Capítol 1

Llenguatges de Dyck

1.1 Definicions

Siguin $A_n = \{a_1, \ldots, a_n\}$ i $B_n = \{b_1, \ldots, b_n\}$ dos alfabets disjunts del mateix nombre n de símbols.
1.1 Anomenem *llenguatge de Dyck* de n parells de símbols, i el representem per D_n^*, el conjunt de mots generats per la gramàtica que té per produccions

$$S \to \lambda \mid SS \mid a_i S b_i \mid b_i S a_i \quad \text{per a cada } i \;\; 1 \leqslant i \leqslant n.$$

Es tracta de mots que presenten ben aparionats els símbols dels alfabets A_n i B_n, de manera que no hi ha imbricacions encavalcades; així, el mot $b_1a_2b_2a_1$ és un mot ben construït; en canvi, el mot $a_1a_2b_1b_2$ no ho és.
1.2 Anomenem *llenguatge de Dyck restringit* de n parells de símbols, i el representem per ${\mathsf{D}'_n}^*$, el conjunt de mots generats per la gramàtica que té per produccions

$$S \to \lambda \mid SS \mid a_i S b_i \quad \text{per a cada } i \; 1 \leqslant i \leqslant n.$$

Es tracta d'interpretar els n parells (a_i, b_i) com a diferents parells de parèntesis en correspondència, i prendre en consideració els mots ben parentitzats.

1.2 Gramàtiques no ambigües

Les gramàtiques que hem utilitzat per definir aquests llenguatges tenen l'avantatge de posar de manifest les propietats que tenen els mots que generen. També (degut a la inclusió de les produccions $S \to \lambda \mid SS$) el de fer evident que els dos llenguatges considerats són monoides.

Però en canvi, tenen l'inconvenient (precisament per l'existència d'aquestes produccions) de ser ambigües. Veurem un procediment sistemàtic per construir gramàtiques no ambigües per a aquests llenguatges i per a d'altres d'associats.

1.2.1 Codis i factors primers

Recordem que s'anomena *monoide lliure* aquell en què tot element diferent del neutre admet una única descomposició en factors primers. Donat un alfabet Σ, una part $C \subseteq \Sigma^*$ s'anomena *codi* si genera lliurement C^*, és a dir, si tot element de C^+ factoritza de manera única en mots de C.

Per exemple, $\{aa, ba, baa\}$ i ba^* són codis, en canvi $\{a, ab, ba\}$ no ho és.

Podem construir una gramàtica no ambigua del monoide generat per un codi a partir de qualsevol gramàtica no ambigua que generi el codi. Es tracta de fer, senzillament, la construcció canònica de l'estrella del codi. En el cas del primer exemple la gramàtica és

$$S \to\ aaS \mid baS \mid baaS \mid \lambda,$$

mentre que en el cas del segon exemple és

$$S \to\ CS \mid \lambda \qquad C \to\ bA \qquad A \to\ aA \mid \lambda.$$

En canvi, si fem la construcció de l'estrella per obtenir el monoide generat pel conjunt $\{a, ab, ba\}$, obtenim la gramàtica ambigua

$$S \to\ aS \mid abS \mid baS \mid \lambda.$$

És clar que podem construir fàcilment (via autòmat finit determinista) una gramàtica no ambigua per a aquest monoide. Es tracta de

$$S \to\ aB \mid baS \mid \lambda \qquad B \to\ aB \mid bS \mid \lambda,$$

però aquest procediment no pot generalitzar-se al cas en què els codis no siguin llenguatges regulars.

En tot cas, si bé és fàcil demostrar que el conjunt de factors primers d'un monoide qualsevol sobre un alfabet genera el monoide considerat, aquesta generacció pot ser ambigua quan el monoide no és lliure, és a dir, quan el conjunt de factors primers no és un codi.

En el cas dels llenguatgesw de Dyck, els factors primers són codis. Així doncs, la construcció de gramàtiques no ambigües per a aquests llenguatges pot basar-se en les gramàtiques que generen els seus conjunts de factors primers.

1.2.2 Factors primers dels llenguatges de Dyck

Representarem per D_n i D'_n els conjunts de factors primers dels llenguatges de Dyck i dels llenguatges de Dyck restringits respectivament. Aquesta notació és coherent amb el fet d'haver designat per D_n^* i ${\mathsf{D}'_n}^*$ els llenguatges originals. En endavant, ens referirem principalment als llenguatges de Dyck sobre un parell de símbols (parèntesis) que designarem per a i b. En aquest cas, en lloc dels noms D'_1 i ${\mathsf{D}'_1}^*$ utilitzarem els D'_{ab} i ${\mathsf{D}'_{ab}}^*$ i introduirem també els noms D'_{ba} i ${\mathsf{D}'_{ba}}^*$ per representar els revessats dels llenguatges anteriors. Aquesta notació és consistent, perquè el revessat de l'estrella d'un llenguatge donat és l'estrella del revessat del llenguatge considerat. En el cas $n = 1$, els mots de D_1^* queden caracteritzats pel fet de tenir el mateix nombre de a's i de b's, sense que aquesta caracterització tan senzilla tingui equivalent en els casos de $n > 1$. Deixem com a exercici la demostració d'aquest fet.

Teorema 1 $\mathsf{D}_1^* = \{w \in \{a,b\}^* \ \mid \ |w|_a = |w|_b\}$.

També deixem com a exercici la caracterització següent de ${\mathsf{D}'_{ab}}^*$.

Teorema 2 ${\mathsf{D}'_{ab}}^* = \{w \in \{a,b\}^* \mid |w|_a = |w|_b \ \wedge \ \forall x \in \mathsf{prefixos}(w) \ |x|_a \geqslant |x|_b\}$.

Com dèiem, basarem la construcció de gramàtiques no ambigües dels llenguatges de Dyck i dels seus derivats en les gramàtiques que generen els factors primers. Començarem, doncs, estudiant aquests conjunts.

Teorema 3 $\mathsf{D}'_{ab} = \{w \in \{a,b\}^+ \mid |w|_a = |w|_b \ \wedge \ \forall x \in \mathsf{prefixos\text{-}propis}(w) \ |x|_a > |x|_b\}$.

Per aclarir l'exposició del que segueix designarem per L el llenguatge ${\mathsf{D}'_{ab}}^*$; així evitarem donar per suposat que L és el tancament de Kleene de D'_{ab}. Partim de la definició de D'_{ab} com a conjunt de factors primers de L, és a dir

$$\mathsf{D}'_{ab} = \{w \in \{a,b\}^* \ \mid \ w \in L - \{\lambda\} \ \wedge \ \forall x, y \in L - \{\lambda\} \ w \neq xy\}.$$

Vegem primerament la inclusió d'esquerra a dreta. Sigui $w \in \mathsf{D}'_{ab}$. Hem de demostrar que $|w|_a = |w|_b$ i que $\forall x \in \mathsf{prefixos\text{-}propis}(w) \ |x|_a > |x|_b$. Partim del fet que $w \in L$ i que $\forall x, y \in L - \{\lambda\} \ w \neq xy$. La primera condició ens garanteix que $|w|_a = |w|_b$. Suposem que existís un prefix propi x de w que no satisfés $|x|_a > |x|_b$. Com que $w \in L$, es tindria $|x|_a = |x|_b$. En tal cas, sigui y tal que $w = xy$. Com que x és propi, $y \neq \lambda$. Tindríem, per una part, que

$$|y|_a = |w|_a - |x|_a = |w|_b - |x|_b = |y|_b$$

i per altra part, per a tot z que fos prefix de y, es tindria $|z|_a = |xz|_a - |x|_a$. Però $|xz|_a \geqslant |xz|_b$, per ser xz prefix de w; i $|x|_a = |x|_b$ per la hipòtesi contradictanda. Tindríem doncs que $|z|_a \geqslant |z|_b$. Resultaria que $y \in L - \{\lambda\}$, contradient que w fos primer.

Vegem, en segon lloc, la inclusió de dreta a esquerra. Per una part, és obvi que tot mot que satisfà les condicions del membre de la dreta pertany a $L - \{\lambda\}$. Per altra part, qualsevol mot x que satisfaci les condicions $w = xy$ amb $x, y \neq \lambda$ és un prefix propi de w. Per hipòtesi, doncs, es té $|x|_a > |x|_b$ i, per tant, $x \notin L$.

Teorema 4 $\mathsf{D}'_{ab} = a{\mathsf{D}'_{ab}}^*b$ *i, simètricament,* $\mathsf{D}'_{ba} = b{\mathsf{D}'_{ba}}^*a$.

Demostrarem només el primer enunciat. L'altre és simètric. Per la mateixa raó del teorema anterior, designarem per L el llenguatge ${\mathsf{D}'_{ab}}^*$.

Vegem primerament la inclusió d'esquerra a dreta. Sigui $w \in \mathsf{D}'_{ab}$. Necessàriament w és de la forma $aw'b$ (tots els mots de L ho són) i $|w'|_a = |w'|_b$. Només queda per veure doncs que tot prefix de w' té almenys tantes a's com b's. Sigui x un prefix qualsevol de w'. Tenim que $|x|_a = |ax|_a - 1$. Però ax és un prefix propi de w i satisfà la condició $|ax|_a > |ax|_b = |x|_b$. Per tant $|x|_a \geqslant |x|_b$.

Vegem, en segon lloc, la inclusió de dreta a esquerra. Sigui $w \in aLb$. w és de la forma $w = aw'b$ on $w' \in L$. És clar que $|w|_a = |w|_b$, perquè $|w'|_a = |w'|_b$. Sigui x un prefix propi de w. x ha de ser de la forma $x = ax'$ on x' és un prefix (que pot ser no propi) de w'. Es té $|x|_a = 1 + |x'|_a \geqslant 1 + |x'|_b > |x'|_b$.

Teorema 5 $\mathsf{D}_1 = \mathsf{D}'_{ab} \cup \mathsf{D}'_{ba}$.

La demostració es basa en el fet que si $w \in \mathsf{D}_1$, o bé tots els prefixos propis de w tenen més a's que b's, o bé tots tenen més b's que a's.

Teorema 6 *Els conjunts* D'_{ab} , D'_{ba} *i* D_1 *són codis.*

En els tres casos de monoides generats per aquests conjunts, la descomposició d'un mot qualsevol en producte de factors és única. Aquesta propietat és una conseqüència immediata del fet que els tres conjunts considerats tenen la *propietat del prefix* que significa que no hi ha cap parell de mots en el conjunt considerat que siguin l'un prefix propi de l'altre. (Convé advertir que aquesta propietat és suficient però no necessària per tal que un conjunt sigui codi. Un exemple de codi que no la satisfà és el conjunt ba^*.)

1.2.3 Gramàtiques no ambigües per a ${\mathsf{D}'_{ab}}^*$, D_1^*, D'_{ab}, D_1 i Ł_{ab}

En lloc d'utilitzar, com és habitual, el símbol S per indicar la variable inicial de les gramàtiques, farem servir símbols diferents per a cada gramàtica. Això ens permetrà combinar les produccions de totes elles. A més dels llenguatges definits fins ara, donem també símbols per als llenguatges de Lukasiewicz (Ł_{ab} i Ł_{ba}) que definirem més endavant. Utilitzarem els símbols següents per a les gramàtiques considerades.

$$Z\colon\ \mathsf{D}_1^* \quad X\colon\ {\mathsf{D}'_{ab}}^* \quad Y\colon\ {\mathsf{D}'_{ba}}^* \quad D\colon\ \mathsf{D}'_{ab} \quad E\colon\ \mathsf{D}'_{ba} \quad F\colon\ \mathsf{D}_1 \quad B\colon\ \text{Ł}_{ab} \quad A\colon\ \text{Ł}_{ba}$$

El punt de partida és l'equació fonamental $\mathsf{D}'_{ab} = a{\mathsf{D}'_{ab}}^*b$, que ens permet definir conjuntament les produccions que generen D'_{ab} i ${\mathsf{D}'_{ab}}^*$. Es tracta de fer

$$X \to\ DX \mid \lambda \qquad D \to\ aXb.$$

Com que la variable D no presenta recursivitat, podem substituir-la en el costat dret de la producció de X. Això ens permet donar per a ${\mathsf{D}'_{ab}}^*$ la gramàtica més compacta

$$X \to\ aXbX \mid \lambda.$$

De manera anàloga, obtenim per a ${\mathsf{D}'_{ba}}^*$ i per a D'_{ba} les gramàtiques

$$Y \to\ bYaY \mid \lambda \qquad E \to\ bYa.$$

La gramàtica de D_1^* s'obté de manera anàloga a partir de l'estrella de D_1 i aquesta s'obté de la reunió $\mathsf{D}_1 = \mathsf{D}'_{ab} \cup \mathsf{D}'_{ba}$. Resulta

$$Z \to\ FZ \mid \lambda \qquad F \to\ D \mid E,$$

on no s'han repetit les produccions corresponents a D, E ni les corresponents a X i Y que hi estan necessàriament associades. De nou podem donar una versió més simplificada de la gramàtica de D_1^* sense més que substituir les variables D, E i F (que no són recursives) pels costats drets de les seves produccions. Otenim

$$Z \to\ aXbZ \mid bYaZ \mid \lambda \qquad X \to\ aXbX \mid \lambda \qquad Y \to\ bYaY \mid \lambda.$$

Podem obtenir diverses variants d'aqueste gramàtiques utilitzant noves variables que substitueixen les donades. Si fem $B = Xb$ i $A = Ya$, és immediat comprovar que resulta $D = aB$ i $E = bA$. L'avantatge d'aquestes dues noves variables A i B és que els llenguatges que generen poden ser descrits per dues gramàtiques formades per una sola variable. Es tracta dels llenguatges de Lukasiewicz,

$$\text{Ł}_{ab}: \quad B \to\ aBB \mid b \qquad \text{Ł}_{ba}: \quad A \to\ bAA \mid a.$$

La introducció d'aquestes variables permet transformar la gramàtica de D_1^* així:

$$Z \to\ aBZ \mid bAZ \mid \lambda \qquad B \to\ aBB \mid b \qquad A \to\ bAA \mid a.$$

Com es dedueix fàcilment de la mateixa estructura de les gramàtiques associades a B i A, els mots que generen corresponen a les codificacions dels arbres binaris. En el cas de B es tracta d'arbres en què el símbol a correspon a nodes interns i el símbol b a fulles, mentre que en el cas de A és a l'inrevés. En tots els casos les codificacions consisteixen en els recorreguts en preordre dels arbres. La introducció dels llenguatges de Lukasiewicz ens permet escriure les següents igualtats que es deriven de les relacions anteriors:

$$\mathsf{D}'_{ab} = a\text{Ł}_{ab} \qquad \mathsf{D}'_{ba} = a\text{Ł}_{ba} \qquad \text{Ł}_{ab} = {\mathsf{D}'_{ab}}^* \cdot b \qquad \text{Ł}_{ba} = {\mathsf{D}'_{ba}}^* \cdot a.$$

1.3 Llenguatges derivats dels de Dyck

1.3.1 Mots en què tot prefix té almenys tantes a's com b's

Aquest llenguatge correspon a la sintaxi de combinar les instruccions comuns d'alguns llenguatges de programació "if...then..." i "if...then...else...". Formalment és

$$L_1 = \{w \in \{a,b\} \mid \forall x \in \mathsf{prefixos}(w)\ |w|_a \geqslant |w|_b\}.$$

Es tracta d'un monoide. És fàcil deduir que els seus factors primers són $\mathsf{D}'_{ab} \cup \{a\}$. Es té $L_1 = (\mathsf{D}'_{ab} \cup \{a\})^*$ Utilitzarem la lletra G com a variable inicial de la gramàtica que genera L_1. Una gramàtica no ambigua s'obté directament per tancament de Kleene dels factors primers

$$G \to\ DG \mid aG \mid \lambda \qquad D \to\ aXb \qquad X \to\ aXbX \mid \lambda$$

o alternativament, utilitzant els canvis ja coneguts,

$$G \to\ aXbG \mid aG \mid \lambda \qquad X \to\ aXbX \mid \lambda$$

o encara

$$G \to\ aBG \mid aG \mid \lambda \qquad B \to\ aBB \mid b.$$

1.3.2 Mots que tenen almenys tantes a's com b's

Es tracta del llenguatge

$$L_2 = \{w \in \{a,b\}^* \mid |w|_a \geqslant |w|_b\}.$$

També en aquest cas tenim que $L_2 = L_2^*$, és a dir que L_2 també és un monoide. Però ara els factors primers no constitueixen un codi. En efecte, els factors primers són $\mathsf{D}_1 \cup \{a\}$, que no permeten una factorització única, com es veu, per exemple, amb $a \cdot ba = ab \cdot a$.

Per tal de generar de manera no ambigua L_2 tenim diferents alternatives:

- $L_2 = \mathsf{D}_1^* \cdot (\{\lambda\} \cup a \cdot L_1)$. Es tracta de descompondre cada mot de L_2 en el màxim prefix que té tantes a's com b's (com a mínim és λ) seguit d'un sufix que, si no és λ, forçosament comença per a i a continuació ve un mot que no pot contenir cap prefix amb més b's que a's. Una gramàtica que tradueix aquesta descomposició és

$$H \to\ aBH \mid bAH \mid aG \mid \lambda \quad B \to\ aBB \mid b \quad A \to\ bAA \mid a \quad G \to\ aBG \mid aG \mid \lambda.$$

- $L_2 = \mathsf{D}_1^* \cdot \left(a{\mathsf{D}'_{ab}}^*\right)^*$. Es tracta d'una variant de la descomposició anterior que dóna origen a la gramàtica

$$H \to\ aXbH \mid bYaH \mid R \quad R \to\ aXR \mid \lambda \quad X \to\ aXbX \mid \lambda \quad Y \to\ bYaY \mid \lambda,$$

  on R és la variable que genera $\left(a{\mathsf{D}'_{ab}}^*\right)^*$, que és el mateix llenguatge que $\{\lambda\} \cup a \cdot L_1$.

- $L_2 = \text{Ł}_{ba}^* \cdot \mathsf{D}_1^*$. Es tracta de considerar el màxim sufix que té tantes a's com b's i descompondre el prefix corresponent en factors que tenen cada un una a més que b's, sense que cap prefix d'aquests factors tingui més a's que b's. Podem associar dues gramàtiques a aquesta descomposició, depenent de quin dels dos llenguatges D_1^* o Ł_{ba}^* generem primer. Si comencem la generació per l'esquerra, la gramàtica és

$$H \to\ AH \mid Z \qquad Z \to\ aBZ \mid bAZ \mid \lambda \qquad A \to\ bAA \mid a \qquad B \to\ aBB \mid b.$$

  En canvi, si comencem la generació per la dreta (per D_1^*), resulta

$$H \to\ HaB \mid HbA \mid M \qquad A \to\ bAA \mid a \qquad B \to\ aBB \mid b \qquad M \to\ AM \mid \lambda,$$

  on la variable M genera Ł_{ba}^*.