Apunts d'Introducció a la Lògica

Rafel Farré i Cirera

Versió provisional:
en construcció

10 de març de 2005

Índex

Capítol 1

Sintaxi i semàntica de la Lògica Proposicional

1.1 Sintaxi de la LProp

Aquí donarem la sintaxi i farem la construcció de les fórmules de la Lògica Proposicional. Partim d'un conjunt de variables proposicionals (o àtoms):

$$\{P, Q, R, S, \ldots\},$$

les connectives lògiques:

$$\neg, \wedge, \vee, \rightarrow, \leftrightarrow$$

i els parèntesis obrint: (i tancant:). Aquests són els símbols que fem servir per a construir les fórmules. Les regles de construcció són les següents:

Definició 1.1.1.

- Les variables proposicionals són fórmules
- Si A i B són fórmules, també són fórmules:

$$\neg A,\ (A \wedge B),\ (A \vee B),\ (A \rightarrow B), (A \leftrightarrow B).$$

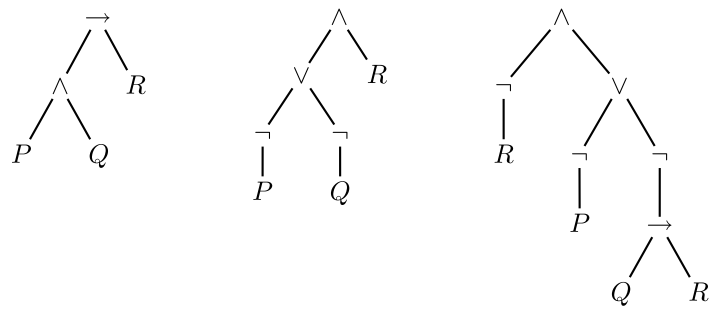
- No hi ha més fórmules que les obtingudes aplicant les dues regles anteriors.

Exemple 1.1.2. $((P \wedge Q) \rightarrow R)$, $(\neg P \vee \neg Q) \wedge R)$, $(\neg R \wedge (\neg P \vee \neg (Q \rightarrow R)))$ són fórmules, mentre que ni $(\rightarrow)\vee$ ni $\wedge\neg)P$ són fórmules.

Cada fórmula, a més de la forma textual, que és la que acabem de donar, es pot posar en forma d'arbre. Aquesta es construeix amb el mateix procés recursiu pel qual es construeixen les fórmules. Sense donar-ne la definició formal direm que els nodes interns estan etiquetats amb connectives lògiques. Les connectives binàries $\wedge, \vee, \rightarrow$ tenen dos fills i la connectiva $\neg$ en té un. A les fulles hi ha les variables proposicionals. La forma arbòria és única.

definició recursiva arbre

Exemple 1.1.3. La forma d'arbre del les tres fórmules de l'exemple anterior 1.1.2 $((P \wedge Q) \to R)$, $(\neg P \vee \neg Q) \wedge R)$, $(\neg R \wedge (\neg P \vee \neg(Q \to R)))$ és la següent:



Com que no hi ha més fórmules que les que es poden construir a partir dels àtoms aplicant connectives segons la definició 1.1.1, tindrem el principi d'inducció següent:

Proposició 1.1.4. *Si $\mathcal{P}$ és una propietat tal que*

- *Les variables proposicionals tenen la propietat $\mathcal{P}$*
- *Si A i B tenen la propietat $\mathcal{P}$, llavors també $\neg A$, $(A \wedge B)$, $(A \vee B)$, $(A \to B)$ i $(A \leftrightarrow B)$ tenen la propietat $\mathcal{P}$*

llavors totes les fórmules de la lògica proposicional tenen la propietat $\mathcal{P}$.

Exemple 1.1.5. Anem a demostrar, a mode d'exemple, que en tota fórmula proposicional A val el següent: el nombre d'aparicions de variables (comptant multiplicitats) és un més que el nombre d'aparicions de connectives binàries. En el cas de les variables és obvi: hi ha una variable i zero connectives. Si A ho satisfà, també ho satisfà $\neg A$ ja que A i $\neg A$ tenen el mateix nombre de connectives binàries i de variables. Si $C = A * B$ amb $*$ una de les connectives binàries ($\wedge, \vee, \to$ o $\leftrightarrow$), notem per n i m el nombre de connectives binàries de A i B respectivament. Llavors, per hipòtesi d'inducció A i B tenen $n+1$ i $m+1$ aparicions de variables respectivament. Llavors, el nombre de connectives binàries de $A * B$ és $n+m+1$ i el seu nombre de variables és $(n+1)+(m+1)$, que és un més que el de connectives binàries.

Exercici 1.1.6. Demostreu que si en una fórmula A la connectiva $\neg$ hi apareix n cops i hi ha m aparicions de connectives binàries, llavors el nombre de subfórmules de A és $1+n+2m$. Les subfórmules d'una fórmula són totes aquelles que cal haver construït prèviament: per a cada node de l'arbre hi ha una subfórmula que 'penja' del node (la fórmula A la considerem subfórmula de si mateixa).

L'objectiu dels parèntesis és d'evitar possibles ambigüitats en la lectura de la fórmula. Tot i que hem intentar introduir els mínims possibles, encara s'en poden suprimir alguns, com per exemple l'exterior. Per conveni, establirem un ordre de prioritats. El que té la prioritat més alta és $\neg$, després venen $\vee, \wedge$

amb igual prioritat i finalment $\to, \leftrightarrow$ també amb la mateixa prioritat. Cal interpretar que el més prioritari té un abast més curt. La *forma abreujada* d'una fórmula és la que s'obté al suprimim tots els parèntesis 'innecessaris', és a dir, tots el parèntesis que, al suprimir-los i aplicant les prioritats indicades, no canvien la fórmula. Quan tenim connectives amb la mateixa prioritat o la mateixa connectiva repetida, no podem suprimir els parèntesis. Per exemple $A \to B \to C$ no sabem si es tracta de la fórmula $(A \to B) \to C$ o de la fórmula $A \to (B \to C)$. El mateix passa, per exemple, amb les fórmules $A \leftrightarrow B \to C$ i $A \vee B \wedge C$: no sabem si ens referim a $(A \leftrightarrow B) \to C$ o a $A \leftrightarrow (B \to C)$ en el primer cas i a la fórmula $(A \vee B) \wedge C$ o a la fórmula $A \vee (B \wedge C)$ en el segon. Hi ha una excepció a aquest fet: quan tenim diversos $\vee$ o diversos $\wedge$. Acceptarem com a correcte la fórmules $A \wedge B \wedge C$. Tot i que formalment és la forma abreujada de dues fórmules en principi diferents (les fórmules $(A \wedge B) \wedge C$ i $A \wedge (B \wedge C)$, són diferents, feu-ne l'arbre), aquestes fórmules són equivalents, com veurem més endavant. De la mateixa manera acceptarem $A \wedge B \wedge C \wedge D$ o més conjuncions. També acceptarem tres o més disjuncions sense parèntesi.

Exemple 1.1.7. la fórmula $((P \wedge Q) \to R)$ la podem escriure $P \wedge Q \to R$. La fórmula $(\neg(P \wedge Q) \to R)$ la podem escriure $\neg(P \wedge Q) \to R$, però no podem suprimir el parèntesi que queda, doncs $\neg P \wedge Q \to R$ és la forma abreujada de la fórmula $((\neg P \wedge Q) \to R)$. La fórmula $\neg((P \wedge Q) \to R)$ la podem escriure $\neg(P \wedge Q \to R)$ però tampoc podem suprimir el parèntesi que queda.

1.2 Semàntica de la LProp

Una interpretació I serà una assignació de valors de veritat a cada una de les variables proposicionals. Els valors de veritat seran el 0 i el 1. El 0 indica 'fals' i el 1 indica 'cert'.

Definició 1.2.1. Una *interpretació* és una aplicació:

$$I : \{P, Q, R, S, \ldots\} \to \{0, 1\}.$$

Un cop assignat un valor de veritat a cada variable proposicional, de manera recursiva, assignem un valor de veritat a cada fórmula proposicional. És a dir, podem estendre I al conjunt de les fórmules proposicionals de la manera següent:

Definició 1.2.2.

- $I(\neg A) := \neg I(A)$,
- $I(A \wedge B) := I(A) \wedge I(B)$,
- $I(A \vee B) := I(A) \vee I(B)$,
- $I(A \to B) := I(A) \to I(B)$,
- $I(A \leftrightarrow B) := I(A) \leftrightarrow I(B)$.

En aquesta definició, cal entendre que les connectives lògiques $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$ operen a l'Àlgebra de Boole binària de la manera habitual:

A	$\neg A$
0	1
1	0

A	B	$A \wedge B$	$A \vee B$	$A \rightarrow B$	$A \leftrightarrow B$
0	0	0	0	1	1
1	0	0	1	0	0
0	1	0	1	1	0
1	1	1	1	1	1

(1.2.1)

Exemple 1.2.3. Considerem les fórmules

$$A := P \wedge Q \rightarrow R, \quad B := (\neg P \vee \neg Q) \wedge R, \quad C := \neg R \wedge (\neg P \vee \neg(Q \rightarrow R))$$

de l'exemple 1.1.2 i la interpretació següent:

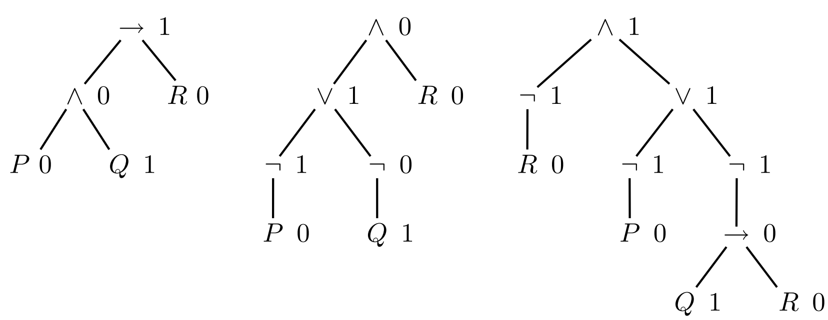
$$\begin{array}{rcccc} I: & \{P, Q, R, \ldots\} & \rightarrow & \{0,1\} \\ & P & \mapsto & 0 \\ & Q & \mapsto & 1 \\ & R & \mapsto & 0. \end{array}$$

Anem a calcular els seus valors de veritat per I:

- $I(A) = I(P \wedge Q \rightarrow R) = I(P \wedge Q) \rightarrow I(R) = I(P) \wedge I(Q) \rightarrow I(R) = 0 \rightarrow 0 = 1$.
- $I(B) = I(\neg P \vee \neg Q) \wedge I(R) = (\neg I(P) \vee \neg I(Q)) \wedge I(R) = (\neg 0 \vee \neg 1) \wedge 0 = 0$.
- $I(B) = I(\neg R) \wedge I(\neg P \vee \neg(Q \rightarrow R)) = \neg I(R) \wedge (\neg I(P) \vee \neg I(Q \rightarrow R))) = \neg I(R) \wedge (\neg I(P) \vee \neg(I(Q) \rightarrow I(R))) = \neg 0 \wedge (\neg 0 \vee \neg(1 \rightarrow 0)) = 1 \wedge (1 \vee \neg 0)) = 1 \wedge 1 = 1$.

Observem que per tal de calcular $I(A)$ només cal tenir definit I per a les variables que apareixen a la fórmula. També es pot calcular $I(A)$ directament sobre l'arbre, calculant recursivament el valor de cada node, començant per les fulles, fins arribar a l'arrel.

Exemple 1.2.4. Si calculem el valor de veritat de les tres fórmules A, B, C per la interpretació I de l'exemple 1.2.3 amb arbres:



Definició 1.2.5. Quan $I(A) = 1$ direm que A *val a* I o també que I *és model de* A. En símbols, ho escriurem

$$I \models A.$$

Dos conceptes bàsics són el de satisfabilitat i el de conseqüència lògica.

Definició 1.2.6 (satisfabilitat)**.** Una fórmula proposicional A és *satisfactible* quan té algun model. Més en general: un conjunt de fórmules proposicionals Γ és *satisfactible* quan tenen un model comú, i.e., existeix una interpretació I que és model de totes les fórmules de Γ. A una fórmula (o un conjunt de fórmules) que no és satisfactible en direm *insatisfactible*.

Exemple 1.2.7. Les tres fórmules de l'exemple 1.2.3 són satisfactibles (una a una). La fórmula $P \wedge \neg P$ és insatisfactible. La fórmula $(P \vee Q) \wedge (P \vee \neg Q) \wedge (\neg P \vee Q) \wedge (\neg P \vee \neg Q)$ és insatisfactible. El conjunt de fórmules $\{P, \neg Q, \neg P \wedge Q\}$ és insatisfactible, encara que cada una de les fórmules del conjunt és satisfactible. El conjunt de fórmules $\{P \vee Q, P \vee \neg Q, \neg P \vee Q, \neg P \vee \neg Q\}$ és insatifactible, mentre que el subconjunt $\{P \vee Q, P \vee \neg Q, \neg P \vee Q\}$ és satisfactible.

Exercici 1.2.8. El conjunt de fórmules $\{A_1, \ldots, A_n\}$ és satisfactible ssi ho és la fórmula $A_1 \wedge \cdots \wedge A_n$.

Observem que justificar la satisfactibilitat d'una fórmula és 'ràpid' si tenim idea de com són els seus models (més precisament: verificar que una interpretació donada I satisfà una fórmula es pot fer en temps 'polinomial'), mentre que si no sabem qui pot ser I i hem de verificar tots els possibles models I requereix en general temps exponencial (més precisament: verificar que una fórmula és satisfactible és un problema NP-complet). Demostrar insatisfabilitat requereix, en principi, repassar totes les possibles assignacions de valors de veritat (interpretacions) a les variable que intervenen a la fórmula (o al conjunt de fórmules). El problema és que si hi ha n variables, hi ha 2^n interpretacions amb aquestes variables.

Definició 1.2.9 (fórmula vàlida)**.** Una fórmula A es diu que és *vàlida* o que és una *tautologia* quan val a tota interpretació. Ho denotarem així:

$$\models A.$$

Exemple 1.2.10. Les fórmula $P \vee \neg P$ i $P \vee P \to P$ són tautologies. La fórmula $(P \wedge Q) \vee (P \wedge \neg Q) \vee (\neg P \wedge Q) \vee (\neg P \wedge \neg Q)$ també és una tautologia, mentre que $(P \wedge Q) \vee (P \wedge \neg Q)$ no ho és. Tampoc són tautologies cap de les fórmules del l'exemple 1.2.3.

Els conceptes de 'fórmula vàlida' i 'fórmula satisfactible' estan estretament lligats:

Proposició 1.2.11. *Si A és una fórmula, llavors:*

1. *A és satisfactible sii $\neg A$ no és vàlida.*

2. A és vàlida sii $\neg A$ és insatisfactible.

Demostració. Farem només 1, l'altre és idèntic. Que la fórmula $\neg A$ no és vàlida vol dir que hi ha una interpretació I amb $I(\neg A) = 0$. Però això és equivalent a l'existència d'una interpretació amb $I(A) = 1$, o sigui, satisfactibilitat de A. □

Exercici 1.2.12. Demostreu que el conjunt de fórmules $\{A_1, \dots, A_n\}$ és satisfactible sii la fórmula $\neg A_1 \vee \cdots \vee \neg A_n$ no és vàlida.

Definició 1.2.13 (conseqüència lògica)**.** Una fórmula proposicional B és *conseqüència lògica* de A (en símbols: $A \models B$) quan tot model de A també és model de B. Més en general: una fórmula proposicional B és *conseqüència lògica* d'un conjunt de fórmules proposicionals Γ (en símbols: $\Gamma \models B$) quan tota interpretació I que és model de totes les fórmules de Γ també és model de B.

Exemple 1.2.14. La fórmula Q és conseqüència lògica de la fórmula $P \wedge (P \to Q)$; en símbols: $P \wedge (P \to Q) \models Q$. Igualment Q és conseqüència lògica del conjunt de fórmules $\{P, P \to Q\}$, en símbols: $\{P, P \to Q\} \models Q$. També $\neg Q$ és conseqüència lògica del conjunt $\{P \vee \neg Q, \neg P\}$, en símbols: $\{P \vee \neg Q, \neg P\} \models \neg Q$. Un altre exemple: $\{P, P \to Q, \neg R \to \neg Q\} \models R$. En canvi P no és conseqüència lògica de $P \vee Q$, en símbols: $P \vee Q \not\models P$.

Observació 1.2.15 (**important**). Totes les nocions anteriorment introduïdes estan molt relacionades com veurem a la proposició següent. De fet, tots els conceptes aquí introduïts es poden definir partir del concepte de conseqüència lògica (i també a partir del concepte de satisfabilitat). Quan fem la lògica de primer ordre n'hi haurà prou amb definir un d'aquest conceptes per a tenir-los tots.

Proposició 1.2.16. *1. Si Γ és un conjunt de fórmules i A és una fórmula, llavors*

$$\Gamma \models A \quad \textit{sii} \quad \Gamma \cup \{\neg A\} \ \textit{és insatisfactible.}$$

2. Si $\{A_1, \dots, A_n\}$ és un conjunt finit de fórmules i A és una fórmula, llavors són equivalents:

(i) $\{A_1, \dots, A_n\} \models A$

(ii) $A_1 \wedge \cdots \wedge A_n \models A$

(iii) la fórmula $A_1 \wedge \cdots \wedge A_n \to A$ *és vàlida.*

Demostració. Per a demostrar *1*, només cal observar que tot model de Γ ho és també de A sii no hi ha cap model de $\Gamma \cup \neg A$.

L'equivalència entre *(i)* i *(ii)* de *2* és conseqüència del fet següent: una interpretació I és model de totes les fórmules del conjunt $\{A_1, \dots, A_n\}$ sii és model de la fórmula de $A_1 \wedge \cdots \wedge A_n$. Veiem que *(ii)* implica *(iii)*. Si una interpretació I és model de $A_1 \wedge \cdots \wedge A_n$ llavors també és model de A i per tant $I(A_1 \wedge \cdots \wedge A_n \to A) = 1$. Si I no és model de la fórmula $A_1 \wedge \cdots \wedge A_n$ llavors també $I(A_1 \wedge \cdots \wedge A_n \to A) = 1$. Ara fem la demostració de $(iii) \Rightarrow (ii)$. Sigui I una interpretació model de $A_1 \wedge \cdots \wedge A_n$. Com que $I(A_1 \wedge \cdots \wedge A_n) = 1$ i $I(A_1 \wedge \cdots \wedge A_n \to A) = 1$ necessàriament $I(A) = 1$. □

Exercici 1.2.17. Demostreu que si Γ és un conjunt de fórmules, llavors:

$$\Gamma \text{ és satisfactible sii } \Gamma \not\models P \wedge \neg P$$

Capítol 2

Sintaxi i semàntica de la Lògica de Primer Ordre

2.1 Relacions i operacions en un conjunt

2.1.1 Parts i Potència d'un conjunt

Si D és un conjunt, anomenaren parts de D, i el denotarem per $P(D)$, al conjunt dels subconjunts de D:

$$P(D) = \{X \mid X \subseteq D\}.$$

Exemple 2.1.1. Si $D = \{\alpha, \beta, \gamma\}$, llavors

$$P(D) = \{\emptyset, \{\alpha\}, \{\beta\}, \{\gamma\}, \{\alpha, \beta\}, \{\beta, \gamma\}, \{\alpha, \gamma\}, \{\alpha, \beta, \gamma\}\}.$$

El cardina d'un conjunt D el denotarem per $|D|$. El cardina de $P(D)$ és

$$|P(D)| = 2^{|D|}.$$

La potència n-èssima d'un conjunt D, que denotarem per D^n, és el conjunt de les seqüències de longitud n de D:

$$D^n = \{(\alpha_1, \ldots, \alpha_n) \mid \text{ cada } \alpha_i \in D\}.$$

Quan $n = 1$ identifiquem D^1 amb D.

Exemple 2.1.2. Si $D = \{\alpha, \beta, \gamma\}$ llavors

$$D^2 = \{(\alpha, \alpha), (\alpha, \beta), (\alpha, \gamma), (\beta, \alpha), (\beta, \beta), (\beta, \gamma), (\gamma, \alpha), (\gamma, \beta), (\gamma, \gamma)\}.$$

Observem que el cardinal de D^n és

$$|D^n| = |D|^n.$$

2.1.2 Relacions

Cada relació té una arietat, que és un nombre natural $n \geq 1$.

Definició 2.1.3. Una *relació n-ària* a D és un subconjunt de D^n.

Al conjunt D en direm domini de la relació.

Exemple 2.1.4. Si $D = \{\alpha, \beta, \gamma\}$,

- $\{(\alpha, \beta), (\beta, \gamma), (\gamma, \alpha)\}$ és una relació binària.
- $\{\alpha, \beta\}$ és una relació 1-ària.
- $\{(\alpha, \alpha, \alpha), (\beta, \beta, \beta), (\gamma, \gamma, \gamma)\}$ és una relació ternària.

Exemple 2.1.5. Una família: Joan és el pare, Maria és la mare i ténen tres fills: Pere, Anna i Manel. En aquest exemple $D = \{Joan, Maria, Pere, Anna, Manel\}$,

- 'Ser dona' és una relació 1-ària: $\{Maria, Anna\}$.
- 'Ser pare de' és una relació binària:
  $\{(Joan, Pere), (Joan, Anna), (Joan, Manel)\}$.
- 'Ser germans' és una relació binària: $\{(Pere, Anna), (Pere, Manel),$ $(Anna, Pere), (Anna, Manel), (Manel, Pere), (Manel, Anna)\}$.
- 'Ser germans' també ho podem entendre com una relació ternària (o de l'arietat que vulguem, però en cada cas és una relació diferent):
  $\{(Pere, Anna, Manel), (Pere, Manel, Anna), (Anna, Pere, Manel),$ $(Anna, Manel, Pere), (Manel, Pere, Anna), (Manel, Anna, Pere)\}$.

Exemple 2.1.6. Si $D = \mathbb{N}$,

- $\{n \mid n \text{ és parell}\}$ és una relació 1-ària ('ser parell').
- $\left\{(n, m) \in \mathbb{N}^2 \mid n < m\right\}$ és una relació binària ('ser menor que')
- $\{(n, m, r) \mid n + m = r\}$ és una relació ternària ('ser la suma d'altres dos').
- $\left\{(n, m, r) \in \mathbb{N}^2 \mid n < r < m\right\}$ és una relació ternària ('estar entre')

Com que el conjunt de les relacions n-àries de D és $P(D^n)$, el nombre de relacions n-àries de D és

$$|P(D^n)| = 2^{|D|^n}.$$

2.1.3 Operacions

Les operacions també tenen una arietat, que és un nombre natural $n \geq 1$.

Definició 2.1.7. Una *operació n-ària* a D és una funció

$$f : D^n \to D.$$

Cal que tots els elements de D^n tinguin imatge. A vegades a les operacions també s'els hi diu lleis internes. Al conjunt D en direm domini de la operació.

Exemple 2.1.8. Considerem el domini $D := \{\alpha, \beta, \gamma\}$.

- la següent és una funció 1-ària a D:

$$\begin{array}{ccc} D & \rightarrow & D \\ \alpha & \mapsto & \beta \\ \beta & \mapsto & \gamma \\ \gamma & \mapsto & \alpha \end{array}$$

- La següent és una funció binària a D:

$$\begin{array}{ccc} D^2 & \rightarrow & D \\ (\alpha,\alpha) & \mapsto & \alpha \\ (\alpha,\beta) & \mapsto & \gamma \\ (\alpha,\gamma) & \mapsto & \beta \\ (\beta,\alpha) & \mapsto & \gamma \\ (\beta,\beta) & \mapsto & \beta \\ (\beta,\gamma) & \mapsto & \alpha \\ (\gamma,\alpha) & \mapsto & \beta \\ (\gamma,\beta) & \mapsto & \alpha \\ (\gamma,\gamma) & \mapsto & \gamma. \end{array}$$

Exemple 2.1.9. En el domini dels naturals $\mathbb{N}$, tenim les funcions següents:

- La funció que envia a cada nombre el seu 'següent' $\mathbb{N} \rightarrow \mathbb{N}$, (definida per $x \mapsto x + 1$) és una operació 1-ària.
- La suma i el producte de naturals $\mathbb{N}^2 \rightarrow \mathbb{N}$, (definides per $(x, y) \mapsto x + y$ i $(x, y) \mapsto xy$ respectivament) són operacions binàries.
- La funció $\mathbb{N}^3 \rightarrow \mathbb{N}$, definida per $(x, y, z) \mapsto (x + y)z$ és una operació ternària.
- La funció $\mathbb{N}^n \rightarrow \mathbb{N}$, definida per $(x_1, \ldots, x_n) \mapsto x_1 + \cdots + x_n$ és una operació n-ària.

El conjunt de operacions n-àries de D és el conjunt de funcions de D^n a D. Com que el nombre de funcions de A a B és $|B|^{|A|}$, el nombre de operacions n-àries de D és

$$|D|^{|D^n|} = |D|^{|D|^n}.$$

2.2 Sintaxi de la LPO

2.2.1 Vocabulari

El vocabulari o conjunt de símbols no lògics de la LPO consta del següent:

- Símbols de constant. Tindrem una infinitat de símbols de constant:

$$a, b, c, d, \ldots$$

- Símbols de funció. Cada símbol de funció te una arietat [1]. Tindrem una quantitat infinita de símbols de funció n-ària per a cada $n \geq 1$:

$$f, g, h, \ldots$$

  Per tal d'evitar confusions, a vegades escriurem f/n per tal d'indicar que l'arietat de f és n.

- Símbols de relació. Cada símbol de relació té també una arietat[2]. Tindrem una quantitat infinita de símbols de relació n-ària per a cada $n \geq 1$:

$$P, Q, R, S, \ldots$$

  També R/n per indicar que n és l'arietat de R.

A més del vocabulari no lògic tindrem un conjunt infinit de variables:

$$x, y, z, \ldots \quad x_1, x_2, x_3, \ldots \quad y_1, y_2, y_3, \ldots,$$

les connectives proposicionals $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$, els quantificadors universal i existencial: $\forall, \exists$, el símbol d'igualtat $\approx$ i finalment els parèntesis i la coma: ( ) ,

2.2.2 Termes

Abans de construir les fórmules cal construir els termes, pensats per a designar elements d'un cert domini. Els termes es construeixen aplicant les regles següents:

Definició 2.2.1 (termes)**.**

- Les constants i les variables són termes.
- Si $t_1, \ldots, t_n$ són termes i f és un símbol de funció d'arietat n, llavors $f(t_1, \ldots, t_n)$ també és un terme.
- No hi ha més termes que els construïts aplicant les dues regles anteriors.

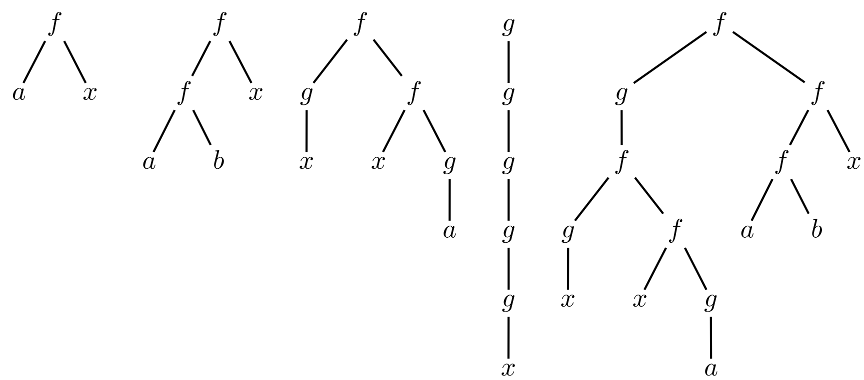
Exemple 2.2.2. Suposem que en el vocabulari tenim dos símbols de constant a, b un símbol de funció 1-ària g i un símbol de funció binària f. Els següents són termes: x, y, a, b, $f(x,y)$, $f(a,x)$, $f(f(a,b),x)$, $f(g(x), f(x, g(a)))$, $g(g(g(g(g(x)))))$, $f(g(f(g(x), f(x, g(a)))), f(f(a,b),x))$.

[1] Cal pensar que els símbols de funció serveixen per a denotar funcions (o operacions), i la arietat no és més que el nombre de variables de la funció que denoten (o el nombre d'operands de l'operació). Per exemple, si volem un símbol de funció per a denotar la 'suma', que és una d'operació binària, tindrà arietat 2

[2] Els símbols de relació serveixen per a denotar relacions o propietats, i aquestes, de manera natural tenen una arietat. Per exemple, la propietat de 'ser alt' és d'un sol individu i tindrà arietat 1, mentre que la relació 'ser amic de' relaciona 2 individus i tindrà arietat 2

Els termes es poden representar en forma d'arbre ordenat. Els nodes s'etiqueten amb els símbols de funció i tenen tants fills com la seva arietat. I així successivament.

Exemple 2.2.3. La forma d'arbre dels cinc últims termes de l'exemple 2.2.2 és el següent:



Observem que el subarbre que 'penja' de cada node representa també un terme. Les fulles estan etiquetades amb constants i variables i els nodes interns amb símbols de funció.

Els termes sense variables reben el nom de *termes tancats*.

2.2.3 Fórmules

Definició 2.2.4 (fórmules atòmiques)**.** Hi ha dos tipus de fórmules atòmiques:

- $Pt_1 \dots t_n$, on P és un símbol de relació n-ària i $t_1, \dots, t_n$ són termes.
- $t_1 \approx t_2$, on t_1 i t_2 són termes.

A vegades, per tal de facilitar-ne la lectura escriurem $P(t_1, \dots, t_n)$ enlloc de $Pt_1, \dots, t_n$. Les fórmules de la LPO es construeixen, a partir de les atòmiques aplicant regles:

Definició 2.2.5.
- Les fórmules atòmiques són fórmules
- Si A i B són fórmules, també són fórmules:

$$\neg A,\ (A \wedge B),\ (A \vee B),\ (A \rightarrow B),\ (A \leftrightarrow B),$$

- Si A és una fórmula i x una variable, també són fórmules:

$$\forall x A,\ \exists x A.$$

- No hi ha més fórmules que les obtingudes aplicant les tres regles anteriors.

Exemple 2.2.6. Suposem que en el vocabulari tenim els símbols següents: $P/2$, $Q/1$, $f/2$, $g/1$, a, b. Llavors les següents són fórmules:

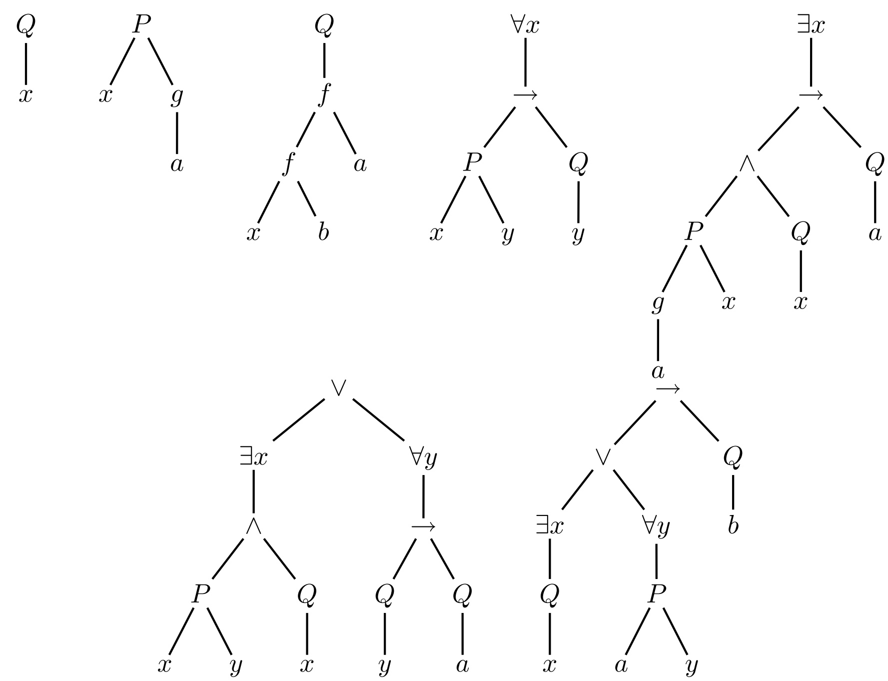
$$Qx, \quad Pxg(a), \quad Qf(f(x,b),a),$$

$$\forall x(Pxy \to Qy), \quad \exists x((Pg(a)x \wedge Qx) \to Qa)$$

$$(\exists x(Pxy \wedge Qx) \vee \forall y(Qy \to Qa)), \quad ((\exists xQx \vee \forall yPay) \to Qb)$$

De manera anàloga a la lògica proposicional, les fórmules es representen per arbres. Veiem con s'obté la forma d'arbre amb exemples.

Exemple 2.2.7. La forma d'arbre de les formules de l'exemple 2.2.6 és el següent:



Com en el cas de les fórmules de la LProp, la forma d'arbre està unívocament determinada. A les fulles hi ha constants i variables. Els nodes interns estan etiquetats amb connectives, símbols de funció i símbols de relació. A l'arbre que 'penja' de cada node hi ha, o be un terme, o be una fórmula.

També hi ha una notació abreujada: es poden suprimir parèntesis, sempre que no introdueixin ambigüitats en la lectura sota el conveni de prioritats següent: Primer $\forall x, \exists x, \neg$, segon $\wedge, \vee$ amb igual prioitat i tercer: $\to, \leftrightarrow$. La prioritat més alta es tradueix en abast més curt i, per conveni, cal suprimir tots el parèntesis 'redundants'.

Exemple 2.2.8. Les quatre últimes fórmules de l'exemple 2.2.6 es poden escriure de forma abreujada:

$$\forall x(Pxy \to Qy), \quad \exists x(Pg(a)x \wedge Qx \to Qa)$$

$$\exists x(Pxy \wedge Qx) \vee \forall y(Qy \to Qa), \quad \exists xQx \vee \forall yPay \to Qb$$

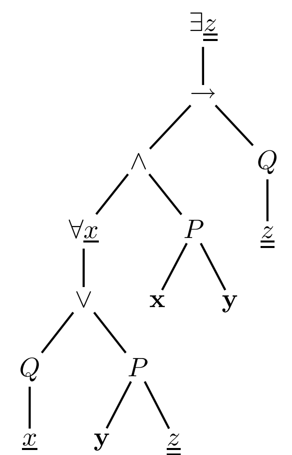
2.2.4 Variables lliure i substitucions

En una fórmula, una variable pot aparèixer varies vegades. Una aparició d'una variable x és *lliure* si no està sota l'efecte d'un quantificador $\forall x$ o $\exists x$ (ha de ser la mateixa variable: per exemple $\exists y$ no lliga x). Això es pot precisar millor pensant en l'arbre. Una aparició de la variable x serà una fulla de l'arbre etiquetada amb x, i serà lliure quan no hi ha un node per sobre etiquetat amb $\forall x$ o $\exists x$. Altrament direm que és una aparició *lligada*.

Exemple 2.2.9. La variable x apareix dues vegades a la fórmula

$$\exists z((\forall x(Qx \vee P\mathbf{y}z) \wedge P\mathbf{xy}) \to Qz);$$

la primera vegada apareix lligada, mentre que la segona apareix lliure. La variable y té dues aparicions lliures i la variable z té dues aparicions lligades. Les aparicions lliures de les variables estan en negreta. L'arbre de la fórmula ens pot ajudar a visualitzar-ho:



El quantificador $\exists z$ lliga totes les ocurrències de la z que estan 'per sota', en aquest cas totes (que hem subratllat $\underline{\underline{z}}$). El quantificador $\forall x$ lliga totes les ocurrències de la x que estan 'per sota', en a quest cas, només la $\underline{x}$ subratllada.

Definició 2.2.10. Les *variables lliures* d'una fórmula són aquelles que tenen alguna aparició lliure a la fórmula.

Per exemple,les variables lliures de la fórmula de l'exemple 2.2.9 són x i y.

Definició 2.2.11. Un *enunciat* o *fórmula tancada* és una fórmula que no té variables lliures.

Per exemple, la fórmula

$$\exists x \forall y(\forall x(Qx \vee Qy) \wedge \neg Pxy))$$

és un enunciat.

Definició 2.2.12. Una substitució és una aplicació que envia algunes variables a termes:

$$\begin{array}{ccc} x_1 & \mapsto & t_1 \\ x_2 & \mapsto & t_2 \\ x_3 & \mapsto & t_3 \\ \vdots & \vdots & \vdots \\ x_n & \mapsto & t_n \end{array}$$

i que denotarem així:

$$\{t_1/x_1, t_2/x_2, t_3/x_3, \ldots, t_n/x_n\} \tag{2.2.1}$$

Si A és una fórmula i s denota la substitució (2.2.1),

$$As$$

és el resultat d'aplicar **simultàniament**[3] s a totes les aparicions lliures de A, és a dir, substituïm cada aparició lliure de x_i a A per t_i, i això per cada $i = 1 \ldots n$ simultàniament.

Exemple 2.2.13.

1. $Pxy \to Qf(y,z)\,\{y/x, a/y\} = Pya \to Qf(a,z)$. Observeu que si primer fem la substitució $\{y/x\}$ i a continuació apliquem $\{a/y\}$ obtenim un resultat difrent:

$$Pxy \to Qf(y,z)\,\{y/x\}\,\{a/y\} = Paa \to Qf(a,z).$$

2. $Pxy \to \forall y Qf(y,z)\,\{y/x, a/y\} = Pya \to \forall y Qf(y,z)$. Aquí no hem substituït la segona aparició de y perquè està lligada.

3. $\exists y Pxy \to Qf(y,z)\,\{y/x, a/y\} = \forall y Pyy \to Qf(a,z)$. Aquí no hem substituït la primera aparició de y perquè està lligada.

Quan substituïm x per t a la fórmula A és d'esperar que $A\,\{t/x\}$ expressa que 't té la propietat A' (cal pensar que A expressa una propietat de x). A vegades, però, canvia radicalment el significat de la fórmula. Per exemple si A és la fórmula

$$A = \forall y Pxy,$$

A expressa que l'element x està relacionat (mitjançant P) amb qualsevol element. Així, quan fem la substitució $\{a/x\}$, la fórmula $A\,\{a/x\} = \forall y Pay$ expressa que 'a està relacionat amb qualsevol element'. Quan apliquem la substitució $\{y/x\}$ a A obtenim $A\,\{y/x\} = \forall y Pyy$, i aquesta fórmula expressa que tot element està relacionat amb si mateix (la relació R és reflexiva). Si substituïm $\{g(y)/x\}$ la fórmula $A\,\{g(y)/x\} = \forall y Pg(y)y$ expressa que $g(y)$ està relacionat amb y pera qualsevol y. Òbviament en aquests dos últims casos la fórmula ha canviat el seu 'significat'. El problema és que al fer la substitució, la variable y

[3] totes les variables x_i alhora. Si, per exemple primer substituïm x_1 per t_1 i a continuació x_2 per t_2, aquesta segona substitució afectarà les substitucions de x_1 per t_1 en el cas en que t_1 conté la variable x_2.

del terme pel qual reemplacem x queda afectada per el quantificador $\forall y$. Un altre exemple d'aquest fenomen el podem observar amb la fórmula $A = \exists y \neg x \approx y$. Aquesta fórmula (té la variable lliure x) 'diu' que 'hi ha algun element diferent de x'. $A\{z/x\} = \exists y \neg z \approx y$ expressa que 'hi ha un element diferent de z', mentre que $A\{y/x\} = \exists y \neg y \approx y$ expressa l'existència d'un element diferent d'ell mateix (cosa totalment absurda). Un altre cop el 'significat' de la fórmula ha canviat radicalment perquè el quantificador $\exists y$ afecta la variable y. Quan això no passi, direm que 'la substitució és lliure':

Definició 2.2.14. Una substitució s *és lliure a una fórmula* A (per comoditat també es diu que As és lliure) si cap de les variables dels termes que hem substituït queda lligada per cap quantificador de A. Més precisament: cada vegada que reemplacem una aparició lliure de x_i a A per t_i, cap de les variables de t_i quedi afectada per cap quantificador de A.

Exemple 2.2.15. A l'exemple 2.2.13, la primera substitució és lliure (ja que A no té quantificadors) la segona també és lliure, mentre que la tercera no ho és: al substituir x per y ha quedat sota l'efecte del quantificador $\exists y$.

Observació 2.2.16. Hi ha dos casos importants en els quals la substitució As és lliure:

- Quan la fórmula A no té quantificadors,
- Quan tots els termes de la substitució s són tancats.

2.3 Semàntica de la LPO

2.3.1 Interpretacions

Bàsicament, per donar una interpretació en LPO caldrà donar un domini (l'univers on estan els individus), més la interpretació dels símbols no lògics en el domini: els símbols de constant s'interpreten com a element del domini, els símbols de funció s'interpreten com a operacions del domini i els símbols de relació s'interpreten com relacions concretes entre individus del domini. També cal, per raons tècniques, interpretar les variables com a elements del domini, encara que això només afectarà les variables lliures d'una fórmula.

Definició 2.3.1. Una *interpretació* I per a la LPO consta de:

- Un conjunt no buit D, que anomenarem domini.
- Per a cada símbol de constant a un element a^I del domini D (a^I és la *interpretació* de a).
- Per a cada símbol de funció n-ària f una operació n-ària f^I al domini D:

  $$f^I : D^n \to D$$

  (f^I és la *interpretació* de f).

- Per a cada símbol de relació n-ària R una relació n-ària R^I al domini D:

$$R^I \subseteq D^n$$

  (R^I és la *interpretació* de R).

- Per a cada variable x un element x^I del domini D (x^I és la *interpretació* de x).

Habitualment tenim una fórmula (o un conjunt finit de fórmules). En aquesta situació, no cal que I 'interpreti' tots els símbols del vocabulari, es suficient amb que interpreti tots el símbols que pareixen a les fórmules en qüestió. El conjunt format pels quatre primers ingredients de la interpretació (és a dir, sense la interpretació de les variables) rep el nom d'*estructura*.

2.3.2 Interpretació dels termes

Una interpretació I ens permet interpretar els termes (del mateix vocabulari que I) com a elements del domini D de manera natural:

Definició 2.3.2 (interpretació dels termes)**.** Si t és un terme i I una interpretació, definim $I(t)$, la *interpretació* del terme t així:

- $I(a) = a^I$ si a és un símbol de constant.
- $I(x) = x^I$ si x és una variable.
- $I(f(t_1, \ldots, t_n)) = f^I\,(I(t_1), \ldots, I(t_n))$ si $t_1, \ldots, t_n$ són termes i f és un símbol de funció d'arietat n.

Exemple 2.3.3. En el vocabulari format per $a, b, f/2, g/1$ considerem la interpretació I que té per domini $D := \{\alpha, \beta, \gamma\}$ i que interpreta els símbols així: $a^I := \alpha$, $b^I := \beta$, $c^I := \gamma$, $x^I = \alpha$, $y^I := \beta$, $z^I := \gamma$,

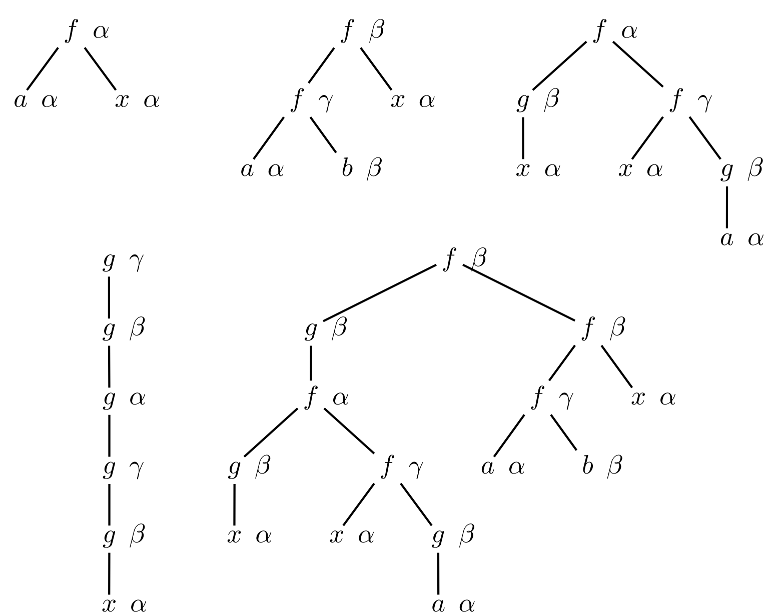
$$\begin{array}{cccc} g^I: & D & \to & D \\ & \alpha & \mapsto & \beta \\ & \beta & \mapsto & \gamma \\ & \gamma & \mapsto & \alpha \end{array}$$

$$\begin{array}{cccc} f^I: & D^2 & \to & D \\ & (\alpha,\alpha) & \mapsto & \alpha \\ & (\alpha,\beta) & \mapsto & \gamma \\ & (\alpha,\gamma) & \mapsto & \beta \\ & (\beta,\alpha) & \mapsto & \gamma \\ & (\beta,\beta) & \mapsto & \beta \\ & (\beta,\gamma) & \mapsto & \alpha \\ & (\gamma,\alpha) & \mapsto & \beta \\ & (\gamma,\beta) & \mapsto & \alpha \\ & (\gamma,\gamma) & \mapsto & \gamma. \end{array}$$

Anem a interpretar els termes de l'exemple 2.2.2: $f(a,x)$, $f(f(a,b),x)$, $f(g(x), f(x, g(a)))$, $g(g(g(g(g(x)))))$, $f(g(f(g(x), f(x, g(a)))), f(f(a,b),x))$.

- Comencem per el primer: $I(f(a,x)) = f^I(a^I, x^I) = f^I(\alpha,\alpha) = \alpha$.
- Si fem el segon, obtenim $I(f(f(a,b),x)) = f^I(f^I(a^I,b^I),x^I) = f^I(f^I(\alpha,\beta),\alpha) = f^I(\gamma,\alpha) = \beta$.
- Pel que fa al càlcul de la interpretació del tercer: $f(g(x), f(x, g(a))) = f^I(g^I(x^I), f^I(x^I, g^I(a^I))) = f^I(g^I(\alpha), f^I(\alpha, g^I(\alpha))) = f^I(\beta, f^I(\alpha,\beta)) = f^I(\beta,\gamma) = \alpha$.
- Si calculem el quart: $I(g(g(g(g(g(x)))))) = g^I(g^I(g^I(g^I(g^I(x^I))))) = g^I(g^I(g^I(g^I(g^I(\alpha))))) = g^I(g^I(g^I(g^I(\beta)))) = g^I(g^I(g^I(\gamma))) = g^I(g^I(\alpha)) = g^I(\beta) = \gamma$.
- Finalment, l'últim: $I(f(g(f(g(x), f(x, g(a)))), f(f(a,b),x))) = f^I(g^I(f^I(g^I(x^I), f^I(x^I, g^I(a^I)))), f^I(f^I(a^I,b^I),x^I)) = f^I(g^I(f^I(g^I(\alpha), f^I(\alpha, g^I(\alpha)))), f^I(f^I(\alpha,\beta),\alpha)) = f^I(g^I(f^I(\beta, f^I(\alpha,\beta))), f^I(\gamma,\alpha)) = f^I(g^I(f^I(\beta,\gamma)),\beta) = f^I(g^I(\alpha),\beta) = f^I(\beta,\beta) = \beta$.

Tots aquest càlculs es poden fer directament sobre l'arbre del terme. En els exemples anteriors:



2.3.3 Validesa d'una fórmula en una interpretació

A continuació anem a definir quan una fórmula A val en una interpretació I, i ho farem definint $I(A) \in \{0,1\}$, el valor de veritat de A a I. Això es fa de manera recursiva. Comencem pel cas atòmic.

Definició 2.3.4. • Si A és la fórmula $Pt_1 \dots t_n$, on P és un símbol de relació n-ària i $t_1, \dots, t_n$ són termes, llavors:

$$I(A) = \begin{cases} 1, & \text{si } (I(t_1), \dots, I(t_n)) \in P^I; \\ 0, & \text{altrament.} \end{cases}$$

- Si A és la fórmula $t_1 \approx t_2$, on t_1 i t_2 són termes, llavors:

$$I(A) = \begin{cases} 1, & \text{si } I(t_1) = I(t_2); \\ 0, & \text{altrament.} \end{cases}$$

Exemple 2.3.5. • $I(Pab) = 1$, ja que $(\alpha, \beta) \in P^I$.

- $I(Pyy) = 0$, ja que $(\beta, \beta) \notin P^I$.
- $I(f(x, a) \approx g(g(b))) = 1$, ja que $I(f(x, a)) = I(g(g(b))) = \alpha$.
- $I(g(f(x, y)) \approx b) = 0$, ja que $I(g(f(x, y))) = \alpha$ però $I(b) = \beta$.

La resta es fa de manera anàloga a la lògica proposicional llevat dels quantificadors. Per a aquests, necessitem canviar l'assignació d'una variable particular. Si I és una interpretació i α pertany al domini de I, $I_{[\alpha/x]}$ denota la mateixa interpretació, llevat que ara interpretem x com l'element α (en símbols: $x^{I_{[\alpha/x]}} = \alpha$).

Definició 2.3.6. • $I(\neg A) = \neg I(A)$,

- $I(A \wedge B) = I(A) \wedge I(B)$,
- $I(A \vee B) = I(A) \vee I(B)$,
- $I(A \to B) = I(A) \to I(B)$,
- $I(A \leftrightarrow B) = I(A) \leftrightarrow I(B)$,
- $I(\forall x A) = \begin{cases} 1, & \text{si per a tot } \alpha \in D \ I_{[\alpha/x]}(A) = 1; \\ 0, & \text{altrament,} \end{cases}$
- $I(\exists x A) = \begin{cases} 1, & \text{si per a algun } \alpha \in D \ I_{[\alpha/x]}(A) = 1; \\ 0, & \text{altrament.} \end{cases}$

Exemple 2.3.7 (continuació de 2.3.3). Ara ampliem el vocabulari de l'exemple 2.3.3 amb un parell de símbols de predicat $Q/1$ i $P/2$. Ara a la interpretació I de l'exemple 2.3.3 hi hem d'afegir la interpretació d'aquest dos nous símbols:

$$P^I := \{(\alpha, \alpha), (\alpha, \beta), (\alpha, \gamma)\} \qquad Q^I := \{\beta, \gamma\}.$$

Ara:

- $I(\exists y Pyy) = 1$ ja que $I_{[\alpha/y]}(Pyy) = 1$, doncs $(\alpha, \alpha) \in P^I$.
- $I(\forall x Pax) = 1$ ja que $I_{[\alpha/x]}(Pax) = I_{[\beta/x]}(Pax) = I_{[\gamma/x]}(Pax) = 1$, doncs $(\alpha, \alpha) \in P^I$, $(\beta, \beta) \in P^I$ i $(\gamma, \gamma) \in P^I$.

- $I(\exists x(Pxa \wedge Qx)) = 0$ ja que $I_{[\alpha/x]}(Pxa \wedge Qx) = I_{[\beta/x]}(Pxa \wedge Qx) = I_{[\gamma/x]}(Pxa \wedge Qx) = 0$. El primer perquè $I_{[\alpha/x]}(Qx) = 0$ ($\alpha \notin Q^I$), el segon i el tercer perquè $I_{[\beta/x]}(Pxa) = I_{[\gamma/x]}(Pxa) = 0$ doncs $(\beta, \alpha) \notin P^I$ i $(\gamma, \alpha) \notin P^I$.

Observació 2.3.8 (important). És clar que el valor $I(A)$ només depèn de la interpretació dels símbols no lògics que apareixen a la fórmula A. També es cert que només cal tenir definida la interpretació de les variables lliure de A, la resta resulta indiferent. En particular, en els enunciats, no cal tenir interpretades les variables. Així els enunciats descriuen propietats de les estructures (la seva valides depèn només de l'estructura), mentre que les fórmules amb variables lliures descriuen propietats 'd'elements' de les estructures. Per exemple, l'enunciat $\forall x \forall y \forall z\, (Rxx \wedge (Rxy \to Ryx) \wedge (Rxy \wedge Ryz \to Rxz))$ expressa que la relació R és simètrica, mentre que la fórmula $\forall y Rxy$ expressa que 'l'element x està relacionat amb tot-hom'.

Quan el domini D de la interpretació I és finit (i amb pocs elements per tal de que sigui eficient) hi ha una manera de reduir el càlcul del valor $I(A)$ a lògica proposicional (o a càlcul Booleà): el mètode d'expansions. Si $D = \{\alpha_1, \ldots, \alpha_n\}$, afegim $a_1, \ldots, a_n$ nous símbols de constant al vocabulari (expandim el llenguatge) amb el conveni $a_i^I = \alpha_i$ (expandim la interpretació I). Ara les regles per avaluar els quantificadors seran:

- $I(\forall x A) = I(A\,\{a_1/x_1\} \wedge \cdots \wedge A\,\{a_n/x_n\})$,
- $I(\exists x A) = I(A\,\{a_1/x_1\} \vee \cdots \vee A\,\{a_n/x_n\})$.

Exemple 2.3.9. Considerem la interpretació (per al vocabulari P^1, Q^1) amb domini $D = \{0, 1\}$ i les interpretacions de P i Q donades per $P^I = \{0\}$, $Q^I = \{\}$. Anem a comprovar que $I(\exists x(Px \to Qx)) = 1$ i que $I(\exists x Px \to \exists x Qx) = 0$ mitjançant expansions. Prenem dues constants a, b que s'interpretin com a 0 i 1 respectivament:

$$a^I = 0 \quad b^I = 1.$$

Si avaluem la primera tenim $I(\exists x(Px \to Qx)) = I((Pa \to Qa) \vee (Pb \to Qb)) = (1 \to 0) \vee (0 \to 0) = 0 \vee 1 = 1$, mentre que si avaluem la segona obtenim $I(\exists x Px \to \exists x Qx) = I((Pa \vee Pb) \to (Qa \vee Qb)) = (1 \vee 0) \to (0 \vee 0) = 1 \to 0 = 0$.

Exemple 2.3.10. Anem a fer l'exemple 2.3.7 amb expansions. Farem els tres últims casos i les fórmules $\forall x \exists y Pxy$, $\exists x \forall y Pxy$. Com que $D = \{\alpha, \beta, \gamma\}$ i ja tenim tres constants que s'interpreten com aquest elements, les usarem: $a^I = \alpha, b^I = \beta, c^I = \gamma$.

- $I(\exists y Pyy) = I(Paa \vee Pbb \vee Pcc) = I(Paa) \vee I(Pbb) \vee I(Pcc) = 1 \vee 0 \vee 0 = 1$.
- $I(\forall x Pax) = I(Paa \wedge Pab \wedge Pac) = I(Paa) \wedge I(Pab) \wedge I(Pac) = 1 \wedge 1 \wedge 1 = 1$.
- $I(\exists x(Pxa \wedge Qx)) = I((Paa \wedge Qa) \vee (Pba \wedge Qb) \vee (Pca \wedge Qc)) = I(Paa \wedge Qa) \vee I(Pba \wedge Qb) \vee I(Pca \wedge Qc) = (1 \wedge 0) \vee (0 \wedge 1) \vee (1 \wedge 0) = 0$.

- $I(\forall x\exists yPxy) = I(\exists yPay \wedge \exists yPby \wedge \exists yPcy) = I((Paa \vee Pab \vee Pac) \wedge (Pba \vee Pbb \vee Pbc) \wedge (Pca \vee Pcb \vee Pcc)) = (1 \vee 1 \vee 1) \wedge (0 \vee 0 \vee 0) \wedge (0 \vee 0 \vee 0) = 0.$
- $I(\exists x\forall yPxy) = I(\forall yPay \vee \forall yPby \vee \forall yPcy) = I((Paa \vee Pab \wedge Pac) \vee (Pba \wedge Pbb \wedge Pbc) \vee (Pca \wedge Pcb \wedge Pcc)) = (1 \wedge 1 \wedge 1) \vee (0 \wedge 0 \wedge 0) \vee (0 \wedge 0 \wedge 0) = 1.$

2.3.4 Satisfabilitat, validesa i conseqüència lògica

Un cop definit el valor de veritat d'una fórmula en una interpretació es defineixen els conceptes de satisfabilitat, validesa i conseqüència lògica de manera anàloga a com es fa a la lògica proposicional. També farem servir les mateixes notacions. Quan una fórmula A val a una interpretació I (és a dir, quan $I(A) = 1$) direm que I *és model de* A i ho indicarem així: $I \models A$. Un conjunt de fórmules Γ és *satisfactible* si existeix una interpretació I que fa certes totes les fórmules de Γ (la mateixa I per a totes les fórmules). A una tal interpretació en direm *model de* Γ. Una fórmula A és *conseqüència lògica* de Γ, en símbols: $\Gamma \models A$ sii tot model de Γ (tota interpretació que fa certes totes les fórmules de Γ) també és model de A. Una fórmula A és *vàlida* quan val a tota interpretació. També hi ha exactament les mateixes relacions entre 'satisfabilitat', 'conseqüència lògica' i 'validesa'; és a dir, valen les proposicions 1.2.11 i 1.2.16.

Exemple 2.3.11.

1. Veiem que $\{\forall xQx, \forall x(Qx \rightarrow \exists yPxy), \exists y\forall x\neg Pxy\}$ és satisfactible. Intuïtivament es veu que de les dues primeres fórmules s'en segueix (és conseqüència lògica) la fórmula $\forall x\exists yPxy$, per tant la interpretació que busquem també l'haurà de satisfer. Deixo com a exercici verificar que la interpretació següent
$$I := \begin{cases} D = \{0,1\}, \\ P^I = \{(0,0),(1,0)\}, \\ Q^I = \{0,1\}. \end{cases}$$
és model de les tres fórmules.

2. Veiem que $\{\exists xPx, \forall x(Px \rightarrow Qx)\} \not\models \forall xQx$. Cal trobar un model I de les dues primeres fórmules que no sigui model de $\forall xQx$. Necessitem que P^I tingui com a mínim un element: (el 0, per exemple) i que hi hagi algun element que no satisfaci Q (l'1, per exemple). Com que tots els elements que satisfan P també satisfan Q, necessitarem que el 0 satisfaci Q i que l'1 no satisfaci P. Això ens dona el model:
$$I := \begin{cases} D = \{0,1\}, \\ P^I = \{0\}, \\ Q^I = \{0\}. \end{cases}$$

3. Veiem que la fórmula $\forall x\exists yPxy$ no és vàlida. Necessitem una interpretació I que la faci falsa. N'hi ha prou, per exemple, amb fer que P^I sigui buit:
$$I := \begin{cases} D = \{0\}, \\ P^I = \{\}. \end{cases}$$

Proposició 2.3.12. *Si A és una fórmula, llavors:*

1. *A és vàlida sii $\forall xA$ és vàlida.*
2. *A és satisfactible sii $\exists xA$ és satisfactible.*

Demostració. Suposem que A és vàlida i sigui I una interpretació qualsevol. Com que A és vàlida $I_{[\alpha/x]}(A) = 1$ per a tot α del domini de I. Així $I(\forall xA) = 1$. Com que I era qualsevol $\forall xA$ és una fórmula vàlida. Recíprocament, si $\forall A$ és vàlida i I és una interpretació qualsevol, $I(\forall xA) = 1$ i per tant $I(A) = 1$ (ja que $I = I_{[x^I/x]}$). Com que I és qualsevol, tenim que A és vàlida. Això demostra *1*. La demostració de *2* la deixo com a exercici. □

Així, si quantifiquem universalment totes les variables lliures d'una fórmula A, n'obtenim una d'equivalent en quant a validesa i que no té variables lliure (és un enunciat). A aquest enunciat li direm clausura universal de A i el denotarem per $\forall A$. El mateix podem fer amb satisfabilitat: reemplaçant una fórmula A per la obtinguda quantificant existencialment totes les variables lliures de A, que anomenarem clausura existencial i denotarem $\exists A$ obtenim un enunciat equivalent en quant a satisfabilitat:

Corol·lari 2.3.13.
1. *La fórmula A és vàlida sii la seva clausura universal $\forall A$ és vàlida.*
2. *La fórmula A és satisfactible sii la seva clausura existencial $\exists A$ és satisfactible.*

Per exemple la fórmula $Pxy \to \exists yQf(z, y, z)$ és equivalent en validesa amb $\forall x\forall y\forall z(Pxy \to \exists yQf(z, y, z))$ i en quant a satisfabilitat amb $\exists x\exists y\exists z(Pxy \to \exists yQf(z, y, z))$.