para toda estructura \mathbf{B} y para todo $\bar{b} \in B^n$ los siguientes son equivalentes:*

1. $\mathbf{B} \models \varphi[\bar{b}]$.
2. Hay un embedding $\gamma : \mathbf{A} \rightarrow \mathbf{B}$ tal que $\gamma(\bar{a}) = \bar{b}$

DEMOSTRACIÓN. Sean a'_1, \dots, a'_m tales que $\langle a_1, \dots, a_n, a'_1, \dots, a'_m \rangle^{\mathbf{A}} = \mathbf{A}$ y sea

$$\varphi = \exists y_1, \dots, y_m \left(\bigwedge_{\alpha \in \Delta_{(\bar{a}, \bar{a}'), \mathbf{A}}} \alpha(\bar{x}, \bar{y}) \right)$$

que es fórmula por el Teorema 7.

Supongamos $\mathbf{B} \models \varphi[\bar{b}]$, entonces existen $b'_1, \dots, b'_m \in B$ tales que $\mathbf{B} \models \bigwedge_{\alpha \in \Delta_{(\bar{a}, \bar{a}'), \mathbf{A}}} \alpha(\bar{x}, \bar{y}) [\bar{b}, \bar{b}']$ por Teorema 5 hay un isomorfismo $\gamma : \langle \bar{a}, \bar{a}' \rangle^{\mathbf{A}} \rightarrow \langle \bar{b}, \bar{b}' \rangle^{\mathbf{B}}$ tal que $\gamma(\bar{a}, \bar{a}') = (\bar{b}, \bar{b}')$, y como $\langle \bar{a}, \bar{a}' \rangle^{\mathbf{A}} = \mathbf{A}$ y $\langle \bar{b}, \bar{b}' \rangle^{\mathbf{B}} \leq \mathbf{B}$, γ es un embedding de \mathbf{A} en \mathbf{B} tal que $\gamma(\bar{a}) = \bar{b}$.

Supongamos que hay un embedding $\gamma : \mathbf{A} \rightarrow \mathbf{B}$ tal que $\gamma(\bar{a}) = \bar{b}$, por lo tanto $\gamma : \mathbf{A} \rightarrow \mathbf{S} \leq \mathbf{B}$ es un isomorfismo. Como $\mathbf{A} \models \varphi[\bar{a}]$, por ser γ un isomorfismo $\mathbf{S} \models \varphi[\gamma(\bar{a})]$ y entonces $\mathbf{S} \models \varphi[\bar{b}]$. Por lo tanto existe \bar{b}' en $S \subseteq B$ tal que $\mathbf{S} \models \bigwedge_{\alpha \in \Delta_{(\bar{a}, \bar{a}'), \mathbf{A}}} \alpha(\bar{x}, \bar{y}) [\bar{b}, \bar{b}']$. Como $\bigwedge_{\alpha \in \Delta_{(\bar{a}, \bar{a}'), \mathbf{A}}} \alpha(\bar{x}, \bar{y})$ es una fórmula abierta, por Teorema 2, $\mathbf{B} \models \bigwedge_{\alpha \in \Delta_{(\bar{a}, \bar{a}'), \mathbf{A}}} \alpha(\bar{x}, \bar{y}) [\bar{b}, \bar{b}']$, entonces $\mathbf{B} \models \varphi[\bar{b}]$ \square

TEOREMA 21. *Sea \mathbf{A} una estructura finita y $a_1, \dots, a_n \in A$. Entonces hay una fórmula existencial positiva $\varphi(x_1, \dots, x_n)$ tal que para toda estructura \mathbf{B} y para todo $\bar{b} \in B^n$ los siguientes son equivalentes:*

1. $\mathbf{B} \models \varphi[\bar{b}]$.
2. Hay un homomorfismo $h : \mathbf{A} \rightarrow \mathbf{B}$ tal que $h(\bar{a}) = \bar{b}$

DEMOSTRACIÓN. Sean a'_1, \dots, a'_m tales que $\langle a_1, \dots, a_n, a'_1, \dots, a'_m \rangle^{\mathbf{A}} = \mathbf{A}$ y sea

$$\varphi = \exists y_1, \dots, y_m \left(\bigwedge_{\alpha \in \Delta_{(\bar{a}, \bar{a}')}^+, \mathbf{A}} \alpha(\bar{x}, \bar{y}) \right)$$

que es fórmula por el Teorema 7.

Supongamos $\mathbf{B} \models \varphi[\bar{b}]$, entonces existen $b'_1, \dots, b'_m \in B$ tales que $\mathbf{B} \models \bigwedge_{\alpha \in \Delta_{(\bar{a}, \bar{a}')}^+, \mathbf{A}} \alpha(\bar{x}, \bar{y})[\bar{b}, \bar{b}']$

por Teorema 14 hay un homomorfismo $h : \langle \bar{a}, \bar{a}' \rangle^{\mathbf{A}} \rightarrow \langle \bar{b}, \bar{b}' \rangle^{\mathbf{B}}$ tal que $h(\bar{a}, \bar{a}') = (\bar{b}, \bar{b}')$, y como $\langle \bar{a}, \bar{a}' \rangle^{\mathbf{A}} = \mathbf{A}$ y $\langle \bar{b}, \bar{b}' \rangle^{\mathbf{B}} \leq \mathbf{B}$, h es un homomorfismo de \mathbf{A} en \mathbf{B} tal que $h(\bar{a}) = \bar{b}$.

Supongamos que hay un homomorfismo $h : \mathbf{A} \rightarrow \mathbf{B}$ tal que $h(\bar{a}) = \bar{b}$. Como $\mathbf{A} \models \varphi[\bar{a}]$, entonces $\mathbf{A} \models \Delta_{(\bar{a}, \bar{a}')}^+, \mathbf{A}[\bar{a}, \bar{a}']$ y por Teorema 12 $\mathbf{B} \models \Delta_{(\bar{a}, \bar{a}')}^+, \mathbf{A}[h(\bar{a}), h(\bar{a}')]$. Entonces $\mathbf{B} \models \Delta_{(\bar{a}, \bar{a}')}^+, \mathbf{A}[\bar{b}, \bar{b}']$, por lo que claramente $\mathbf{B} \models \varphi[\bar{b}]$. \square

TEOREMA 22. Sean $\mathcal{L} \subseteq \mathcal{L}'$ lenguajes de primer orden, sea $R \in \mathcal{L}' - \mathcal{L}$ un símbolo de relación n -ario. Para una clase finita \mathcal{K} de \mathcal{L}' -estructuras, los siguientes son equivalentes:

1. Hay una \mathcal{L} -fórmula que define R en \mathcal{K} .
2. Para todas $\mathbf{A}, \mathbf{B} \in \mathcal{K}$, todo isomorfismo $\gamma : \mathbf{A}_{\mathcal{L}} \rightarrow \mathbf{B}_{\mathcal{L}}$ preserva R .

DEMOSTRACIÓN. $1 \Rightarrow 2$) Sea φ una \mathcal{L} -fórmula que define a R en \mathcal{K} , y sea $\gamma : \mathbf{A} \rightarrow \mathbf{B}$ un isomorfismo. Supongamos $\mathbf{A} \models \varphi[\bar{a}]$ entonces $\mathbf{B} \models \varphi[\gamma(\bar{a})]$ y como φ define a R en \mathcal{K} , entonces $\gamma(\bar{a}) \in R^{\mathbf{B}}$.

$2 \Rightarrow 1$) Para cada $\mathbf{A} \in \mathcal{K}$, cada $\bar{a} \in R^{\mathbf{A}}$, tomo \bar{a}' tal que $\langle \bar{a}, \bar{a}' \rangle^{\mathbf{A}} = \mathbf{A}$. Ahora tomo $\delta_{\mathbf{A}, (\bar{a}, \bar{a}')}(\bar{x}, \bar{y}) = \bigwedge_{\alpha \in \Delta_{(\bar{a}, \bar{a}')}^+, \mathbf{A}} \alpha(\bar{x}, \bar{y})$, que es fórmula por Teorema 7. Ahora tomo

$$\varphi = \bigvee_{\mathbf{A} \in \mathcal{K}} \left(\left(\bigvee_{\bar{a} \in R^{\mathbf{A}}} (\exists \bar{y} \delta_{\mathbf{A}, (\bar{a}, \bar{a}')}(\bar{x}, \bar{y})) \right) \wedge \left(\forall x_1, \dots, x_{|\mathbf{A}|+1} \bigvee_{i \neq j} x_i = x_j \right) \right)$$

Sea $(b_1, \dots, b_n) \in R^{\mathbf{B}}$. Entonces como $\varphi = \dots \vee \left((\exists \bar{y} \delta_{\mathbf{B}, (\bar{b}, \bar{b}')}(\bar{x}, \bar{y}) \vee \dots) \wedge (\forall x_1, \dots, x_{|\mathbf{B}|+1} \bigvee_{i \neq j} x_i = x_j) \right) \dots$, y como $\mathbf{B} \models \exists \bar{y} \delta_{\mathbf{B}, (\bar{b}, \bar{b}')}(\bar{x}, \bar{y})[\bar{b}, \bar{b}']$ y $\mathbf{B} \models \forall x_1, \dots, x_{|\mathbf{B}|+1} \bigvee_{i \neq j} x_i = x_j[\bar{b}, \bar{b}']$ entonces $\mathbf{B} \models \varphi[\bar{b}]$.

Sea $\mathbf{B} \models \varphi[b_1, \dots, b_n]$. Entonces hay algún $\mathbf{A} \in \mathcal{K}$, y algún $\bar{a} \in R^{\mathbf{A}}$ tales que $\mathbf{B} \models (\exists \bar{y} \delta_{\mathbf{A}, (\bar{a}, \bar{a}')}(\bar{x}, \bar{y})) \wedge (\forall x_1, \dots, x_{|\mathbf{A}|+1} \bigvee_{i \neq j} x_i = x_j)[\bar{b}]$. Entonces hay un \bar{z} en \mathbf{B} tal que $\mathbf{B} \models \delta_{\mathbf{A}, (\bar{a}, \bar{a}')}(\bar{x}, \bar{y})[\bar{b}, \bar{z}]$. Como $\mathbf{B} \models \Delta_{(\bar{a}, \bar{a}')}^+, \mathbf{A}[\bar{b}, \bar{z}]$, por Teorema 5 hay $\gamma : \langle \bar{a}, \bar{a}' \rangle^{\mathbf{A}} \rightarrow \langle \bar{b}, \bar{b}' \rangle^{\mathbf{B}}$ isomorfismo tal que $(\gamma(\bar{a}), \gamma(\bar{a}')) = (\bar{b}, \bar{z})$ con $\langle \bar{a}, \bar{a}' \rangle^{\mathbf{A}} = \mathbf{A}$ y $\langle \bar{b}, \bar{b}' \rangle^{\mathbf{B}} \leq \mathbf{B}$, lo que implica que $|\mathbf{A}| \leq |\mathbf{B}|$. Pero como $\mathbf{B} \models \forall x_1, \dots, x_{|\mathbf{A}|+1} \bigvee_{i \neq j} x_i = x_j[b_1, \dots, b_n]$ entonces $|\mathbf{B}| \leq |\mathbf{A}|$, luego $|\mathbf{B}| = |\mathbf{A}|$, por lo que es claro que $\langle \bar{b}, \bar{b}' \rangle^{\mathbf{B}} = \mathbf{B}$. Como $\gamma : \mathbf{A} \rightarrow \mathbf{B}$ es un isomorfismo tal que $(\gamma(\bar{a}), \gamma(\bar{a}')) = (\bar{b}, \bar{z})$ y sabemos que los isomorfismos entre estructuras de \mathcal{K} , preservan R , como $\bar{a} \in R^{\mathbf{A}}$, $\gamma(\bar{a}) = \bar{b} \in R^{\mathbf{B}}$.

Como φ es disyunción de existenciales, por teorema puede ser convertida en una fórmula existencial. \square

TEOREMA 23. Sean $\mathcal{L} \subseteq \mathcal{L}'$ lenguajes de primer orden, sea $R \in \mathcal{L}' - \mathcal{L}$ un símbolo de relación n -ario. Para una clase finita \mathcal{K} de \mathcal{L}' -estructuras, los siguientes son equivalentes:

1. Hay una \mathcal{L} -fórmula existencial que define R en \mathcal{K} .
2. Para todas $\mathbf{A}, \mathbf{B} \in \mathcal{K}$, todo embedding $\gamma : \mathbf{A}_{\mathcal{L}} \rightarrow \mathbf{B}_{\mathcal{L}}$ preserva R .

DEMOSTRACIÓN. $1 \Rightarrow 2$) Sea φ existencial que define a R en \mathcal{K} , y sea $\gamma : \mathbf{A} \rightarrow \mathbf{B}$ un embedding. Entonces $\gamma : \mathbf{A} \rightarrow \mathbf{S} \leq \mathbf{B}$ es un isomorfismo. Supongamos $\mathbf{A} \models \varphi[\bar{a}]$ entonces $\mathbf{S} \models \varphi[\gamma(\bar{a})]$. Como φ es de la forma $\varphi = \exists \bar{y} \psi(\bar{x}, \bar{y})$ con ψ abierta, existe un \bar{s} tal que $\mathbf{S} \models \psi[\gamma(\bar{a}), \bar{s}]$. Como ψ es abierta y $\mathbf{S} \leq \mathbf{B}$, por el Teorema 2 $\mathbf{B} \models \psi[\gamma(\bar{a}), \bar{s}]$. Entonces $\mathbf{B} \models \varphi[\gamma(\bar{a})]$ y como φ define a R en \mathcal{K} , entonces $\gamma(\bar{a}) \in R^{\mathbf{B}}$.

$2 \Rightarrow 1$) Para cada $\mathbf{A} \in \mathcal{K}$, cada $\bar{a} \in R^{\mathbf{A}}$, tomo \bar{a}' tal que $\langle \bar{a}, \bar{a}' \rangle^{\mathbf{A}} = \mathbf{A}$. Ahora tomo $\delta_{\mathbf{A}, (\bar{a}, \bar{a}')}(\bar{x}, \bar{y}) = \bigwedge_{\alpha \in \Delta_{(\bar{a}, \bar{a}'), \mathbf{A}}} \alpha(\bar{x}, \bar{y})$, que es fórmula por Teorema 7. Ahora tomo

$$\varphi = \bigvee_{\mathbf{A} \in \mathcal{K}} \left(\bigvee_{\bar{a} \in R^{\mathbf{A}}} (\exists \bar{y} \delta_{\mathbf{A}, (\bar{a}, \bar{a}')}(\bar{x}, \bar{y})) \right)$$

Sea $(b_1, \dots, b_n) \in R^{\mathbf{B}}$. Entonces como $\varphi = \dots \vee \exists \bar{y} \delta_{\mathbf{B}, (\bar{b}, \bar{b}')}(\bar{x}, \bar{y}) \vee \dots$, y como $\mathbf{B} \models \exists \bar{y} \delta_{\mathbf{B}, (\bar{b}, \bar{b}')}(\bar{x}, \bar{y}) [\bar{b}, \bar{b}']$, $\mathbf{B} \models \varphi[\bar{b}]$.

Sea $\mathbf{B} \models \varphi[b_1, \dots, b_n]$. Entonces para algún $\mathbf{A} \in \mathcal{K}$, y algún $\bar{a} \in R^{\mathbf{A}}$ tales que $\mathbf{B} \models \exists \bar{y} \delta_{\mathbf{A}, (\bar{a}, \bar{a}')}(\bar{x}, \bar{y}) [\bar{b}]$. Entonces hay un \bar{z} en \mathbf{B} tal que $\mathbf{B} \models \delta_{\mathbf{A}, (\bar{a}, \bar{a}')}(\bar{x}, \bar{y}) [\bar{b}, \bar{z}]$. Como $\mathbf{B} \models \Delta_{(\bar{a}, \bar{a}'), \mathbf{A}}[\bar{b}, \bar{z}]$, por Teorema 5 hay $\gamma : \langle \bar{a}, \bar{a}' \rangle^{\mathbf{A}} \rightarrow \langle \bar{b}, \bar{b}' \rangle^{\mathbf{B}}$ isomorfismo tal que $(\gamma(\bar{a}), \gamma(\bar{a}')) = (\bar{b}, \bar{z})$ y como $\langle \bar{a}, \bar{a}' \rangle^{\mathbf{A}} = \mathbf{A}$ y $\langle \bar{b}, \bar{b}' \rangle^{\mathbf{B}} \leq \mathbf{B}$, γ es un embedding de \mathbf{A} en \mathbf{B} y como $\bar{a} \in R^{\mathbf{A}}$, $\gamma(\bar{a}) \in R^{\mathbf{B}}$, $\bar{b} \in R^{\mathbf{B}}$.

Como φ es disyunción de existenciales, por teorema puede ser convertida en una fórmula existencial. \square

TEOREMA 24. Sean $\mathcal{L} \subseteq \mathcal{L}'$ lenguajes de primer orden, sea $R \in \mathcal{L}' - \mathcal{L}$ un símbolo de relación n -ario. Para una clase finita \mathcal{K} de \mathcal{L}' -estructuras, los siguientes son equivalentes:

1. Hay una \mathcal{L} -fórmula existencial positiva que define R en \mathcal{K} .
2. Para todas $\mathbf{A}, \mathbf{B} \in \mathcal{K}$, todo homomorfismo $\gamma : \mathbf{A}_{\mathcal{L}} \rightarrow \mathbf{B}_{\mathcal{L}}$ preserva R .

DEMOSTRACIÓN. Igual a la prueba del Teorema 23, tomando el diagrama abierto positivo. \square

TEOREMA 25. Sean $\mathcal{L} \subseteq \mathcal{L}'$ lenguajes de primer orden, sea $R \in \mathcal{L}' - \mathcal{L}$ un símbolo de relación n -ario. Para una clase finita \mathcal{K} de \mathcal{L}' -estructuras, los siguientes son equivalentes:

1. Para cada $m \geq 1$, para todas $\mathbf{A}_1, \dots, \mathbf{A}_m, \mathbf{B} \in \mathcal{K}$, todo homomorfismo $h : (\mathbf{A}_1 \times \dots \times \mathbf{A}_m)_{\mathcal{L}} \rightarrow \mathbf{B}_{\mathcal{L}}$ preserva R .
2. Hay una \mathcal{L} -fórmula primitiva positiva que define R en \mathcal{K} .

DEMOSTRACIÓN. $2 \Rightarrow 1$) Sea $\varphi = \exists \bar{y} \psi(\bar{x}, \bar{y})$ primitiva positiva que define a R en \mathcal{K} . Sean $m \geq 1$, $\mathbf{A}_1, \dots, \mathbf{A}_m, \mathbf{B} \in \mathcal{K}$ y $h : (\mathbf{A}_1 \times \dots \times \mathbf{A}_m)_{\mathcal{L}} \rightarrow \mathbf{B}_{\mathcal{L}}$ un homomorfismo. Sean

$$\bar{a}_1 = (a_{11}, \dots, a_{1m}), \dots, \bar{a}_n = (a_{n1}, \dots, a_{nm}) \in S$$

tales que

$$(a_{1i}, \dots, a_{ni}) \in R^{A_i} \text{ para cada } i \in \{1, \dots, m\}.$$

Veamos que $(h(\bar{a}_1), \dots, h(\bar{a}_n)) \in R^{\mathbf{B}}$. Como $(a_{1i}, \dots, a_{ni}) \in R^{A_i}$ y φ define a R en \mathcal{K} , tenemos que

$$\mathbf{A}_i \models \varphi[a_1, \dots, a_{ni}] \text{ para } i \in \{1, \dots, m\}.$$

Luego, por el Lema 18,

$$\mathbf{A}_1 \times \dots \times \mathbf{A}_m \models \varphi[\bar{a}_1, \dots, \bar{a}_n].$$

Entonces hay $b_1, \dots, b_k \in A_1 \times \dots \times A_m$, tales que

$$\mathbf{A}_1 \times \dots \times \mathbf{A}_m \models \psi[\bar{a}_1, \dots, \bar{a}_n, b_1, \dots, b_k].$$

Como ψ es abierta positiva, aplicando el Teorema 12 obtenemos que $\mathbf{B} \models \varphi[h(\bar{a}_1), \dots, h(\bar{a}_n)]$. Finalmente, como φ define a R en \mathcal{K} y $\mathbf{B} \in \mathcal{K}$, concluimos que $(h(\bar{a}_1), \dots, h(\bar{a}_n)) \in R^{\mathbf{B}}$.

1 \Rightarrow 2) Supongamos $\mathcal{K} = \{\mathbf{A}_1, \dots, \mathbf{A}_m\}$, R n -aria. Sean $\bar{x}_1, \dots, \bar{x}_n \in \mathbf{A}_1^{|R^{\mathbf{A}_1}|} \times \dots \times \mathbf{A}_m^{|R^{\mathbf{A}_m}|}$ tales que $\bar{x}_i = \left(x_{i1}^1, \dots, x_{i|R^{\mathbf{A}_1}|}^1, \dots, x_{i1}^m, \dots, x_{i|R^{\mathbf{A}_m}|}^m \right)$ para cada $i \in \{1, \dots, m\}$ con $\bigcup R^{\mathbf{A}_j} = \left\{ \left(x_{1i}^j, \dots, x_{ni}^j \right) \text{ con } i \in \{1, |R^{\mathbf{A}_j}|\} \right\}$ para cada $i \in \{1, \dots, m\}$. Sean $\bar{x}'_1, \dots, \bar{x}'_k \in \mathbf{A}_1^{|R^{\mathbf{A}_1}|} \times \dots \times \mathbf{A}_m^{|R^{\mathbf{A}_m}|}$, tales que $\langle \bar{x}_1, \dots, \bar{x}_n, \bar{x}'_1, \dots, \bar{x}'_k \rangle_{\mathbf{A}_1^{|R^{\mathbf{A}_1}|} \times \dots \times \mathbf{A}_m^{|R^{\mathbf{A}_m}|}} = \mathbf{A}_1^{|R^{\mathbf{A}_1}|} \times \dots \times \mathbf{A}_m^{|R^{\mathbf{A}_m}|}$. Sea

$$\varphi = \exists \bar{y} \bigwedge_{\substack{\alpha \in \Delta^+ \\ \mathbf{A}_1^{|R^{\mathbf{A}_1}|} \times \dots \times \mathbf{A}_m^{|R^{\mathbf{A}_m}|}, (\bar{x}_1, \dots, \bar{x}_n, \bar{x}'_1, \dots, \bar{x}'_k)}} \alpha(\bar{x}, \bar{y})$$

. Supongamos $\mathbf{B} \models \varphi[b_1, \dots, b_n]$ y veamos que $(b_1, \dots, b_n) \in R^{\mathbf{B}}$. Entonces hay b'_1, \dots, b'_k tal que $\mathbf{B} \models \Delta^+_{\mathbf{A}_1^{|R^{\mathbf{A}_1}|} \times \dots \times \mathbf{A}_m^{|R^{\mathbf{A}_m}|}, (\bar{x}_1, \dots, \bar{x}_n, \bar{x}'_1, \dots, \bar{x}'_k)} [b_1, \dots, b_n, b'_1, \dots, b'_k]$,

por Teorema 14 hay h homomorfismo de $\langle \bar{x}_1, \dots, \bar{x}_n, \bar{x}'_1, \dots, \bar{x}'_k \rangle_{\mathbf{A}_1^{|R^{\mathbf{A}_1}|} \times \dots \times \mathbf{A}_m^{|R^{\mathbf{A}_m}|}}$ en $\langle b_1, \dots, b_n, b'_1, \dots, b'_k \rangle^{\mathbf{B}}$ tal que $h(\bar{x}_i) = b_i$ y $h(\bar{x}'_i) = b'_i$, claramente h es un homomorfismo de $\mathbf{A}_1^{|R^{\mathbf{A}_1}|} \times \dots \times \mathbf{A}_m^{|R^{\mathbf{A}_m}|}$ en \mathbf{B} . Como por hipótesis h preserva R , entonces como $(x_{1i}^j, \dots, x_{ni}^j) \in R^{\mathbf{A}_j}$, $(h(\bar{x}_1), \dots, h(\bar{x}_n)) \in R^{\mathbf{B}}$, que es exactamente $(b_1, \dots, b_n) \in R^{\mathbf{B}}$.

Ahora supongamos $(b_1, \dots, b_n) \in R^{\mathbf{B}}$ y veamos que $\mathbf{B} \models \varphi[b_1, \dots, b_n]$, entonces hay algún j tal que $\mathbf{A}_j = \mathbf{B}$ y hay algún k y algún j tal que

$$(x_{1k}^j, \dots, x_{nk}^j) = (b_1, \dots, b_n)$$

. Entonces como $\mathbf{A}_1^{|R^{\mathbf{A}_1}|} \times \dots \times \mathbf{B}^{|R^{\mathbf{B}|}} \times \dots \times \mathbf{A}_m^{|R^{\mathbf{A}_m}|} \models \varphi[\bar{x}_1, \dots, \bar{x}_n]$ por Teorema 18 $\mathbf{B} \models \varphi[x_{1k}^j, \dots, x_{nk}^j]$, por lo tanto $\mathbf{B} \models \varphi[b_1, \dots, b_n]$. \square

Algoritmos de chequeo de definibilidad

En este capítulo describimos algoritmos para el chequeo de definibilidad de relaciones en conjuntos finitos de estructuras finitas por fragmentos de primer orden.

3.1. Definiciones e ideas preliminares

Para evitar las complicaciones resultantes de que tanto una estructura \mathbf{A} y una subestructura propia \mathbf{A}_0 pertenezcan al conjunto \mathcal{K} , definiremos el concepto de clase *disjunta*. En general asumiremos que nuestra clase \mathcal{K} es siempre disjunta. Notar que no hay pérdida de generalidad ya que bastaría con renombrar elementos para convertir una clase no disjunta en una disjunta equivalente. Una clase \mathcal{K} de estructuras será *disjunta* si dadas $\mathbf{A}, \mathbf{B} \in \mathcal{K}$ se tiene que $A \cap B \neq \emptyset$ si y solo si $\mathbf{A} = \mathbf{B}$. La clase \mathcal{K} será *normal* si es disjunta, finita y cada una de las estructuras en \mathcal{K} es finita.

A la hora de reducir la complejidad de nuestros algoritmos en la búsqueda de homomorfismos utilizaremos conjuntos reducidos de flechas, que *generan* el resto. Dados \mathcal{F}_0 y \mathcal{F} dos conjuntos de funciones, diremos que \mathcal{F}_0 *genera* a \mathcal{F} si $\mathcal{F}_0 \subseteq \mathcal{F}$ y para toda $f \in \mathcal{F}$ hay $f_1, \dots, f_n \in \mathcal{F}_0$ (quizás repetidas) tales que $f = f_1 \circ \dots \circ f_n$.

Dado un conjunto \mathcal{K} de estructuras, para referirnos a las flechas nombradas en los Teoremas 22, 11, 24, 16 definimos los siguientes conjuntos:

$$\text{iso}(\mathcal{K}) = \{\gamma : \gamma \text{ isomorfismo de } \mathbf{A} \text{ en } \mathbf{B} \text{ con } \mathbf{A}, \mathbf{B} \in \mathcal{K}\}$$

$$\text{sub iso}(\mathcal{K}) = \{\gamma : \gamma \text{ isomorfismo de } \mathbf{A}_0 \text{ en } \mathbf{B}_0 \text{ con } \mathbf{A}_0, \mathbf{B}_0 \in \mathcal{S}(\mathcal{K})\}$$

$$\text{hom}(\mathcal{K}) = \{\gamma : \gamma \text{ homomorfismo de } \mathbf{A} \text{ en } \mathbf{B} \text{ con } \mathbf{A}, \mathbf{B} \in \mathcal{K}\}$$

$$\text{sub hom}(\mathcal{K}) = \{\gamma : \gamma \text{ homomorfismo de } \mathbf{A}_0 \text{ en } \mathbf{B}_0 \text{ con } \mathbf{A}_0, \mathbf{B}_0 \in \mathcal{S}(\mathcal{K})\}$$

3.2. Preprocesamiento

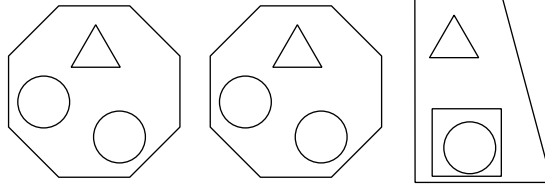
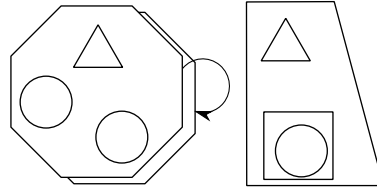
Una vez que tenemos una clase normal \mathcal{K} en la que chequear definibilidad tratamos de reducir su redundancia basandonos en el siguiente resultado:

LEMA 26. Sean $\mathcal{L} \subseteq \mathcal{L}'$ lenguajes de primer orden, \mathcal{K} una clase normal de \mathcal{L}' -estructuras y $\mathbf{A}, \tilde{\mathbf{A}} \in \mathcal{K}$, con $\mathbf{A} \neq \tilde{\mathbf{A}}$, tales que hay un \mathcal{L} -isomorfismo $\gamma : \mathbf{A} \rightarrow \tilde{\mathbf{A}}$. Entonces para toda $R \in \mathcal{L}' - \mathcal{L}$ y toda \mathcal{L} -fórmula φ son equivalentes:

1. φ define a R en \mathcal{K} .
2. γ es un $\mathcal{L} \cup \{R\}$ -isomorfismo y φ define a R en $\mathcal{K} - \{\tilde{\mathbf{A}}\}$.

DEMOSTRACIÓN. 1 \Rightarrow 2) Trivial.

2 \Rightarrow 1) $\tilde{\mathbf{A}} \models R[\bar{a}]$ sii $\mathbf{A} \models R[\gamma^{-1}(\bar{a})]$ sii $\mathbf{A} \models \varphi[\gamma^{-1}(\bar{a})]$ sii $\tilde{\mathbf{A}} \models \varphi[\bar{a}]$. \square

FIGURA 3.2.1. \mathcal{K} antes del preprocesamientoFIGURA 3.2.2. \mathcal{K} luego del preprocesamiento

Independientemente del tipo de definibilidad que uno quiera chequear el Lema 26 sugiere que uno puede comenzar por reducir \mathcal{K} eliminando copias de estructuras $\mathcal{L} \cup \{R\}$ -isomorfas. El Algoritmo 1 aprovecha este resultado aprovechando además para chequear por \mathcal{L} -isomorfismos y al descubrir uno de estos comprobar si es un $\mathcal{L} \cup \{R\}$ -isomorfismo, lo que permite encontrar contraejemplos de definibilidad desde esta etapa.

Algorithm 1 Preprocesamiento

```

1: for  $\mathbf{A} \in \mathcal{K}$  do
2:   for  $\mathbf{B} \in \mathcal{K} - \{\mathbf{A}\}$  do
3:     if hay  $\gamma : \mathbf{A} \rightarrow \mathbf{B}$   $\mathcal{L}$ -isomorfismo then
4:       if  $\gamma$  es un  $\mathcal{L} \cup \{R\}$ -isomorfismo then
5:          $\mathcal{K} = \mathcal{K} - \{\mathbf{B}\}$ 
6:       else
7:         return  $\gamma$  ▷  $R$  no es definible y  $\gamma$  es contraejemplo
8:       end if
9:     end if
10:  end for
11: end for
12: return  $\mathcal{K}$  ▷  $\mathcal{K}$  sin estructuras  $\mathcal{L}$ -isomorfas

```

Notar que los isomorfismos revisados por este algoritmo deben ser necesariamente revisados para comprobar definibilidad en cualquiera de los formatos, con la ventaja de que al chequear estos en primer lugar, podría reducirse la clase \mathcal{K} en el proceso.

Observar además que, para poder aplicar los teoremas de definibilidad de las secciones 2.1 y 2.2, habría que verificar que cada uno de los \mathcal{L} -isomorfismos entre estructuras de \mathcal{K} preserven R y vemos que basta con chequear sólo uno gracias al Lema 26.

Para ejemplificar el funcionamiento del algoritmo, supongamos que \mathcal{K} está dado por la Figura 3.2.1, donde cada tipo de isomorfismo está representado por una forma geométrica.

El algoritmo de preprocesamiento detecta el \mathcal{L} -isomorfismo entre los dos octágonos y revisa que sea también un $\mathcal{L} \cup \{R\}$ -isomorfismo, dando lugar a la Figura 3.2.2, donde la flecha es el isomorfismo a chequear.

3.3. Definibilidad abierta

A partir del Teorema 11, habría que chequear todo isomorfismo entre subestructuras de estructuras de \mathcal{K} , como en el siguiente lema.

LEMA 27. *Sean $\mathcal{L} \subseteq \mathcal{L}'$ lenguajes de primer orden tales que $R \in \mathcal{L}' - \mathcal{L}$ es un símbolo de relación n -ario y \mathcal{K} un conjunto normal de \mathcal{L}' -estructuras. Entonces si cada $\gamma \in \text{sub iso}(\mathcal{K}_{\mathcal{L}})$ preserva R , R es definible por una \mathcal{L} -fórmula abierta en \mathcal{K} .*

DEMOSTRACIÓN. Directo del Teorema 11.

En el siguiente teorema describimos un conjunto suficiente de flechas para generar $\text{sub iso}(\mathcal{K}_{\mathcal{L}})$. Notar que al tomar \mathcal{K} como un conjunto normal de \mathcal{L} -estructuras sin estructuras isomorfas estamos pensando en la aplicación previa del Algoritmo 1, basado en el Lema 26. \square

TEOREMA 28. *Sea \mathcal{K} un conjunto normal de \mathcal{L} -estructuras donde no hay estructuras isomorfas. Sea $\mathcal{S} \subseteq \mathcal{S}(\mathcal{K})$ tal que contiene exactamente un representante por cada tipo de isomorfismo en $\mathcal{S}(\mathcal{K})$. Sea $\mathcal{F} \subseteq \text{sub iso}(\mathcal{K})$ tal que:*

1. *para cada $\mathbf{S} \in \mathcal{S}$ se tiene que $\text{aut}(\mathbf{S}) \subseteq \mathcal{F}$,*
2. *para cada $\mathbf{A} \in \mathcal{S}(\mathcal{K}) - \mathcal{S}$ hay $\gamma, \gamma^{-1} \in \mathcal{F}$ para algún isomorfismo γ de \mathbf{A} en el representante del tipo de isomorfismo de \mathbf{A} en \mathcal{S} .*

Entonces \mathcal{F} genera $\text{sub iso}(\mathcal{K})$.

DEMOSTRACIÓN. Sea $\gamma \in \text{sub iso}(\mathcal{K})$, entonces $\gamma : \mathbf{A}_0 \leq \mathbf{A} \rightarrow \mathbf{B}_0 \leq \mathbf{B}$ isomorfismo, con $\mathbf{A}, \mathbf{B} \in \mathcal{K}$.

Si $\mathbf{A}_0, \mathbf{B}_0 \in \mathcal{S}$, como no hay estructuras isomorfas en \mathcal{S} , $\mathbf{A}_0 = \mathbf{B}_0$, por lo tanto $\gamma \in \text{aut}(\mathbf{A}_0) \subseteq \mathcal{F}$.

Si $\mathbf{A}_0 \in \mathcal{S}$ pero $\mathbf{B}_0 \notin \mathcal{S}$, entonces \mathbf{A}_0 es el representante de \mathbf{B}_0 en \mathcal{S} , y por lo tanto hay un isomorfismo $\delta : \mathbf{B}_0 \rightarrow \mathbf{A}_0$ tal que $\delta, \delta^{-1} \in \mathcal{F}$. Claramente $\delta\gamma = \lambda \in \text{aut}(\mathbf{A}_0) \subseteq \mathcal{F}$, por lo tanto $\gamma = \delta^{-1}\lambda$.

Si $\mathbf{A}_0, \mathbf{B}_0 \notin \mathcal{S}$, hay $\mathbf{C}_0 \in \mathcal{S}$ que es representante de \mathbf{A}_0 y \mathbf{B}_0 . Luego hay $\delta : \mathbf{A}_0 \rightarrow \mathbf{C}_0$ y $\delta' : \mathbf{B}_0 \rightarrow \mathbf{C}_0$ tales que $\delta, \delta^{-1}, \delta', \delta'^{-1} \in \mathcal{F}$. Claramente $\delta'\gamma\delta^{-1} = \lambda \in \text{aut}(\mathbf{C}_0) \subseteq \mathcal{F}$, por lo tanto $\gamma = \delta'^{-1}\lambda\delta$. \square

COROLARIO 29. *Sean $\mathcal{L} \subseteq \mathcal{L}'$ lenguajes de primer orden tales que $R \in \mathcal{L}' - \mathcal{L}$ es un símbolo de relación n -ario y \mathcal{K} un conjunto normal de \mathcal{L}' -estructuras. Supongamos \mathcal{F} es como en el Teorema 28 para la familia $\mathcal{K}_{\mathcal{L}}$. Entonces si cada $\gamma \in \mathcal{F}$ preserva R , se tiene que R es definible por una \mathcal{L} -fórmula abierta en \mathcal{K} .*

DEMOSTRACIÓN. Directa ya que la composición de funciones que preservan R , trivialmente también preserva R . Luego aplicando el Teorema 28 y el Lema 27 queda probado. \square

El Teorema 28 sugiere un subconjunto de isomorfismos que bastan para revisar definibilidad abierta a través del Corolario 29.

En el Algoritmo 2 se construye \mathcal{S} a la vez que se van revisando los isomorfismos en \mathcal{F} . Para esto recorreremos las estructuras en \mathcal{K} de mayor a menor tamaño. Al momento de procesar una $\mathbf{A} \in \mathcal{K}$ lo primero que se hace es buscar un \mathcal{L} -isomorfismo γ entre \mathbf{A} y un miembro \mathcal{S} . Si un tal γ existe, tanto γ como su inversa son revisados (miembros de \mathcal{F} correspondientes al punto (2) del Teorema 28). Si no hay un tal γ , la estructura \mathbf{A} es agregada a \mathcal{S} y sus \mathcal{L} -automorfismos son revisados (miembros de \mathcal{F} correspondientes al punto (1) del Teorema 28). A continuación las subestructuras de \mathbf{A} son procesadas de la misma manera.

Notar que en la línea 14 no se necesita revisar si los automorfismos son $\mathcal{L} \cup \{R\}$ -automorfismos ya que, como se recorren todos su inversa también es revisada por preservación.

Además, una vez que una subestructura resulta ser $\mathcal{L} \cup \{R\}$ -isomorfa a un representante en \mathcal{S} , sabemos que todas sus subestructuras ya tienen representante en \mathcal{S} y que los isomorfismos internos preservan. Esto permite disminuir la cantidad de subestructuras a revisar, como se ve en la línea 10.

Algorithm 2 Chequeo de definibilidad abierta

```

1:  $\mathcal{S} = \emptyset$ 
2: for  $\mathbf{A} \in \mathcal{K}$ , desde la mayor a la menor cardinalidad do
3:    $Sub = [\mathbf{C} : \mathbf{C} \in \mathbb{S}_{\mathcal{L}}(\mathbf{A}), \text{ de la mayor a menor cardinalidad}]$ 
4:   for  $\mathbf{B} \in Sub$  do
5:      $(iso, ce) = \text{TIENEREPRESENTANTE}(\mathbf{B}, \mathcal{S}) \triangleright \text{hay un } \mathcal{L}\text{-iso de } \mathbf{B} \text{ en } \mathbf{S} \in \mathcal{S}$ 
6:     if  $ce \neq \text{Null}$  then
7:       return  $ce$   $\triangleright ce$  es contraejemplo
8:     end if
9:     if  $iso$  then
10:       $Sub = [\mathbf{C} : \mathbf{C} \in Sub \text{ tal que } C \not\subseteq B]$ 
11:    else
12:       $\mathcal{S} = \mathcal{S} \cup \{\mathbf{B}\}$ 
13:      for  $\alpha \in \text{Aut}_{\mathcal{L}}(\mathbf{B})$  do
14:        if  $\alpha$  no preserva  $R$  then
15:          return  $\alpha$   $\triangleright \alpha$  es contraejemplo
16:        end if
17:      end for
18:    end if
19:  end for
20: end for
21: return  $\triangleright R$  es abierta-definible

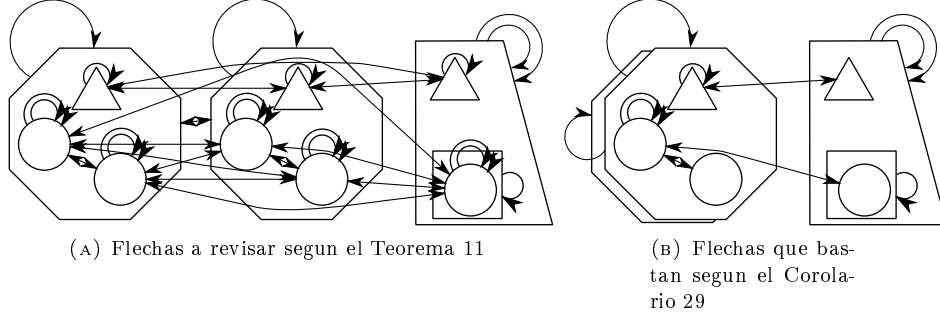
22: function TIENEREPRESENTANTE( $\mathbf{A}_0, \mathcal{S}$ )
23:   for  $\mathbf{S} \in \mathcal{S}$ , con  $|\mathbf{A}_0| = |\mathbf{S}|$  do
24:     if hay  $\gamma : \mathbf{A}_0 \rightarrow \mathbf{S}$   $\mathcal{L}$ -isomorfismo then
25:       if  $\gamma$  no es un  $\mathcal{L} \cup \{R\}$ -isomorfismo then
26:         return  $(\text{True}, \gamma)$   $\triangleright \gamma$  es contraejemplo
27:       else
28:         return  $(\text{True}, \text{Null})$   $\triangleright \gamma$  es isomorfismo con el representante
29:       end if
30:     end if
31:   end for
32:   return  $(\text{False}, \text{Null})$   $\triangleright$  No tiene representante
33: end function

```

Para esquematizar la ganancia obtenida en cuanto al chequeo de una menor cantidad de flechas podemos observar en la Figura 3.3.2a el conjunto \mathcal{K} junto con todas las flechas que deberían ser revisadas según el Teorema 11. Mientras que en la Figura 3.3.2b, se puede ver las flechas que bastan para comprobar definibilidad abierta según el Corolario 29 a la manera del algoritmo anterior.

Notar que en la Figura 3.3.2b, la aplicación del Algoritmo 1 (Preprocesamiento) genera que uno de los octágonos aparezca como *sombra* del otro y baste solo revisar el isomorfismo entre ellos por el Lema 26.

FIGURA 3.3.1



3.4. Definibilidad abierta-positiva

A partir del Teorema 16, habría que chequear todo homomorfismo entre subestructuras de estructuras de \mathcal{K} , como en el siguiente lema.

LEMA 30. Sean $\mathcal{L} \subseteq \mathcal{L}'$ lenguajes de primer orden tales que $R \in \mathcal{L}' - \mathcal{L}$ es un símbolo de relación n -ario y \mathcal{K} un conjunto normal de \mathcal{L}' -estructuras. Entonces si cada $\gamma \in \text{sub hom}(\mathcal{K}_{\mathcal{L}})$ preserva R , R es definible por una \mathcal{L} -fórmula abierta positiva en \mathcal{K} .

DEMOSTRACIÓN. Directo del Teorema 16.

En el siguiente teorema describimos un conjunto suficiente de flechas para generar $\text{sub hom}(\mathcal{K}_{\mathcal{L}})$. Notar que, nuevamente, al tomar \mathcal{K} como un conjunto normal de \mathcal{L} -estructuras sin estructuras isomorfas estamos pensando en la aplicación previa del Algoritmo 1, basado en el Lema 26. \square

TEOREMA 31. Sea \mathcal{K} un conjunto normal de \mathcal{L} -estructuras donde no hay estructuras isomorfas. Sea $\mathcal{S} \subseteq \mathbb{S}(\mathcal{K})$ tal que contiene exactamente un representante por cada tipo de isomorfismo en $\mathbb{S}(\mathcal{K})$. Sea $\mathcal{H} \subseteq \text{sub hom}(\mathcal{K})$ tal que:

- $\mathcal{F} \subseteq \mathcal{H}$ con \mathcal{F} como en el Teorema 28,
- para cada $\mathbf{A}, \mathbf{B} \in \mathcal{S}$ todo $h : \mathbf{A} \rightarrow \mathbf{B}$ homomorfismo sobreyectivo esta en \mathcal{H} .

Entonces \mathcal{H} genera $\text{sub hom}(\mathcal{K})$.

DEMOSTRACIÓN. Sea $h \in \text{sub hom}(\mathcal{K})$, entonces $h : \mathbf{A}_0 \leq \mathbf{A} \rightarrow \mathbf{B}_0 \leq \mathbf{B}$ homomorfismo, con $\mathbf{A}, \mathbf{B} \in \mathcal{K}$.

Si $\mathbf{A}_0, h(\mathbf{A}_0) \in \mathcal{S}$, h es homomorfismo sobreyectivo entre estructuras de \mathcal{S} , por lo que $h \in \mathcal{H}$.

Si $\mathbf{A}_0 \in \mathcal{S}$ pero $h(\mathbf{A}_0) \notin \mathcal{S}$, hay \mathbf{S} representante de $h(\mathbf{A}_0)$ en \mathcal{S} y un isomorfismo $\delta : h(\mathbf{A}_0) \rightarrow \mathbf{S}$ tal que $\delta, \delta^{-1} \in \mathcal{F} \subseteq \mathcal{H}$. Luego $\delta^{-1}h = f$ donde f es un homomorfismo claramente sobreyectivo de \mathbf{A}_0 en \mathbf{S} , por lo que $f \in \mathcal{H}$. Finalmente $h = \delta f$.

Si $\mathbf{A}_0, h(\mathbf{A}_0) \notin \mathcal{S}$, hay respectivos $\mathbf{S}, \mathbf{S}' \in \mathcal{S}$ representantes de \mathbf{A}_0 y $h(\mathbf{A}_0)$. Luego hay $\delta : \mathbf{A}_0 \rightarrow \mathbf{S}$ y $\delta' : h(\mathbf{A}_0) \rightarrow \mathbf{S}'$ tales que $\delta, \delta^{-1}, \delta', \delta'^{-1} \in \mathcal{F} \subseteq \mathcal{H}$. Luego $\delta' h \delta^{-1} = f$ donde f es un homomorfismo claramente sobreyectivo de \mathbf{S} en \mathbf{S}' , por lo que $f \in \mathcal{H}$. Finalmente $h = \delta'^{-1} f \delta$. \square

COROLARIO 32. Sean $\mathcal{L} \subseteq \mathcal{L}'$ lenguajes de primer orden tales que $R \in \mathcal{L}' - \mathcal{L}$ es un símbolo de relación n -ario y \mathcal{K} un conjunto normal de \mathcal{L}' -estructuras. Supongamos \mathcal{F} es como en el Teorema 31 para la familia $\mathcal{K}_{\mathcal{L}}$. Entonces si cada

$\gamma \in \mathcal{F}$ preserva R , se tiene que R es definible por una \mathcal{L} -fórmula abierta positiva en \mathcal{K} .

DEMOSTRACIÓN. Directa ya que la composición de funciones que preservan R , trivialmente también preserva R . Luego aplicando el Teorema 31 y el Lema 30 queda probado. \square

El Teorema 31 sugiere un subconjunto de homomorfismos que bastan para revisar definibilidad abierta positiva a través del Corolario 32. A su vez, a la hora de buscar homomorfismos entre estructuras aprovechamos el siguiente resultado, que nos permite organizar la búsqueda de homomorfismos biyectivos entre dos subestructuras en un único sentido.

TEOREMA 33 (Las estructuras finitas cumplen la propiedad de Cantor-Bernstein). *Dadas \mathbf{A} y \mathbf{B} estructuras finitas, si hay $f : A \rightarrow B$ y $g : B \rightarrow A$ homomorfismos inyectivos entonces hay un $\gamma : \mathbf{A} \rightarrow \mathbf{B}$ isomorfismo.*

DEMOSTRACIÓN. Sean \mathbf{A}, \mathbf{B} estructuras finitas y $f : A \rightarrow B$ y $g : B \rightarrow A$ homomorfismos inyectivos. Notar que gf es una permutación de \mathbf{A} , y como \mathbf{A} es finito hay $k \geq 1$ tal que $(gf)^k = \text{Id}_A$. Luego como $g^{-1} = f(gf)^{k-1}$ vemos que g^{-1} es homomorfismo. \square

En el Algoritmo 3 se construye \mathcal{S} a la vez que se van revisando los homomorfismos en \mathcal{F} . Para esto recorremos las estructuras en \mathcal{K} de mayor a menor tamaño. Al momento de procesar una $\mathbf{A} \in \mathcal{K}$, mientras que para definibilidad abierta lo primero que se hacía era buscar un \mathcal{L} -isomorfismo para cada $\mathbf{S} \in \mathcal{S}$, ahora se buscan homomorfismos biyectivos primero, chequeandolos por preservación. Esto da lugar a tres casos:

1. Uno de estos homomorfismos biyectivos es un \mathcal{L} -isomorfismo.
2. Hay homomorfismos biyectivos de \mathbf{A} en \mathbf{S} pero ninguno es un \mathcal{L} -isomorfismo.
3. No hay homomorfismos biyectivos de \mathbf{A} en \mathbf{S} .

En el primer caso se procede como en el Algoritmo 2. En el segundo caso podemos estar seguros de que no hay homomorfismos biyectivos de \mathbf{S} en \mathbf{A} , ya que por el Teorema 33, debería haber un \mathcal{L} -isomorfismo entre ellos. Mientras que en el tercer caso deberemos buscar homomorfismos biyectivos de \mathbf{S} en \mathbf{A} , para revisar su preservación.

Esto nos permite estar seguros de que todos los homomorfismos biyectivos entre estructuras de \mathcal{S} han sido chequeados. Finalmente lo único que falta es chequear los homomorfismos sobreyectivos pero no inyectivos entre estructuras de \mathcal{S} . Para lo cual basta con buscar homomorfismos sobreyectivos de \mathbf{A} en \mathbf{B} tales que $|A| > |B|$.

Para esquematizar la ganancia obtenida en cuanto al chequeo de una menor cantidad de flechas podemos observar en la Figura 3.4.2a el conjunto \mathcal{K} junto con todas las flechas que deberían ser revisadas según el Teorema 16, donde las flechas punteadas son los homomorfismos. Mientras que en la Figura 3.4.2b, se puede ver las flechas que bastan para comprobar definibilidad abierta según el Teorema 31 a la manera del algoritmo anterior.

3.5. Definibilidad de primer orden

Por el Teorema 22, luego de haber aplicado el Algoritmo 1 para preprocesamiento, solo basta con revisar los automorfismos para $\mathbf{A} \in \mathcal{K}$, como hacemos en el Algoritmo 4.

Algorithm 3 Chequeo de definibilidad abierta-positiva

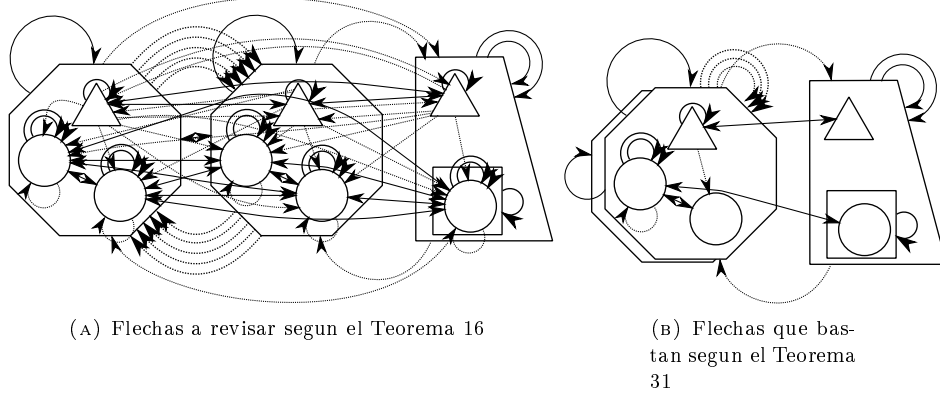
```

1:  $\mathcal{S} = \emptyset$ 
2: for  $\mathbf{A} \in \mathcal{K}$ , desde la mayor a la menor cardinalidad do
3:    $Sub = [\mathbf{C} : \mathbf{C} \in \mathbb{S}_{\mathcal{L}}(\mathbf{A}), \text{ de la mayor a menor cardinalidad}]$ 
4:   for  $\mathbf{B} \in Sub$  do
5:      $(iso, ce) = \text{TIENEREPRESENTANTE}(\mathbf{B}, \mathcal{S}) \triangleright \text{hay un } \mathcal{L}\text{-iso de } \mathbf{B} \text{ en } \mathbf{S} \in \mathcal{S}$ 
6:     if  $ce \neq \text{null}$  then
7:       return  $ce$   $\triangleright ce$  es contraejemplo
8:     end if
9:     if  $iso$  then
10:       $Sub = [\mathbf{C} : \mathbf{C} \in Sub \text{ tal que } C \not\subseteq B]$ 
11:    else
12:       $\mathcal{S} = \mathcal{S} \cup \{\mathbf{B}\}$ 
13:      for  $\alpha \in \text{Aut}_{\mathcal{L}}(\mathbf{B})$  do
14:        if  $\alpha$  no preserva  $R$  then
15:          return  $\alpha$   $\triangleright \alpha$  es contraejemplo
16:        end if
17:      end for
18:    end if
19:  end for
20: end for
21: for  $\mathbf{A} \in \mathcal{S}$  do
22:   for  $\mathbf{B} \in \mathcal{S}$ , con  $|\mathbf{B}| < |\mathbf{A}|$  do
23:    for  $\gamma : \mathbf{A} \rightarrow \mathbf{B}$  con  $\gamma$   $\mathcal{L}$ -homomorfismo sobreyectivo do
24:      if  $\gamma$  no preserva  $R$  then
25:        return  $\gamma$   $\triangleright \gamma$  es contraejemplo
26:      end if
27:    end for
28:  end for
29: end for
30: return  $\triangleright R$  es definible por una abierta positiva

31: function TIENEREPRESENTANTE( $\mathbf{A}_0, \mathcal{S}$ )
32:   for  $\mathbf{S} \in \mathcal{S}$ , con  $|\mathbf{A}_0| = |\mathbf{S}|$  do
33:     bihomos =  $[\gamma : \gamma : \mathbf{A}_0 \rightarrow \mathbf{S} \text{ con } \gamma \text{ } \mathcal{L}\text{-homomorfismo biyectivo}]$ 
34:     for  $\gamma \in \text{bihomos}$  do
35:       if  $\gamma$  es un  $\mathcal{L}$ -isomorfismo, con  $\mathbf{S} \in \mathcal{S}$  then
36:         if  $\gamma$  no es un  $\mathcal{L} \cup \{R\}$ -isomorfismo then
37:           return  $(\text{True}, \gamma)$   $\triangleright \gamma$  es contraejemplo
38:         else
39:           return  $(\text{True}, \text{Null})$   $\triangleright \gamma$  es isomorfismo con el representante
40:         end if
41:       else if  $\gamma$  no preserva  $R$  then
42:         return  $(\text{False}, \gamma)$   $\triangleright \gamma$  es contraejemplo
43:       end if
44:     end for
45:     if bihomos =  $\emptyset$  then
46:       for  $\gamma : \mathbf{S} \rightarrow \mathbf{A}_0$  con  $\gamma$   $\mathcal{L}$ -homomorfismo biyectivo do
47:        if  $\gamma$  no preserva  $R$  then
48:          return  $(\text{False}, \gamma)$   $\triangleright \gamma$  es contraejemplo
49:        end if
50:      end for
51:     end if
52:     return  $(\text{False}, \text{Null})$   $\triangleright$  No tiene representante
53:   end for
54: end function

```

FIGURA 3.4.1

**Algorithm 4** Chequeo de definibilidad de primer orden

```

1: for  $\mathbf{A} \in \mathcal{K}$  do
2:   for  $\alpha \in \text{Aut}_{\mathcal{L}}(\mathbf{A})$  do
3:     if  $\alpha$  no es un  $\mathcal{L} \cup \{R\}$ -automorfismo then
4:       return  $\alpha$   $\triangleright \alpha$  es contraejemplo
5:     end if
6:   end for
7: end for
8: return  $\triangleright R$  es definible en primer orden

```

3.6. Definibilidad existencial

Por el Teorema 23, luego de aplicar el Algoritmo 1 de preprocesamiento, solo basta con revisar los embeddings entre $\mathbf{A}, \mathbf{B} \in \mathcal{K}$, como hacemos en el Algoritmo 5.

Algorithm 5 Chequeo de definibilidad existencial

```

1: for  $\mathbf{A} \in \mathcal{K}$  do
2:   for  $\mathbf{B} \in \mathcal{K}$  do
3:     for  $\gamma \in \text{Emb}_{\mathcal{L}}(\mathbf{A}, \mathbf{B})$  do
4:       if  $\gamma$  no es un  $\mathcal{L} \cup \{R\}$ -embedding then
5:         return  $\gamma$   $\triangleright \gamma$  es contraejemplo
6:       end if
7:     end for
8:   end for
9: end for
10: return  $\triangleright R$  es existencial definible

```

3.7. Definibilidad existencial-positiva

Por el Teorema 24, luego de aplicar el Algoritmo 1 de preprocesamiento, solo basta con revisar los homomorfismos cada par $\mathbf{A}, \mathbf{B} \in \mathcal{K}$, como hacemos en el Algoritmo 6.

Algorithm 6 Chequeo de definibilidad existencial-positiva

```

1: for  $\mathbf{A} \in \mathcal{K}$  do
2:   for  $\mathbf{B} \in \mathcal{K}$  do
3:     for  $\alpha \in \text{Hom}_{\mathcal{L}}(\mathbf{A}, \mathbf{B})$  do
4:       if  $h$  no preserva  $R$  then
5:         return  $h$   $\triangleright h$  es contraejemplo
6:       end if
7:     end for
8:   end for
9: end for
10: return  $\triangleright R$  es existencial positiva definible

```

3.8. Detección de homomorfismos

Para resolver CSP utilizamos Minion.

DEMOSTRACIÓN. Sea γ un endomorfismo inyectivo de \mathbf{A} . Como A es finito tenemos que γ es sobre y además $\gamma^{-1} = \gamma^k$ para algún $k \geq 1$. \square

3.8.1. CSP. Como aparece en Wikipedia 2016, un CSP (constraint satisfaction problem) es definido como una terna $\langle X, D, C \rangle$, donde

$X = \{X_1, \dots, X_n\}$ es un conjunto de variables

$D = \{D_1, \dots, D_n\}$ es un conjunto con los respectivos dominios de los valores

$C = \{C_1, \dots, C_m\}$ es un conjunto de restricciones

Cada variable X_i se mueve en los valores del respectivo dominio no vacío D_i . Cada restricción $C_j \in C$ es un par $\langle t_j, R_j \rangle$, donde t_j es una k -upla de variables y R_j es una relación k -aria en el correspondiente dominio de cada variable. Una valuación sobre las variables es una función desde un subconjunto de las variables a un particular conjunto de valores en los correspondientes dominios de valores. Una valuación v satisface la restricción $\langle t_j, R_j \rangle$ si los valores asignados a las variables t_j satisfacen la relación R_j .

Una valuación es consistente si no viola ninguna de las restricciones. Una valuación es completa si incluye todas las variables. Una valuación es solución si es consistente y completa. En ese caso diremos que la valuación resuelve el CSP.

3.8.2. CSP para calcular homomorfismos. Sean \mathbf{A}, \mathbf{B} \mathcal{L} -estructuras, queremos encontrar homomorfismos de \mathbf{A} en \mathbf{B} resolviendo una instancia de CSP. Tomamos

$$X = \{X_i : i \in A\}$$

$$D_i = B$$

$$C = C^R \cup C^f$$

donde C^R es tal que para cada $R \in \mathcal{L}$ n -aria, para cada $(a_1, \dots, a_n) \in R^{\mathbf{A}}$, se da que $((X_{a_1}, \dots, X_{a_n}), R^{\mathbf{B}}) \in C^R$ y C^f es tal que para cada $f \in \mathcal{L}$ n -aria, para cada $(a_1, \dots, a_n, a') \in \text{graph}(f^{\mathbf{A}})$, se da que $((X_{a_1}, \dots, X_{a_n}, X_{a'}), \text{graph}(f^{\mathbf{B}})) \in C^f$.

Supongamos que $V : X \rightarrow B$ es una valuación solución de (X, D, C) , entonces el homomorfismo γ determinado por V será $\gamma(a) = V(X_a)$.

Si además quisiéramos que el homomorfismo fuera inyectivo, bastaría con agregar una restricción:

$((X_1, \dots, X_m), \text{para todo par } (i, j) \text{ con } i \neq j \text{ y } i, j \in \{1, \dots, m\} \text{ se da que } X_i \neq X_j)$

donde X_1, \dots, X_m son todas las variables en X .

Para que fuera sobreyectivo bastaría agregar la siguiente restriccion:

$$\left((X_1, \dots, X_m), \bigcup_{i=1}^m \{X_i\} = B \right)$$

En el caso particular de la búsqueda de automorfismos, el lema 34 nos permite simplificar la busqueda, encontrando endomorfismos inyectivos.

LEMA 34. *Sea \mathbf{A} una estructura finita, entonces todo endomorfismo inyectivo de \mathbf{A} es un automorfismo de \mathbf{A} .*

3.9. Generación de subestructuras

Dada una \mathcal{L} -estructura \mathbf{A} , generamos sus subestructuras recorriendo $\mathcal{P}(A)$, desde la mayor a la menor cardinalidad filtrando aquellos $A_0 \in \mathcal{P}(A)$ que no son cerrados bajo \mathcal{L} .

Además cuando detectamos que una subestructura \mathbf{A}_0 ya tiene representante en \mathcal{S} durante nuestros algoritmos, disminuimos los subconjuntos a chequear, revisando solo $\mathcal{P}(A) - \mathcal{P}(A_0)$, ya que si \mathbf{A}_0 tiene representante en \mathcal{S} , entonces cada subestructura de \mathbf{A}_0 tiene representante en \mathcal{S} .

Álgebras de Lindenbaum

4.1. Definiciones e ideas básicas

Sea \mathbf{A} una \mathcal{L} -estructura, y sea

$$\Sigma \in \{\text{Fo}(\mathcal{L}), \text{E}(\mathcal{L}), \text{E}^+(\mathcal{L}), \text{Op}(\mathcal{L}), \text{Op}^+(\mathcal{L})\}.$$

Notar que la colección de relaciones n -arias definibles en \mathbf{A} por fórmulas en Σ es cerrada bajo uniones e intersecciones (y también bajo complementación cuando $\Sigma \in \{\text{Fo}(\mathcal{L}), \text{Op}(\mathcal{L})\}$). Llamaremos *Álgebra de Lindenbaum* (de relaciones n -arias definibles por Σ en \mathbf{A}) al reticulado distributivo (o álgebra de Boole) resultante. Utilizaremos la siguiente notación:

- $\mathbf{Fo}_n(\mathbf{A})$ = Álgebra de Boole de relaciones n -arias definibles en \mathbf{A} ,
- $\mathbf{E}_n(\mathbf{A})$ = Reticulado de relaciones n -arias definibles por existenciales en \mathbf{A} ,
- $\mathbf{E}_n^+(\mathbf{A})$ = Reticulado de relaciones n -arias definibles por existenciales positivas en \mathbf{A} ,
- $\mathbf{Op}_n(\mathbf{A})$ = Álgebra de Boole de relaciones n -arias definibles por abiertas en \mathbf{A} ,
- $\mathbf{Op}_n^+(\mathbf{A})$ = Reticulado de relaciones n -arias definibles por abiertas positivas en \mathbf{A}

Definimos:

$$\text{sub hom}(\mathbf{A}) = \{\gamma : \gamma \text{ homomorfismo de } \mathbf{A}_0 \text{ en } \mathbf{B}_0 \text{ con } \mathbf{A}_0, \mathbf{B}_0 \in \mathbb{S}(\mathbf{A})\}$$

$$\text{sub iso}(\mathbf{A}) = \{\gamma : \gamma \text{ isomorfismo de } \mathbf{A}_0 \text{ en } \mathbf{B}_0 \text{ con } \mathbf{A}_0, \mathbf{B}_0 \in \mathbb{S}(\mathbf{A})\}$$

Dado que estas álgebras podrían ser muy grandes utilizamos los siguientes resultados para representarlas mediante un subconjunto de sus elementos.

TEOREMA 35 (Teorema de representación de Stone). *Toda álgebra de Boole finita \mathbf{A} es isomorfa al álgebra definida por $\mathcal{P}(At)$ donde $At \subseteq A$ es el conjunto de átomos en \mathbf{A} .*

TEOREMA 36 (Teorema de representación de Birkhoff). *Todo reticulado distributivo finito \mathbf{L} es isomorfo al reticulado de conjuntos descendientes del poset de elementos join-irreducibles en \mathbf{L} .*

LEMA 37. *Dada un álgebra de Boole finita \mathbf{A} , un elemento es un átomo sii es join-irreducible.*

Los resultados anteriores nos permiten centrarnos únicamente en los elementos join-irreducibles de estas álgebras. Mientras que los Lemas 38 y 39, presentados a continuación, sugieren una manera clara de calcularlos.

LEMA 38. *Una relación $r \subseteq A^n$ es join-irreducible en $\mathbf{Op}(\mathbf{A})$ sii hay $\bar{a} \in A^n$ tal que $r = \{h(\bar{a}) : h \in \text{sub iso}(\mathbf{A})\}$.*

DEMOSTRACIÓN. Supongamos que $r \subseteq A^n$ es join-irreducible en $\mathbf{Op}(\mathbf{A})$. Como r es cerrado bajo $\text{sub iso}(\mathbf{A})$ por pertenecer a $\mathbf{Op}(\mathbf{A})$, es claro que

$$r = \bigcup_{\bar{x} \in r} \{h(\bar{x}) : h \in \text{sub iso}(\mathbf{A})\}$$

, pero como r es join-irreducible hay una $\bar{a} \in r$ tal que $r = \{h(\bar{a}) : h \in \text{sub iso}(\mathbf{A})\}$.

Supongamos que $r = \{h(\bar{a}) : h \in \text{sub iso}(\mathbf{A})\}$ para algun $\bar{a} \in A^n$. Sean $r_1, \dots, r_m \in \mathbf{Op}(\mathbf{A})$ tales que

$$r = r_1 \cup \dots \cup r_m$$

, es claro que hay un r_j tal que $\bar{a} \in r_j$. Ademas como r_j es cerrado bajo sub iso (\mathbf{A}) por pertenecer a $\mathbf{Op}(\mathbf{A})$, entonces $r = r_j$. \square

LEMA 39. *Una relación $r \subseteq A^n$ es join-irreducible en $\mathbf{Op}^+(\mathbf{A})$ sii hay $\bar{a} \in A^n$ tal que $r = \{h(\bar{a}) : h \in \text{sub hom}(\mathbf{A})\}$.*

DEMOSTRACIÓN. Igual a la del Lema 38 \square

4.2. Algoritmos

Con la intencion de simplificar la exposicion, supondremos que \mathcal{K} tiene una unica estructura \mathbf{A} . La generalizacion es facilmente deducible ya que los algoritmos son casi iguales a los presentados en el Capitulo 3.

Como un algebra de Lindenbaum de relaciones definibles puede llegar a ser un objeto muy grande y dificil de manipular, nos basta generar los elementos join-irreducibles del algebra segun los Teoremas 35 para aquellas algebras de lindenbaum que son algebras de Boole y los atomos (que tambien son los elementos join-irreducibles segun el Lema 37) segun el Teorema 36, para aquellas que son reticulados.

Para generar los elementos join-irreducibles del algebra de Lindenbaum de relaciones de ancho n definibles por abiertas en \mathbf{A} basta con obtener el conjunto \mathcal{F} de flechas que se chequean por preservacion en los algoritmos del Capítulo 3 que genera sub iso (\mathbf{A}) por el Teorema 28, y luego calcular $\{h(\bar{a}) : h \in \text{sub iso}(\mathbf{A})\}$ para cada $\bar{a} \in A^n$, ya que por el Lema 38, todos los elementos join-irreducibles son de esta forma. De la misma manera podemos construir el algebra de Lindenbaum de relaciones de ancho n definibles por abiertas positivas en \mathbf{A} , basandonos en el Teorema 31 y en el Lema 39.

Los algoritmos 7 y 8, generan el conjunto \mathcal{F} definidos por los Teoremas 28 y 31.

Algorithm 7 Construcción de $\mathbf{Op}_n(\mathbf{A})$

```

1:  $\mathcal{S} = \emptyset$ 
2:  $\mathcal{F} = \emptyset$ 
3:  $Sub = [\mathbf{C} : \mathbf{C} \in \mathbb{S}_{\mathcal{L}}(\mathbf{A}), \text{ de la mayor a menor cardinalidad}]$ 
4: for  $\mathbf{B} \in Sub$  do
5:    $iso = \text{TIENEREPRESENTANTE}(\mathbf{A}_0, \mathcal{S})$ 
6:   if  $iso \neq \text{Null}$  then
7:      $\mathcal{F} = \mathcal{F} \cup \{iso\}$ 
8:      $Sub = [\mathbf{C} : \mathbf{C} \in Sub \text{ tal que } C \not\subseteq B]$ 
9:   else
10:     $\mathcal{S} = \mathcal{S} \cup \{\mathbf{B}\}$ 
11:    for  $\alpha \in \text{Aut}_{\mathcal{L}}(\mathbf{B})$  do
12:       $\mathcal{F} = \mathcal{F} \cup \{\alpha\}$ 
13:    end for
14:  end if
15: end for
16: return  $\mathcal{F}$  ▷  $R$  es abierta-definible

17: function  $\text{TIENEREPRESENTANTE}(\mathbf{A}_0, \mathcal{S})$ 
18:   for  $\mathbf{S} \in \mathcal{S}$ , con  $|\mathbf{A}_0| = |\mathbf{S}|$  do
19:     if hay  $\gamma : \mathbf{A}_0 \rightarrow \mathbf{S}$  isomorfismo then
20:       return  $\gamma$  ▷  $\gamma$  es isomorfismo con el representante
21:     end if
22:   end for
23:   return Null ▷ No tiene representante
24: end function

```

Una vez calculado \mathcal{F} basta con calcular $\{h(\bar{a}) : h \in \mathcal{F}\}$ para cada $\bar{a} \in A^n$, ya que \mathcal{F} genera al conjunto correspondiente al tipo de definibilidad. Esto lo hacemos con el Algoritmo 9, lo cual ya nos da los elementos join-irreducibles o los átomos, según el tipo de definibilidad en cuestión.

Algorithm 9 Saturación por un conjunto de flechas

```

1:  $\mathcal{A} = \emptyset$ 
2: for  $\bar{a} \in A^n$  do
3:    $\mathcal{A} = \mathcal{A} \cup \{\text{CLAUSURA}(\bar{a}, \mathcal{E})\}$ 
4: end for
5: return  $\mathcal{A}$  ▷  $R$  es abierta-definible

6: function  $\text{CLAUSURA}(\bar{a}, \mathcal{E})$ 
7:    $e = \{\bar{a}\}$ 
8:   while  $e$  crezca do
9:     for  $\bar{a} \in e$  do
10:      for  $\gamma \in \mathcal{E}$  do
11:         $e = e \cup \{\gamma(\bar{b})\}$ 
12:      end for
13:    end for
14:   end while
15:   return  $e$ 
16: end function

```

Algorithm 8 Construcción de $\mathbf{Op}_n^+(\mathbf{A})$

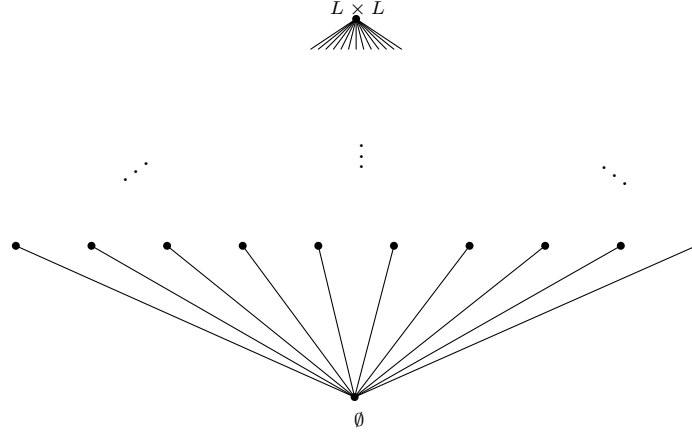
```

1:  $\mathcal{S} = \emptyset$ 
2:  $\mathcal{F} = \emptyset$ 
3:  $Sub = [\mathbf{C} : \mathbf{C} \in \mathbb{S}_{\mathcal{L}}(\mathbf{A}), \text{ de la mayor a menor cardinalidad}]$ 
4: for  $\mathbf{B} \in Sub$  do
5:    $(iso, \mathcal{F}_0) = \text{TIENEREPRESENTANTE}(\mathbf{A}_0, \mathcal{S})$ 
6:    $\mathcal{F} = \mathcal{F} \cup \mathcal{F}_0$ 
7:   if  $iso \neq \text{null}$  then
8:      $\mathcal{F} = \mathcal{F} \cup \{iso\}$ 
9:      $Sub = [\mathbf{C} : \mathbf{C} \in Sub \text{ tal que } C \not\subseteq B]$ 
10:  else
11:     $\mathcal{S} = \mathcal{S} \cup \{\mathbf{B}\}$ 
12:    for  $\alpha \in \text{Aut}_{\mathcal{L}}(\mathbf{B})$  do
13:       $\mathcal{F} = \mathcal{F} \cup \{\alpha\}$ 
14:    end for
15:  end if
16: end for
17: for  $\mathbf{A} \in \mathcal{S}$  do
18:   for  $\mathbf{B} \in \mathcal{S}$ , con  $|\mathbf{B}| < |\mathbf{A}|$  do
19:    for  $\gamma : \mathbf{A} \rightarrow \mathbf{B}$  con  $\gamma$  homomorfismo sobreyectivo do
20:       $\mathcal{F} = \mathcal{F} \cup \{\gamma\}$ 
21:    end for
22:  end for
23: end for
24: return  $\mathcal{F}$  ▷  $R$  es definible por una abierta positiva

25: function TIENEREPRESENTANTE( $\mathbf{A}_0, \mathcal{S}$ )
26:    $\mathcal{F}_0 = \emptyset$ 
27:   for  $\mathbf{S} \in \mathcal{S}$ , con  $|\mathbf{A}_0| = |\mathbf{S}|$  do
28:      $bihomo = \text{False}$ 
29:     for  $\gamma : \mathbf{A}_0 \rightarrow \mathbf{S}$  con  $\gamma$  homomorfismo biyectivo do
30:        $bihomo = \text{True}$ 
31:       if  $\gamma$  es un  $\mathcal{L}$ -isomorfismo, con  $\mathbf{S} \in \mathcal{S}$  then
32:         return  $(\gamma, \emptyset)$  ▷  $\gamma$  es isomorfismo con el representante
33:       else
34:          $\mathcal{F}_0 = \mathcal{F}_0 \cup \{\gamma\}$ 
35:       end if
36:     end for
37:     if  $\neg bihomo$  then
38:       for  $\gamma : \mathbf{S} \rightarrow \mathbf{A}_0$  con  $\gamma$  homomorfismo biyectivo do
39:          $\mathcal{F}_0 = \mathcal{F}_0 \cup \{\gamma\}$ 
40:       end for
41:     end if
42:     return  $(\text{Null}, \mathcal{F}_0)$  ▷ No tiene representante
43:   end for
44: end function

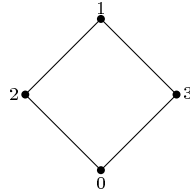
```

Para los demas tipos de definibilidad, el procedimiento seria analogo. Generar \mathcal{F} basados en los algoritmos del capitulo 3 y saturar segun el Algoritmo 9.

FIGURA 4.3.2. Reticulado $\mathbf{E}_2(\mathbf{2} \times \mathbf{2})$ 

4.3. Ejemplos y aplicación

4.3.1. Utilizando el paquete Definability para SageMath. A manera de prueba de las herramientas utilizamos Definability para recorrer todas las relaciones binarias definibles en el reticulado $\mathbf{2} \times \mathbf{2}$, representado en la figura 4.3.1.

FIGURA 4.3.1. Reticulado $\mathbf{2} \times \mathbf{2}$ 

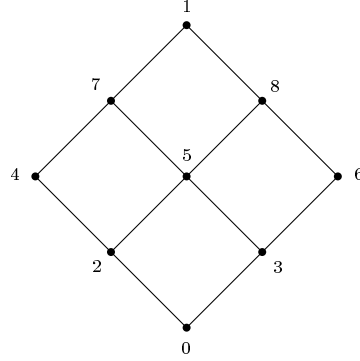
Al correr el algoritmo de generación de los elementos join-irreducibles de $\mathbf{E}_2(\mathbf{2} \times \mathbf{2})$, Definability nos devolvió las siguientes relaciones:

- $\{(0, 0)\}$
- $\{(0, 1)\}$
- $\{(0, 2), (0, 3)\}$
- $\{(1, 0)\}$
- $\{(1, 1)\}$
- $\{(1, 2), (1, 3)\}$
- $\{(2, 0), (3, 0)\}$
- $\{(2, 1), (3, 1)\}$
- $\{(2, 2), (3, 3)\}$
- $\{(2, 3), (3, 2)\}$

Aquí se puede notar las ventajas de utilizar el Teorema 36, ya que el reticulado descrito por estos 10 elementos tiene cardinalidad $2^{10} = 1024$, ya que son todas las uniones entre elementos join-irreducibles.

En cuanto a los elementos join-irreducibles de $\mathbf{E}_2^+(\mathbf{2} \times \mathbf{2})$ nos devolvió:

- $\{(0, 0), (1, 1), (2, 2), (3, 3)\}$
- $\{(0, 0), (0, 1), (0, 2), (0, 3), (1, 1), (2, 1), (2, 2), (3, 1), (3, 3)\}$

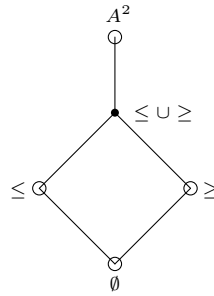
FIGURA 4.3.4. Reticulado $\mathbf{3} \times \mathbf{3}$ 

- $\{(0, 0), (1, 0), (1, 1), (1, 2), (1, 3), (2, 0), (2, 2), (3, 0), (3, 3)\}$
- $\{(0, 0), (0, 1), (0, 2), (0, 3), \dots, (3, 0), (3, 1), (3, 2), (3, 3)\}$

Las cuales pueden ser interpretadas respectivamente como:

- Δ
- $\{(x, y) : x \leq y\}$
- $\{(x, y) : x \geq y\}$
- A^2

Lo que mediante el Teorema 36, define un subreticulado de $\mathbf{E}_2(\mathbf{2} \times \mathbf{2})$ anterior, mucho mas pequeño, como se puede ver en la Figura 4.3.3, donde los nodos sin rellenar son los elementos join-irreducibles devueltos por el paquete.

FIGURA 4.3.3. Reticulado de relaciones binarias definibles por existenciales positivas en $\mathbf{2} \times \mathbf{2}$ 

Luego probamos en el reticulado $\mathbf{3} \times \mathbf{3}$ representado en la figura 4.3.4.

Al interpretar los elementos join-irreducibles de $\mathbf{E}_2^+(\mathbf{3} \times \mathbf{3})$ calculados por el paquete, nos dio nuevamente:

- Δ
- $\{(x, y) : x \leq y\}$
- $\{(x, y) : x \geq y\}$
- A^2

Lo que representa un reticulado isomorfo al presentado en 4.3.3. Esto nos llevo a conjeturar el Teorema 43, que probamos a continuación.

4.3.2. Caracterización de las relaciones binarias definibles por existenciales positivas en reticulados distributivos.

LEMA 40. Sea \mathbf{L} un reticulado y R una relación binaria no vacía sobre L preservada por endomorfismos en \mathbf{L} . Entonces vale que $\Delta^L \subseteq R$.

DEMOSTRACIÓN. Esto es una consecuencia directa de que para cada $a \in L$ la función de L en L que vale constantemente a es un endomorfismo. \square

TEOREMA 41 (Teorema del filtro primo). Sea $\langle L, \vee, \wedge \rangle$ un reticulado distributivo y F un filtro. Supongamos $x \in L - F$. Entonces hay un filtro primo P tal que $x \notin P$ y $F \subseteq P$.

LEMA 42. Sea \mathbf{L} un reticulado distributivo y R una relación binaria no vacía sobre L preservada por endomorfismos en \mathbf{L} . Si hay $(a, b) \in R$ tal que $a \not\leq b$ (respectivamente $b \not\leq a$), entonces $\{(x, y) : x \geq y\} \subseteq R$ (respectivamente $\{(x, y) : x \leq y\} \subseteq R$).

DEMOSTRACIÓN. Fijamos $(a, b) \in R$ tal que $a \not\leq b$, y sean $c, d \in L$ tales que $c \geq d$. Veremos que $(c, d) \in R$. Por el Teorema 41 hay un filtro primo P que contiene al filtro generado por a y además $b \notin P$. Definimos $h : L \rightarrow L$ por

$$h(x) = \begin{cases} c & \text{si } x \in P, \\ d & \text{si } x \notin P. \end{cases}$$

Es fácil ver que h es un endomorfismo. Finalmente como $(a, b) \in R$ y h preserva R , $(h(a), h(b)) = (c, d) \in R$. \square

TEOREMA 43. Sea \mathbf{L} un reticulado distributivo y R una relación binaria sobre L definible por un fórmula existencial positiva en \mathbf{L} . Se da una de las siguientes:

- $R = \Delta$,
- $R = \{(x, y) : x \leq y\}$,
- $R = \{(x, y) : x \geq y\}$,
- $R = \{(x, y) : x \leq y\} \cup \{(x, y) : x \geq y\}$,
- $R = L \times L$.

DEMOSTRACIÓN. Si $R \subseteq \Delta$, por el Lema 40 $R = \Delta$.

Si $R \subseteq \{(x, y) : x \leq y\}$, pero $R \not\subseteq \Delta$, hay $(a, b) \in R$ tales que $a < b$, entonces $a \not\geq b$ y por el Lema 42 $R = \{(x, y) : x \leq y\}$. Análogo para $R \subseteq \{(x, y) : x \geq y\}$.

Si $R \not\subseteq \{(x, y) : x \leq y\}$ y $R \not\subseteq \{(x, y) : x \geq y\}$, aplicando dos veces el Lema 42 $\{(x, y) : x \leq y\} \cup \{(x, y) : x \geq y\} \subseteq R$. Si además $(a, b) \in R$ pero $a \not\leq b$ y $a \not\geq b$, tomo $(c, d) \in R$, si son comparables ya esta están en R por lo anterior. Si son incomparables aplico dos veces el Teorema 41 y tomo dos filtros primos P y Q tales que $a \in P$ pero $b \notin P$ y $b \in Q$ pero $a \notin Q$. Ahora definimos la siguiente función

$$h(x) = \begin{cases} c \vee d & \text{si } x \in P \cup Q \\ c & \text{si } x \in P - Q \\ d & \text{si } x \in Q - P \\ c \wedge d & \text{si } x \notin P \text{ y } x \notin Q \end{cases}$$

que fácilmente puede verse que es un endomorfismo. Por lo tanto como $(a, b) \in R$ y h preserva R , $(h(a), h(b)) = (c, d) \in R$. Por lo tanto $R = L \times L$. \square

Documentación de Definability

5.1. Introducción

Definability es un paquete que desarrollamos para SageMath, un software matemático licenciado bajo la GPL. Nuestro paquete implementa los algoritmos vistos en las secciones anteriores y permite dada una clase de estructuras, decidir definibilidad o generar las álgebras de Lindenbaum de relaciones definibles.

Nuestro paquete se basa en parte en las librerías desarrolladas por Peter Jipsen para utilizar Minion, Universal Álgebra Calculator, Mace4 y Prover9 desde SageMath que se pueden obtener en <http://math.chapman.edu/~jipsen/sagepkg/>.

El paquete se desarrolla en <https://github.com/pablogventura/tesis>.

5.2. Instalación

5.2.1. Requerimientos.

- SageMath 6.7 o superior.
- Minion 1.8 o superior.
- LADR versión de noviembre de 2009, (interfaz de línea de comandos de Prover9, Mace4, y otros programas)
- Git

5.2.2. Instalación del paquete en SageMath. Suponiendo que tenemos instalado SageMath en la ruta absoluta SAGEDIR, nos movemos al directorio donde se quiere descargar el paquete Definability, seguimos los siguientes comandos para instalar el paquete en nuestra instalación de SageMath:

```
$ git clone https://github.com/pablogventura/tesis.git
$ cd tesis/package
$ ./make_spkg.sh
$ mv definability-hash.spkg SAGEDIR
$ cd SAGEDIR
$ ./sage -i definability-hash.spkg
```

5.3. Uso del paquete

Una vez instalado el paquete basta con arrancar Sage con el comando:

```
$ ./sage
```

y una vez en la consola de Sage, importamos el paquete con el comando:

```
sage: import definability
```

5.4. Generación y entrada de estructuras

Al momento de ingresar estructuras para su chequeo puede optarse por la generación automática de estructuras a partir de una teoría de primer orden mediante la interfaz a Mace4, o la entrada de una estructura en particular manualmente.

5.4.1. Generación de estructuras de una teoría de primer orden.

Para definir un objeto del tipo `FO_Theory` que implementa a una teoría de primer orden basta con un nombre, una descripción, y una lista de axiomas en la sintaxis propia de Mace4. Por ejemplo para definir la teoría de reticulados bastaría con la siguiente línea:

```
Lat = definability.FO_Theory("Lat",
                              "Lattices",
                              [ '(x v y) v z = x v (y v z) ',
                                'x v y = y v x ',
                                '(x^y)^z = x^(y^z) ',
                                'x^y = y^x ',
                                '(x v y)^x = x ',
                                '(x^y) v x = x ' ])
```

Ahora para recorrer los reticulados (filtrando isomorfismos) basta con llamar al método `find_models` con la cardinalidad buscada, el cual devuelve un generador. Por ejemplo para recorrer los reticulados de 5 elementos basta con

```
list(Lat.find_models(5))
```

que devolverá una lista con todos los reticulados de 5 elementos filtrando isomorfismos.

Nuestro paquete incluye una gran variedad de teorías de primer orden ya cargadas en el módulo `fotheories`. Por ejemplo: `Lat` para reticulados, `Graph` para grafos, `Grp` para grupos, etc. La lista completa se puede revisar mediante listar el módulo `fotheories` de nuestro paquete con el siguiente comando:

```
dir(definability.fotheories)
```

5.4.2. Entrada manual de una estructura. Supongamos que queremos ingresar manualmente el reticulado 2×2 . Lo primero es definir el tipo de primer orden al que pertenece. Para crear un objeto de la clase `FO_Type` basta con pasar dos diccionarios, el primero para las funciones y el segundo para las relaciones. Los diccionarios contienen las aridades y se indexan por los símbolos correspondientes. Por ejemplo, para definir el tipo de los reticulados, junto con una relación unaria P para chequear:

```
tlat = definability.FO_Type({'v': 2, '^': 2}, {"P": 1})
```

Luego se debe definir un universo para la estructura, como por ejemplo

```
universe = [0, 1, 2, 3]
```

Luego se necesitan definir las funciones supremo e infimo, y una relación unaria P para su chequeo.

Una función, implementada por la clase `FO_Operation` se puede definir mediante un diccionario con claves de tuplas que apuntan al valor de la función, o en el caso de funciones binarias, una matriz A donde el valor $A_{i,j}$ se corresponde a la función evaluada en (i, j) . Para definir una relación utilizando la clase `FO_Relation` basta con una lista de las tuplas que pertenecen a la relación y el universo sobre el que la relación está definida.

Como ejemplo, definimos el infimo y el supremo utilizando cada una de las maneras para definir funciones, y definimos la relación P :

```
meet = definability.FO_Operation({'(0, 0)': 0,
                                   '(0, 1)': 0,
                                   '(0, 2)': 0,
```

```

(0, 3): 0,
(1, 0): 0,
(1, 1): 1,
(1, 2): 2,
(1, 3): 3,
(2, 0): 0,
(2, 1): 2,
(2, 2): 2,
(2, 3): 0,
(3, 0): 0,
(3, 1): 3,
(3, 2): 0,
(3, 3): 3})
join = definability.FO_Operation([ [0,1,2,3],
                                     [1,1,1,1],
                                     [2,1,2,1],
                                     [3,1,1,3] ])
rP = definability.FO_Relation([ (2, ), (3, ) ], uni)

```

Finalmente, para definir el reticulado basta con hacer:

```

l2x2 = definability.FO_Model(tlat, uni, { '^': meet,
                                           'v': join,
                                           'P': rP })

```

5.5. Chequeo de definibilidad

Una vez definidas las estructuras, para chequear definibilidad se utiliza un objeto de la clase `Constellation`, pasando una lista de las estructuras en \mathcal{K} . Esta clase implementa los algoritmos vistos en el Capítulo 3. Por ejemplo siguiendo con el ejemplo anterior:

```

c = definability.Constellation([l2x2])

```

Para chequear definibilidad estan los siguientes metodos:

- `is_existential_definable`
- `is_existential_positive_definable`
- `is_open_definable`
- `is_positive_open_definable`

los cuales toman dos tipos $\mathcal{L} \subseteq \mathcal{L}'$ para decidir definibilidad de las relaciones del tipo $\mathcal{L}' - \mathcal{L}$ en \mathcal{L} , y devuelven una tupla con un booleano y un contraejemplo, si es que lo hay.

Por ejemplo para chequear definibilidad abierta de la relacion P :

```

c.is_open_definable(tlat, tlat + definability.FO_Type({}, {"P": 1}))

```

Lo cual en particular es falso, por lo que el paquete devuelve la tupla:

```

(False, Embedding([0] -> 2,
                  FO_Type({'v': 2, '^': 2}, {}),
                  antitype= ['P'],
                  Injective, ))

```


Donde devuelve un contraejemplo, ya que el embedding del subreticulado formado por el elemento 0 en el subreticulado formado por el elemento 2 (un isomorfismo entre subestructuras de $\mathbf{2} \times \mathbf{2}$) no preserva P , negando el Teorema 11.

5.6. Generación de álgebras de Lindenbaum

Para la generación de las álgebras de Lindenbaum desarrolladas en el capítulo 4 el paquete dispone del módulo lindenbaum.

Este módulo contiene las siguientes funciones:

ji_of_existencial_definable_algebra: para generar los elementos join irreducibles de $\mathbf{E}_n(\mathcal{K})$

ji_of_existencial_positive_definable_algebra: para generar los elementos join irreducibles de $\mathbf{E}_n^+(\mathcal{K})$

atoms_of_open_definable_algebra: para generar los átomos de $\mathbf{Op}_n(\mathcal{K})$

ji_of_open_positive_definable_algebra: para generar los elementos join irreducibles de $\mathbf{Op}_n^+(\mathcal{K})$

Estas funciones toman un objeto del tipo Constellation que determina \mathcal{K} , un tipo para los morfismos en cuestión y una aridad.

Por ejemplo para obtener los elementos join-irreducibles del reticulado de relaciones ternarias definibles en $\mathbf{2} \times \mathbf{2}$ bastaría con la siguiente línea:

```
definability.lindenbaum.  
  ji_of_existencial_positive_definable_algebra(c,  
    definability.examples.tiporet,3)
```

Lo cual devuelve una lista de 22 relaciones join-irreducibles, que acortamos como ejemplo:

```
[ Relation([0, 0, 0],  
           [1, 1, 1],  
           [2, 2, 2],  
           [3, 3, 3]),  
.  
.  
.  
Relation([0, 0, 0],  
         [0, 1, 1],  
         [0, 2, 2],  
         [0, 3, 3],  
         [1, 0, 0],  
         [1, 1, 1],  
         [1, 2, 2],  
         [1, 3, 3],  
         [2, 0, 0],  
         [2, 1, 1],  
         [2, 2, 2],  
         [2, 3, 3],  
         [3, 0, 0],  
         [3, 1, 1],  
         [3, 2, 2],  
         [3, 3, 3]) ]
```

Conclusiones y trabajo futuro

La ventaja de utilizar CSP es que hay múltiples circunstancias en las que uno de estos problemas se pueden resolver muy eficientemente. Esperamos explorar detalladamente estas situaciones para aprovecharlas para la búsqueda de homomorfismos cuando se den propiedades específicas para las estructuras en cuestión.

También esperamos explorar mejores implementaciones para la generación de subestructuras, investigando a fondo los algoritmos utilizados por el Universal Algebra Calculator.

- Que un conjunto de flechas alcance para revisar preservación implica que genera el conjunto de flechas completo?
- Para chequear si alguien es subestructura de otro, quizá conviene recorrer mapeos que son embeddings sabiendo que la imagen va a ser una subestructura isomorfa.
- Que hace Minion cuando se le prohíben soluciones (negativetable). Quizás primero experimentar con ejemplos.
- Explorar la relación entre generadores de una subestructura y homomorfismos.
- De menor a mayor o de mayor a menor las subestructuras?

Bibliografía

- Campercholi, Miguel y Diego Vaggione (2015). *Semantical conditions for the definability of functions and relations*. eprint: 1506.07501. URL: <http://www.arxiv.org/abs/1506.07501>.
- Wikipedia (2016). *Constraint satisfaction problem* — *Wikipedia, The Free Encyclopedia*. [En línea, visitado el 11 de Enero de 2016]. URL: <http://en.wikipedia.org/w/index.php?title=Constraint%20satisfaction%20problem&oldid=695651340>.