Biçimsel Diller ve Otomata Teorisi

Dr. Öğr. Üyesi Hayri Volkan Agun

Bilgisayar Mühendisliği Bölümü

Bursa Teknik Üniversitesi

Kaynaklar

Ders Kitabı

- An Introduction to Formal Languages and Autotamata, Peter Linz, 6th Edition, 2017.
- An Introduction to Computer Theory, Daniel Isaac Aryeh Cohen, 2nd Edition, 1996.

İçerik

- %100 Teorik
- Klasik sınav
- Vize %40, Final %60

Bağlam Bağımsız Diller ve Yığıt Otomatı

- Aşağıda kuralları verilen grammer için Yığıt Otomatı tasarlayınız?

$$S \rightarrow aSbb|a.$$

- Önce grameri anlaşılır hale getirmek için Greibach normal formuna dönüştürelim. Bu norm içinde kurallar sonlu sembol ve/veya onu takip eden birden fazla kural olabilir.
- Örneğin *S-> aAB, S->a* kuralları Greibach normal formundadır.

Yığıt Otomatı

$$S \rightarrow aSbb|a.$$

- Bu durumda aşağıdaki kurallar oluşturulabilir.
  - S -> aSB
  - B -> bA
  - A -> b
  - S -> a
- Bağımsız kural sayısı yığıt otomatında mevcut durum sayısına eşit olacağından S, B ve A için toplam 3 durum olacaktır.

Yığıt Otomatı

- Bu durumda aşağıdaki kurallar oluşturulabilir.
  - S -> aSB
  - B -> bA
  - A -> b
  - S -> a
- Bağımsız kural sayısı yığıt otomatında mevcut durum sayısına eşit olacağından S, B ve A için toplam 3 durum olacaktır.
- Burada yığıt otomatı kuralları ve geçişleri işletmek için kullanılacaktır.
- Başlangıç durumuna S kuralı ile başlamak için q0'da q1 durumuna yığına S ekleyerek geçeriz.

$$\Delta\ (q_0, \lambda, z) = \{(q_1, Sz)\}$$

- Bu otomata karşılık gelen dil nedir? Gösteriniz?

Yığıt Otomatı

- Bu durumda aşağıdaki kurallar oluşturulabilir.
  - S -> aSB
  - B -> bA
  - A -> b
  - S -> a
- Bağımsız kural sayısı yığıt otomatında mevcut durum sayısına eşit olacağından S, B ve A için toplam 3 durum olacaktır.

- İkinci ve tekrarlı durum q1 için yine q1 'e a ile geçeriz ve yığına SB atarız. Yada son kuralda S->a kuralı şeklinde yorumlarız ve yığını boşaltırız.

$$\Delta(q_1,a, S) \text{ -> } \{(q_1, SB), (q_1 , \lambda)\}$$

- Diğer geçişler için B ve A için yine q1 durumunu kullanabiliriz.

$$\Delta(q_1,b, B) \text{ -> } \{(q_1, A)\}$$

$$\Delta(q_1,b, A) \text{ -> } \{(q_1, \lambda)\}$$

- Son durum için q1 'de q2'ye boş geçiş ve yığına birşey ekelemden geçeriz. q2 son durumdur.

$$\Delta(q_1,\lambda, z) \text{ -> } \{(q_2, \lambda)\}$$

Yığıt Otomatı

$$
\begin{aligned}
S &\rightarrow aA, \\
A &\rightarrow aABC \,|bB|\, a, \\
B &\rightarrow b, \\
C &\rightarrow c.
\end{aligned}
$$

- Yandaki kurallar için yığıt otomatını tanımlayınız, kurallarını oluşturunuz?

- Bu yığıt otomatının ***aaabc*** karakter katarı için sonlanan adımlarını tablo üzerinde gösteriniz?

Kural

- Aşağıdaki tanıma uyan tüm yığıt otomatları bağlam bağımsız dile eştir.
  - Yığıt otomatı sadece tek bir sonlu durum barındırmalıdır ve sonlu duruma varırken yığın boş olmalıdır.
  - a herhangi bir sembol ve λ boş geçiş olmak üzere tüm geçişler aşağıdaki gibi olmalıdır.

$$\delta\,(qi, a, A) = \{c_1, c_2, \ldots, c_n\}$$

ve

$$c_i = (q_j, \lambda) \text{ yada } c_i = (q_j, BC)$$

  - Kısaca bir sembol tüketilir ya yığın eleman sayısı artar ya da azalır.

Örnek

- Aşağıda kuralları verilen yığıt otomatı bağlam bağımsız bir dile denk midir? Değilse gösteriniz?

$$
\begin{aligned}
\delta(q_0, a, z) &= \{(q_0, Az)\}, \\
\delta(q_0, a, A) &= \{(q_0, A)\}, \\
\delta(q_0, b, A) &= \{(q_1, \lambda)\}, \\
\delta(q_1, \lambda, z) &= \{(q_2, \lambda)\}.
\end{aligned}
$$

- İkinci geçiş içinde yığın elemanı A değişmemiştir. Bu durumda bu ikinci kurala aykırıdır.

Örnek

- İkinci geçiş içinde yığın elemanı A değişmemiştir. Bu durumda bu ikinci kurala aykırıdır.

$$\begin{aligned}
\delta(q_0, a, z) &= \{(q_0, Az)\}, \\
\delta(q_0, a, A) &= \{(q_0, A)\}, \\
\delta(q_0, b, A) &= \{(q_1, \lambda)\}, \\
\delta(q_1, \lambda, z) &= \{(q_2, \lambda)\}.
\end{aligned}$$

- Bu durumda aşağıdaki gibi q0 durumu üzerindeki döngüye q3 ara kuralı ekleyerek bu belirtilen kurala uydurulabilir. Burada q3 önce A çıkarır sonra A ekler.

$$\begin{aligned}
\delta(q_0, a, z) &= \{(q_0, Az)\}, \\
\delta(q_3, \lambda, z) &= \{(q_0, Az)\}, \\
\delta(q_0, a, A) &= \{(q_3, \lambda)\}, \\
\delta(q_0, b, A) &= \{(q_1, \lambda)\}, \\
\delta(q_1, \lambda, z) &= \{(q_2, \lambda)\}.
\end{aligned}$$

Soru

- Aşağıdaki gramerlerin her biri için karasız yığıt otomatı tasarlayınız?

1. $S \longrightarrow abSb|\lambda.$
2. $S \longrightarrow aSSSab|\lambda.$
3. $S \longrightarrow aSbb|abb.$

Soru

- Yanda kuralları verilen otomata eşlenik olan dilin gramer kurallarını yazınız?

$$M = (\{q_0, q_1\}, \{a, b\}, \{A, z\}, \delta, q_0, z, \{q_1\})$$

$$\begin{aligned}
\delta(q_0, a, z) &= \{(q_0, Az)\}, \\
\delta(q_0, b, A) &= \{(q_0, AA)\}, \\
\delta(q_0, a, A) &= \{(q_1, \lambda)\}.
\end{aligned}$$

Kararlılık

Kararlılık Durumu

- Yığıt otomatları ve bağlam bağımsız diller kararlı olabilirler. Yığıt otomatları için kararlı olma durumu aşağıdaki iki koşul varsa gerçekleşir.

❑ Herhangi bir durumdan sadece başka bir duruma sembol tüketerek geçilebilir.

$$|\delta(q, a, b)| = 1$$

❑ Bir durumdan başka hehangi bir duruma boş geçiş varsa o zaman bu durumdan başka duruma sadece bu geçiş olabilir.

Eğer $|\delta(q, \lambda, b)| = 1$ ise o zaman $|\delta(q, c, b)| = 0$

c ve b burada herhangi bir sembol olabilir.

Kararlı Diller

- Eğer bir bağlam bağımsız dil kararlı yığıt otomatı ile ifade edilebiliyorsa o zaman o dil kararlı bağlam bağımsız bir dildir.
- Örneğin L = {$a^n b^n$: n ≥ 0} için aşağıda ve yanda belirtilen yığıt otomatı kuralları kararlı yapıda mıdır?

M = ({q0, q1, q2} , {a, b} , {0, 1} , δ, q0, 0, {q0})

$$
\begin{aligned}
\delta(q_0, a, 0) &= \{(q_1, 10)\},\\
\delta(q_1, a, 1) &= \{(q_1, 11)\},\\
\delta(q_1, b, 1) &= \{(q_2, \lambda)\},\\
\delta(q_2, b, 1) &= \{(q_2, \lambda)\},\\
\delta(q_2, \lambda, 0) &= \{(q_0, \lambda)\}
\end{aligned}
$$