

Otomata Teorisi Bölüm 4

Bağlamdan Bağımsız Dilbilgisi –Tür 2

4.BAĞLAM-DAN-BAĞIMSIZ DİL BİLGİSİ

- Programlama dilleri, derleyiciler, yorumlayıcılar, sözdizim çözümleyiciler, aritmetik deyimler, ..vb. birçok yazılım bileşeninin bünyesinde yer aldığı için bağlamdan-bağımsız dilbilgisi ve diller ile bu dilleri tanıyan makine modeli bilgisayar bilimleri ve mühendisliği açısından önem taşır.
- Bağlamdan bağımsız dilbilgisi bir dörtlüdür.

V_N = Sözdizim değişkenleri kümesi (sonlu bir küme)

V_T = Uç simgeler kümesi (sonlu bir küme)

V_N ve V_T ayrık kümelerdir. $V_N \cap V_T = \emptyset$

S: Başlangıç değişkeni $S \in V_N$

P: Yeniden yazma ya da türetme kuralları

$$A \Rightarrow b \quad A \in V_N \quad b \in V^*$$

Örnek 4.1

$$G_{4.1} = \langle V_N, V_T, P, S \rangle$$

$$V_N = \{ S \}$$

$$V_T = \{ +, -, *, /, (,), v, c \}$$

$$P: S \Rightarrow S+S \mid S-S \mid S*S \mid S/S \mid (S) \mid v \mid c$$

$G_{4.1}$ tarafından türetilen tümcelerden biri bulunmak istendiğinde:

$$S \Rightarrow S*S \Rightarrow S*(S) \Rightarrow S*(S-S) \Rightarrow v*(S-S) \Rightarrow v*(v-S) \Rightarrow v*(v-c)$$

$$\begin{aligned} S &\Rightarrow S*S \Rightarrow (S)*S \Rightarrow (S)*(S) \Rightarrow (S+S)*(S) \Rightarrow (S+S)*(S-S) \Rightarrow (S+S)*(S-S/S) \\ &\Rightarrow (v+S)*(S-S/S) \Rightarrow (v+c)*(S-S/S) \Rightarrow (v+c)*(v-S/S) \Rightarrow (v+c)*(v-v/S) \\ &\Rightarrow (v+c)*(v-v/c) \end{aligned}$$

$$\begin{aligned} S &\Rightarrow S+S \Rightarrow S*S + S \Rightarrow (S)*S+S \Rightarrow (S/S)*S+S \Rightarrow (S/(S))*S+S \\ &\Rightarrow (S/(S-S))*S+S \Rightarrow (v/(S-S))*S+S \Rightarrow (v/(v-S))*S+S \Rightarrow (v/(v-c))*S+S \\ &\Rightarrow \Rightarrow (v/(v-c))*v+S \Rightarrow (v/(v-c))*v+c \end{aligned}$$

Örnek 4.2

$G_{4.2} = \langle V_N, V_T, P, S \rangle$

$V_N = \{ S, A, B \}$

$V_T = \{ a, b, c, d \}$

P: $S \Rightarrow aAdd$

$A \Rightarrow aAd \mid Ad \mid bBcc$

$B \Rightarrow bBc \mid Bc \mid \lambda$

$G_{4.2}$ tarafından türetilen tümcelerden biri bulunmak istendiğinde:

$S \Rightarrow aAdd \Rightarrow abBccdd \Rightarrow abccdd$

$S \Rightarrow aAdd \Rightarrow aaAddd \Rightarrow aabBccddd \Rightarrow aabccddd$

$S \Rightarrow aAdd \Rightarrow aaAddd \Rightarrow aaAddddd \Rightarrow aaaAdddddd \Rightarrow aaabBccddddd \Rightarrow$
 $aaabBccccddddd \Rightarrow aaabbccccddddd$

Yukarıdaki örneklerden bu bağlamdan bağımsız dilbilgisi tarafından $L(G_{4.2})$ dili aşağıdaki gibi tanımlanabilir:

$L(G_{4.2}) = \{ a^n b^m c^k d^p \mid n, m, p, k \geq 1, p > n, k > m \}$

Örnek 4.3

$G_{4.3} = \langle V_N, V_T, P, S \rangle$

$V_N = \{ S, X, Y, Z \}$

$V_T = \{ a, b, c \}$

P: $S \Rightarrow XY$

$X \Rightarrow aXbb \mid aZbb \mid abb$

$Y \Rightarrow cY \mid c$

$Z \Rightarrow Zb \mid Xb$

$G_{4.3}$ tarafından türetilen tümcelerden biri bulunmak istendiğinde:

$S \Rightarrow XY \Rightarrow abbY \Rightarrow abbcY \Rightarrow abbcc$

$S \Rightarrow XY \Rightarrow aXbbY \Rightarrow aaZbbbbY \Rightarrow aaZbbbbbY \Rightarrow aaXbbbbbbY \Rightarrow aaabbbbbbbY$
 $\Rightarrow aaabbbbbbbbc$

Yukarıdaki örneklerden bu bağlamdan bağımsız dilbilgisi tarafından $L(G_{4.2})$ dili aşağıdaki gibi tanımlanabilir:

$L(G_{4.3}) = \{ a^n b^m c^k \mid n \geq 1, k \geq 1, m \geq 2n \}$

Örnek 4.4

$G_{4.4} = \langle V_N, V_T, P, S \rangle$

$V_N = \{ S, A \}$

$V_T = \{ 0, 1 \}$

P: $S \Rightarrow 0S0 \mid 1S1 \mid 0A0 \mid 1A1$

$A \Rightarrow 0A1 \mid 1A0 \mid 01 \mid 10$

$G_{4.4}$ tarafından türetilen tümcelerden biri bulunmak istendiğinde:

$S \Rightarrow 0S0 \Rightarrow 01A10 \Rightarrow 010A110 \Rightarrow 01010110$

$S \Rightarrow 0S0 \Rightarrow 01S10 \Rightarrow 011S110 \Rightarrow 0110A1110 \Rightarrow 01100A11110 \Rightarrow 011001011110$

$S \Rightarrow 0S0 \Rightarrow 00S00 \Rightarrow 000S000 \Rightarrow 0001A1000 \Rightarrow 00010A11000 \Rightarrow 000100A111000 \Rightarrow 0001000A1111000 \Rightarrow 00010001011111000$

Yukarıdaki örneklerden bu bağlamdan bağımsız dilbilgisi tarafından $L(G_{4.2})$ dili aşağıdaki gibi tanımlanabilir:

$L(G_{4.3}) = \{ uv(v')^R u^R \mid u, v \in (0+1)^*, |u| \geq 1, |v| \geq 1 \}$

u^R u 'nın tersi(reverse)

$(v')^R$ V 'nin tümlerinin tersi

4.2 Türetme Ağacı

- Bir dilbilgisi tarafından türetilen tümcesel yapı ve tümceler aşağıdaki gibi gösterilebilir:

$$S \Rightarrow \beta_1 \Rightarrow \beta_2 \Rightarrow \beta_3 \dots \dots \dots \Rightarrow \beta_{n-1} \Rightarrow \beta_n \Rightarrow w \quad \beta_i \in V^*, \quad w \in V_T^*$$

- Bağlamdan-bağımsız dillerin her tümcesel yapısına ve her tümcesine bir **türetme ya da ayrıştırma ağacı (derivation or parsing tree)** karşı gelir.

Örneğin $G_{4.4}$ tarafından türetilen:

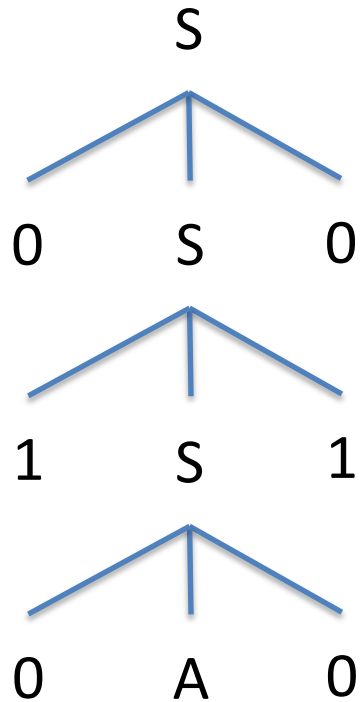
β = 010A010 tümcesel yapısı ile

w = 0010100100 tümcesine

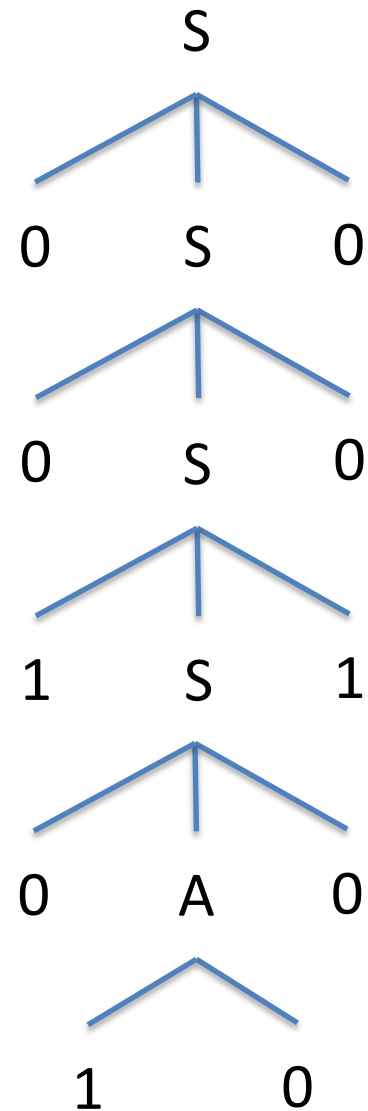
karşı gelen türetme ağaçları Çizim 4.1'de görülmektedir.

Türetme Ağacı: Çizim 4.1

a)



b)



Türetme ya da Ayırıştırma Ağacının Tanımı

- a) Ağacın kökünün etiketi S' 'dir.
- b) Kök dışındaki ara düğümlerin etiketleri sözdizim değişkenleridir ($A \in V_N$).
- c) Eğer ağaç bir tümcesel yapıya karşı geliyorsa, yaprakların etiketleri sözdizim değişkenleri ya da uç simgeler olabilir ($X \in V$). Eğer ağaç bir tümceye karşı geliyorsa yaprakların etiketleri yalnız uç simgeler ($a \in V_T$) olabilir.
- d) Eğer ara bir düğümün etiketi A , bu ara düğümün hemen altındaki düğümlerin etiketleri ise soldan sağa $X_1, X_2, X_3, \dots, X_k$ ise, dilbilgisinin yeniden yazma kuralları arasında
$$A = X_1 X_2 X_3 \dots X_k$$
$$X_1, X_2, X_3, \dots, X_k \in V$$
 kuralı yer almalıdır.
- e) Eğer bir düğümün etiketi λ ise, bu düğüm bir uç düğüm(yaprak) olmalı ve bu düğümün kardeşi bulunmamalıdır.

Soldan ve Sağdan Türetme

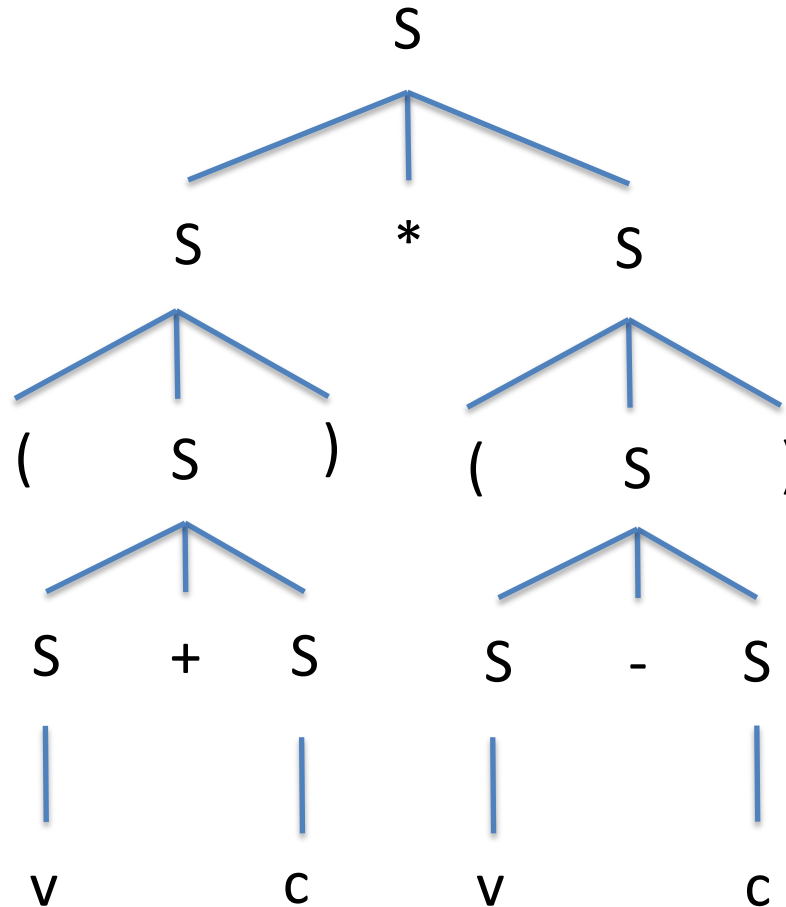
- Yapraklarının etiketleri uç simgeler olan her türetme ağacına dilin bir tümcesi karşı gelir. Eğer dil **belirgin (unambiguous)** bir dil ise de, dilin her tümcesine bir türetme ağacı karşı gelir. Eğer dilin bir tümcesine birden çok türetme ağacı karşı geliyorsa, dil **belirgin olmayan (ambiguous)** bir dildir. Belirginlik dilbilgisinin ve dilin en temel özelliklerinden biridir.
- Tümceleri birden çok anlam taşıyan belirgin olmayan dillerin uygulamada hiçbir değeri yoktur. Bu nedenle, kullanılan dillerin tümünün belirginlik özelliğini sağladığı varsayılacaktır. Buna göre dilin her tümcesine bir türetme ağacı, yapraklarının etiketleri uç simgeler olan her türetme ağacına da bir tümce karşı geldiğini söyleyebiliriz.
- Dilin her tümcesine bir türetme ağacı karşı geldiği gibi, bir soldan türetme, bir de sağdan türetme karşı gelir.

Soldan veya Sağdan Türetme

- Eğer bir tümce türetilirken, her adımda en soldaki değişkene bir türetme uygulanıyorsa, yapılan türetmeye **soldan türetme (leftmost derivation)** denir. Soldan türetmenin her ara adımında, eğer tümcesel yapı birden çok değişken içeriyorsa, öncelik en soldaki değişkene verilir.
- Eğer bir tümce türetilirken, her adımda en sağdaki değişkene bir türetme uygulanıyorsa, yapılan türetmeye **sağdan türetme (rightmost derivation)** denir. Sağdan türetmenin her ara adımında, eğer tümcesel yapı birden çok değişken içeriyorsa, öncelik en sağdaki değişkene verilir.

Türetme ağacı:

$$w_1 = (v+c) * (v-c)$$



$w_1 = (v+c) * (v-c)$ soldan ve sağdan türetme

a) w_1 'in soldan türetilmesi:

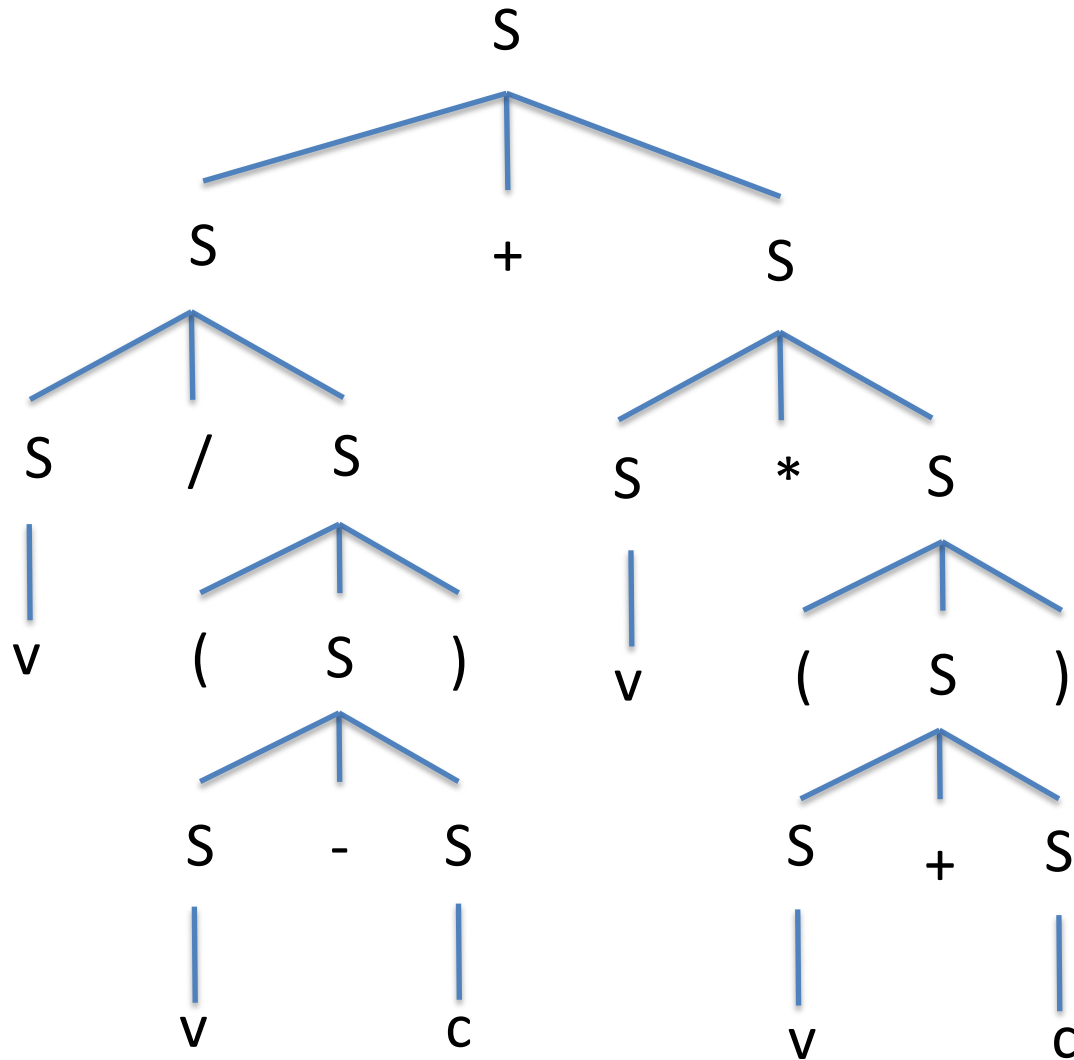
$$\begin{aligned} S &\Rightarrow S * S \\ &\Rightarrow (S) * S \\ &\Rightarrow (S+S) * S \\ &\Rightarrow (v+S) * S \\ &\Rightarrow (v+c) * S \\ &\Rightarrow (v+c) * (S) \\ &\Rightarrow (v+c) * (S-S) \\ &\Rightarrow (v+c) * (v-S) \\ &\Rightarrow (v+c) * (v-c) \end{aligned}$$

b) w_1 'in sağdan türetilmesi:

$$\begin{aligned} S &\Rightarrow S * S \\ &\Rightarrow S * (S) \\ &\Rightarrow S * (S-S) \\ &\Rightarrow S * (S-c) \\ &\Rightarrow S * (v-c) \\ &\Rightarrow (S) * (v-c) \\ &\Rightarrow (S+S) * (v-c) \\ &\Rightarrow (S+c) * (v-c) \Rightarrow (v+c)(v-c) \end{aligned}$$

Türetme ağacı:

$$w_2 = v / (v - c) + v * (v + c)$$



$w_2 = v/(v-c) + v^*(v+c)$ soldan ve sağdan türetme

a) w_2 'in soldan türetilmesi:

$$S \Rightarrow S+S$$

$$\Rightarrow S/S+S$$

$$\Rightarrow v/S*S$$

$$\Rightarrow v/(S)+S$$

$$\Rightarrow v/(S-S)+S$$

$$\Rightarrow v/(v-S)+S$$

$$\Rightarrow v/(v-c)+S$$

$$\Rightarrow v/(v-c)+S*S$$

$$\Rightarrow v/(v-c)+v*S$$

$$\Rightarrow v/(v-c)+v*(S)$$

$$\Rightarrow v/(v-c)+v*(S+S)$$

$$\Rightarrow v/(v-c)+v*(v+S)$$

$$\Rightarrow v/(v-c)+v*(v+c)$$

b) w_2 'in sağdan türetilmesi:

$$S \Rightarrow S+S$$

$$\Rightarrow S+S*S$$

$$\Rightarrow S+S*(S)$$

$$\Rightarrow S+S*(S+S)$$

$$\Rightarrow S+S*(S+c)$$

$$\Rightarrow S+S*(v+c)$$

$$\Rightarrow S+v*(v+c)$$

$$\Rightarrow S/S + v*(v+c)$$

$$\Rightarrow S/(S-S) + v*(v+c)$$

$$\Rightarrow S/(S-c) + v*(v+c)$$

$$\Rightarrow S/(v-c) + v*(v+c)$$

$$\Rightarrow v/(v-c) + v*(v+c)$$

4.3 Dilbilgisinin Yalınlaştırılması

- Bağlamdan bağımsız bir dilbilgisi verildiğinde, bu dilbilgisinin daha az sayıda değişken ve kural içeren ve belirli biçimdeki kuralları içermeyen bir dilbilgisine dönüştürülmesine dilbilgisinin yalınlaştırılması denilir.
- **4.3.4. Yararsız Değişken, Simge ve Kurallar**
- Eğer bir dilbilginin uç simgelerinden biri, bu dilbilgisi tarafından türetilen tümcelerden hiçbirinde yer almıyorsa, bu uç simgeye **yararsız simge (useless symbol)** denilir.
- Eğer bir dilbilgisinin değişkenlerinden biri (**A**), en az bir tümce türetilirken elde edilen tümcesel yapıların en az birinde yer alıyorsa
- $S \Rightarrow \alpha_1 A \alpha_2 \Rightarrow w$
- Bu değişken **yararlı (usefull)** bir değişkendir. Bu özelliği taşımayan değişkenlere ise **yararsız değişken (useless variable)** denilir. Yararsız değişken hiçbir tümcenin türetilmesinde kullanılmayan; kullanıldığı türetmelerin hiçbirinin sonunda ise bir tümce elde edilemeyen değişkendir.

4.3.4. Yararsız Değişken, Simge ve Kurallar

- Eğer bir dilbilgisinin yeniden yazma kurallarından biri ($A \Rightarrow \beta$), en az bir tümcenin türetilmesi sırasında kullanılıyorsa, bu kural **yararlı** bir **kuraldır (usefull rule)**. Hiçbir tümcenin türetilmesinde kullanılmayan kurallara ise **yararsız kurallar (useless rules)** adı verilir.
- Eğer bağlamdan-bağımsız bir dilbilgisi yararsız değişken, uç simge ya da kural içeriyorsa, bu dilbilgisine eşdeğer olan ve yararsız uç simge, değişken ve kural içermeyen bir dilbilgisi bulunabilir. Bunun için de öncelikle değişkenlerden hangilerinin yararlı olduğunun bulunması gerekir. Bir değişkenin (**A**) yararlı bir değişken olabilmesi için:
 1. Bu değişkenden başlanarak, en az bir uç simge dizisinin türetililebilmesi:
 $A \Rightarrow u \quad u \in V_T^*$ (u : dilin bir tümcesi olması şart değil)
 2. Başlangıç değişkeninden bu değişkene ulaşmanın mümkün olması:
 $S \Rightarrow \alpha_1 A \alpha_2$

Bağlamdan-bağımsız bir dilbilgisi (**G**) verildiğinde, yararsız değişken ve kurallar atılarak eşdeğer bir dilbilgisinin bulunması iki adımda gerçekleştirilir.

4.3.4.1. Yararsız Değişken ve Kuralların Atılması

- Bağlamdan-bağımsız bir dilbilgisi (**G**) verildiğinde, yararsız değişken ve kurallar atılarak eşdeğer bir dilbilgisinin bulunması iki adımda gerçekleştirilir.
 - Birinci adımda, algoritma 4.5 kullanılarak uç simge dizgisi türeten değişkenler kümesi bulunur. Bu kümede yer almayan değişkenler yararsız değişkenlerdir. Dilbilgisinden bu değişkenler ile bu değişkenlerin yer aldığı kurallar çıkarılarak eşdeğer bir dilbilgisi (**G'**) bulunur.
 - **Algoritma 4.5.** Uç simge dizgisi türeten değişkenler kümesinin bulunması:
 1. $T_D = \{A \mid (A \Rightarrow w) \in P \text{ ve } w \in V_T^*\}$;
 2. ($T = T_{\text{eski}}$) oluncaya kadar aşağıdaki işlemleri tekrarla:
 - 2.1 $T_{\text{eski}} = T_D$;
 - 2.2 Dilbilgisindeki her değişken (A) için aşağıdaki işlemleri tekrarla:
 - 2.2.1 Her $(A \Rightarrow \beta)$ kuralı için aşağıdaki işlemleri tekrarla:

eğer $\beta \in (T_{\text{eski}} \cup V_T)^*$ ise A'yı T_D 'ye ekle
- Son 2.2.1:
- Son 2.2
- Son 2.

-4.3.4.1. Yararsız Değişken ve Kuralların Atılması

- İkinci adımda, algoritma 4.6 kullanılarak, birinci adımda bulunan dilbilgisindeki (G') değişkenlerden hangilerine başlangıç değişkeninden ulaşılabilirdi bulunur. Başlangıç değişkeninden ulaşılamayan değişkenler de yararsız değişkenlerdir. Dilbilgisinden (G') bu değişkenler ile bu değişkenlerin yer aldığı kurallar çıkarılarak eşdeğer dilbilgisi (G'') bulunur.
 - Algoritma 4.6.
 1. $T_U = \{S\};$
 2. $T_{\text{eski}} = \Phi;$
 3. $(T_U = T_{\text{Eski}})$ oluncuya kadar aşağıdaki işlemleri tekrarla:
 - 3.1. $T_{\text{yeni}} = T_U - T_{\text{eski}};$
 - 3.2. $T_{\text{Eski}} = T_U;$
 - 3.3 T_{yeni} 'deki her değişken için aşağıdaki işlemleri tekrarla:
 - 3.3.1 Her $(A \Rightarrow \beta)$ yeniden yazma kuralı için aşağıdaki işlemleri tekrarla:
 β 'daki tüm değişkenleri T_U 'ya ekle;
- Son 3.3.1;
- Son 3.3.1;
- Son 3;

Örnek 4.7

$$G_{4.7} = \langle V_N, V_T, P, S \rangle$$

$$V_N = \{ S, A, B, C, D, E, F \}$$

$$V_T = \{ a, b \}$$

$$P: S \Rightarrow B \mid AC \\ \mid BS$$

$$A \Rightarrow aA \mid aF$$

$$B \Rightarrow CF \mid b$$

$$C \Rightarrow cC \mid D$$

$$D \Rightarrow C \mid aD \\ \mid BD$$

$$E \Rightarrow aA \mid BSA$$

$$F \Rightarrow bB \mid b$$

- $(G_{4.7})$ 'ye eşdeğer; yararsız değişken, simge ve kural içermeyen bir dilbilgisi bulmak için önce $(G_{4.7})$ 'deki değişkenlerden hangilerinin uç simge dizgisi türeten değişkenler olduğunu bulmak gerekir. Uç simge dizgisi türeten değişkenler kümesi algoritma 4.5 ile aşağıdaki gibi bulunur:
- $T_D = \{ S, A, B, E, F \}$
- Yukarıdaki kümede yer almayan **C** ve **D** değişkenleri yararsız değişkenlerdir.

Örnek 4.7

$$G'_{4.7} = \langle V_N, V_T, P, S \rangle$$

$$V_N = \{ S, A, B, C, D, E, F \}$$

$$V_T = \{ a, b \}$$

$$P: S \Rightarrow B \mid BS$$

$$A \Rightarrow aA \mid aF$$

$$B \Rightarrow b$$

$$E \Rightarrow aA \mid BSA$$

$$F \Rightarrow bB \mid b$$

$$G''_{4.7} = \langle V_N, V_T, P, S \rangle$$

$$V_N = \{ S, B \}$$

$$V_T = \{ b \}$$

$$P: S \Rightarrow B \mid BS$$

$$B \Rightarrow b$$

- Şimdi de $(G'_{4.7})$ 'deki değişkenlerden hangilerinin S 'den ulaşılabilen değişkenler olduğunu bulmak gerekir. S 'den ulaşılabilen değişkenler kümesi algoritma 4.6 kullanılarak aşağıdaki gibi bulunur:

- $T_U = \{ S, B \}$

- Yukarıdaki kümede yer almayan **A**, **E** ve **F** değişkenleri yararsız değişkenlerdir. Dilbilgisinden bu üç değişken ile bu değişkenlerin yer aldığı kurallar çıkarıldığında a simgesinin de yararsız bir simge olduğu görülür. Bu simge de çıkarıldığında $(G_{4.7})$ 'ye eşdeğer; yararsız simge, değişken ve kural içermeyen aşağıdaki dilbilgisi bulunur:

4.5.2. Örnek 2

Örnek 2

Bağlamdan-bağımsız $G_{S.4.2}$ dilbilgisi aşağıdaki gibi tanımlanıyor.

$$G_{S.4.2} = \langle V_N, V_T, P, S \rangle$$

$$V_N = \{ S, D, E \}$$

$$V_T = \{ a, b, c \}$$

$$P : S \Rightarrow aaS \mid aaD \mid$$

$$D \Rightarrow bEb \mid ccD$$

$$E \Rightarrow bEb \mid c$$

- $G_{S.4.2}$ tarafından tanınan bağlamdan-bağımsız dilin ($L_{S.4.2}$) küme tanımını oluşturunuz.
- $G_{S.4.2}$ 'yi Chomsky normal biçimine (CNF) dönüştürünüz. Dönüştürme sırasında kullanacağınız yeni değişkenleri A, B, C , ve X_1, X_2, X_3, \dots diye adlandırınız. Oluşturduğunuz CNF dilbilgisinin tanımını eksiksiz ve düzenli biçimde yazınız.

4.5.2. Cevap 2

$$\mathbf{a)} \ L_{S.4.2} = \{ (aa)^n (cc)^m b^k c b^k \mid n \geq 1, m \geq 0, k \geq 1 \}$$

4.5.5. Örnek 5

Örnek 5

Aşağıda tanımlanan bağlamdan-bağımsız dillerin her birini türeten bağlamdan bağımsız bir dilbilgisi tanımlayınız.

$$\text{a) } L_{4.5.1} = \{ a^{2k} b^n c^{2n} d^k \mid k \geq 1, n \geq 1 \}$$

$$\text{b) } L_{4.5.2} = \{ a^{n+m} b^k c^{m+k} d^{2n} \mid n, m, k \geq 0 \}$$

4.5.5. Cevap 5

a) $G_{4.5.1} = \langle V_N, V_T, P, S \rangle$

$V_N = \{ S, A \}$

$V_T = \{ a, b, c, d \}$

$P : S \Rightarrow aaSd \mid aaAd$

$A \Rightarrow bAcc \mid bcc$

b) $G_{4.5.2} = \langle V_N, V_T, P, S \rangle$

$V_N = \{ S, A, B \}$

$V_T = \{ a, b, c, d \}$

$P : S \Rightarrow aSdd \mid A$

$A \Rightarrow aAc \mid B$

$B \Rightarrow bBc \mid \lambda$

4.5.6. Örnek 6

Örnek 6

Aşağıdaki bağlamdan-bağımsız her dilbilgisinin türettiği dilin küme tanımını veriniz.

a) $G_{S.4.6.1} = \langle V_N, V_T, P, S \rangle$

$$V_N = \{ S, A, B, C \}$$

$$V_T = \{ a, b \}$$

$$P : S \Rightarrow aS \mid aA$$

$$A \Rightarrow aA \mid bB$$

$$B \Rightarrow aB \mid bC$$

$$C \Rightarrow aC \mid a$$

b) $G_{S.4.6.2} = \langle V_N, V_T, P, S \rangle$

$$V_N = \{ S, X, Y \}$$

$$V_T = \{ a, b, c, d, e \}$$

$$P : S \Rightarrow aSe \mid XY$$

$$X \Rightarrow aXc \mid b$$

$$Y \Rightarrow cYe \mid d$$

4.5.6. Cevap 6

$$a) L_{S.4.6.1} = \{ a^n b a^m b a^k \mid n \geq 1, m \geq 0, k \geq 1 \}$$

$$b) L_{S.4.6.2} = \{ a^{n+m} b c^{m+k} d e^{k+n} \mid n, m, k \geq 0 \}$$

4.5.7. Örnek 7

Örnek 7

Aşağıda tanımlanan bağlamdan-bağımsız dillerin her birini türeten bağlamdan bağımsız bir dilbilgisi tanımlayınız.

$$\text{a) } L_{S.4.7.1} = \{ a^n b^m c^k \mid m > 0, k > 0, n > m + k \}$$

$$\text{b) } L_{S.4.7.2} = \{ a^n b^m c^k \mid n > 0, m > 0, k = 2n + m \}$$

4.5.7. Cevap 7

a) $G_{S.4.7.1} = \langle V_N, V_T, P, S \rangle$

$V_N = \{ S, A, B \}$

$V_T = \{ a, b, c \}$

$P : S \Rightarrow aaAc$

$A \Rightarrow aAc \mid aA \mid aBb$

$B \Rightarrow aBb \mid aB \mid \lambda$

b) $G_{S.4.7.2} = \langle V_N, V_T, P, S \rangle$

$V_N = \{ S, A, B \}$

$V_T = \{ a, b, c \}$

$P : S \Rightarrow aAcc$

$A \Rightarrow aAcc \mid bBc$

$B \Rightarrow bBc \mid \lambda$

4.5.8. Örnek 8

Örnek 8

Aşağıdaki bağlamdan-bağımsız her dilbilgisinin türettiği dilin küme tanımını veriniz.

$$\text{a) } G_{S.4.8.1} = \langle V_N, V_T, P, S \rangle$$

$$V_N = \{ S, B, C \}$$

$$V_T = \{ a, b, d \}$$

$$P : S \Rightarrow aSaa \mid B$$

$$B \Rightarrow bbBdd \mid C$$

$$C \Rightarrow bd$$

$$\text{b) } G_{S.4.6.2} = \langle V_N, V_T, P, S \rangle$$

$$V_N = \{ S, A \}$$

$$V_T = \{ a, b, c, d \}$$

$$P : S \Rightarrow abSdc \mid A$$

$$A \Rightarrow cdAba \mid \lambda$$

4.5.8. Cevap 8

$$a) L_{S.4.8.1} = \{ a^n b^{2k+1} d^{2k+1} a^{2n} \mid n \geq 0, k \geq 0 \}$$

$$b) L_{S.4.8.2} = \{ (ab)^n (cd)^k (ba)^k (dc)^n \mid n \geq 0, k \geq 0 \}$$