

Biçimsel Diller ve Otomata Teorisi

Sunu IV Sonlu Otomata

İZZET FATİH ŞENTÜRK



Dilleri Tanımlamak İçin Başka Bir Yöntem

- Aşağıdaki tanıma uyan birkaç oyun • Taşlar bir oyun tahtasına dizilir • Zarlar atılır ve rastgele bir sayı oluşturulur • Sayıya bağlı olarak taşlar, tamamen kurallar tarafından belirlenmiş bir şekilde yeniden düzenlenir
- Oyuncunun tahtayı değiştirme seçeneği yoktur
 - Her şeyi zar belirler • Zarı kim atarsa atсын, hiçbir beceri veya seçim söz konusu değildir • Kazanan tamamen zarı kimin attığına değil, hangi sayı dizisinin üretildiğine bağlıdır.

eyaletler

- Taşların tahtadaki olası tüm konumları • Durumlar diyelim
- Oyun, girdi ile bir durumdan diğerine değişir belirli sayıda
 - Her sayı için bir ve yalnızca bir sonuç durumu vardır • Oyun, bir sayı girildikten sonra aynı durumda olabilir • Zafer anlamına gelen bir durum vardır: son durum • Oyun, ilk durumla başlar (benzersiz)

Sonlu Otomat

- Sonlu: Alfabedeki olası durumların sayısı ve harflerin sayısı (olası zarlar) sınırlıdır • Otomat: Durumların değişimi tamamen girdi tarafından yönetilir
- Sıradaki durumun belirlenmesi otomatiktir
- Otomat kelimesinin çoğulu otomatadır

Tanım: Sonlu Otomat

- Sonlu bir otomat, üç şeyden oluşan bir koleksiyondur
 - Sonlu bir durumlar kümesi. Bunlardan biri başlangıç durumudur ve başlangıç olarak adlandırılır . durum. Bazıları (belki hiçbiri) nihai durumlardır
 - Olası giriş harflerinden oluşan bir Σ alfabesi
 - Her durumu ve her bir harf için ifade eden sonlu bir geçişler kümesi sonraki gitmek için hangi eyaletin alfabesi
- Tanım, bir FA'nın nasıl çalıştığını açıklamaz
 - Giriş dizesini en soldaki harften başlayarak harf harf okur • Başlangıç durumundan başlayarak, harfler bir durum dizisini belirler • Dizi son giriş harfi okunduğunda sona erer

Örnek

- Giriş alfabesinde a ve b olmak üzere iki harf vardır • Üç durum vardır, x, y ve z • Geçiş kuralları
 - Kural 1, x durumundan ve a girişinden y durumuna gidin •
 - Kural 2, x durumundan ve b girişinden z durumuna gidin •
 - Kural 3, y durumundan ve a girişinden x durumuna gidin •
 - Kural 4, Başlangıç y durumu ve b girişi, z durumuna gidin •
 - Kural 5, z durumundan ve herhangi bir girişten, z durumunda kalın
- Başlangıç durumu x'tir ve tek son durum z'dir • Bu, mükemmel şekilde tanımlanmış bir FA'dır çünkü üç şartı da karşılar: durumlar, alfabe, geçişler
- Giriş dizisi aaa veya abba olduğunda ne olur (kabul edildi mi yoksa reddedildi mi?)

Örnek

- Bir FA tarafından kabul edilen diziler , bu FA ile ilişkili dildir .
- Giriş dizisinde b ile karşılaşılır karşılaşmaz, FA z durumuna atlar ve z durumundan çıkmak imkansızdır • FA, içinde b harfi bulunan tüm dizileri kabul eder • $(a + b)^*b(b + b)^*$

Geçiş Tablosu

- Kuralları tablo biçiminde özetlemek çok daha basit
 - Her satır, FA'daki durumlardan biridir •

Her sütun, giriş alfabesinin bir harfidir • Girişler, FA'nın taşındığı yeni durumlardır

- FA için geçiş tablosu:

		b
x'i başlat	Y	
Y	X	
son z		

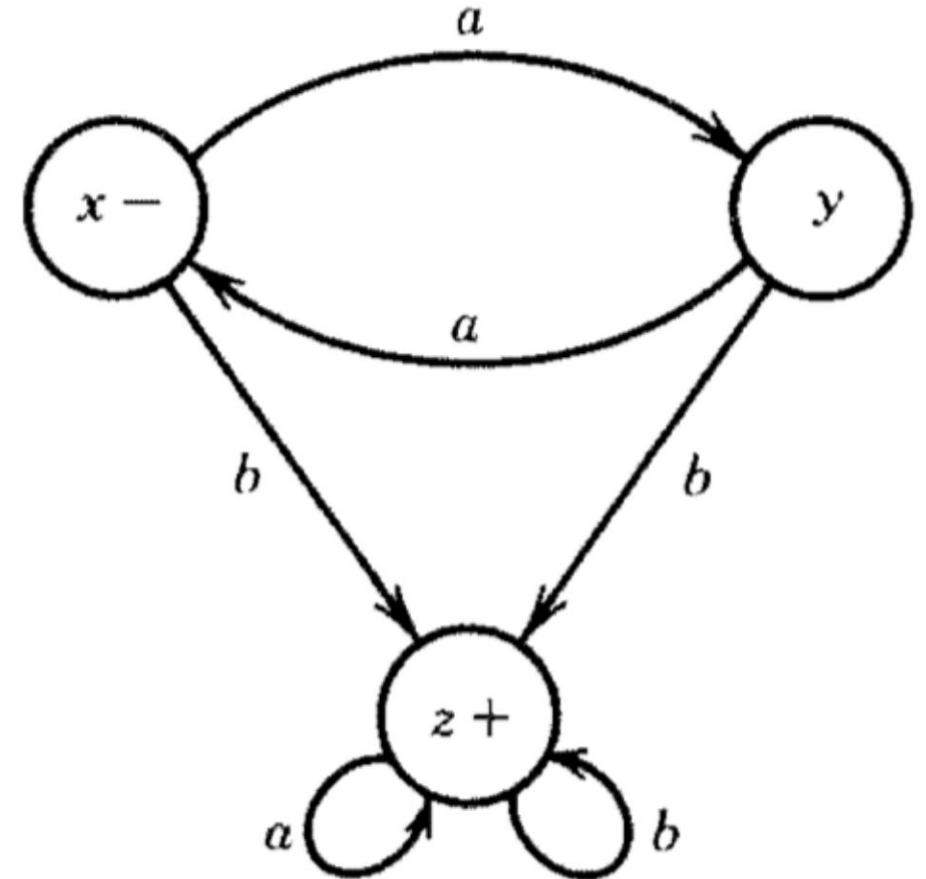
FA'nın Soyut Tanımı

1. Sonlu bir $Q = \{q_0, q_1, q_2, \dots\}$ durumları kümesi olup , bunların q_0 'i başlangıç durumu
2. Son durumlar olarak adlandırılan Q 'nun bir alt kümesi
3. Bir alfabe $\Sigma = \{x_1, x_2, x_3, \dots\}$
4. Her bir durum ve harf çiftini bir durumla ilişkilendiren bir geçiş fonksiyonu δ

$$\delta(q_i, x_j) = x_k$$

Geçiş Diyagramı

- Her durumu küçük bir daire ile temsil edin
- Her durumdan diğer durumlara oklar çizin
- Okları karşılık gelen alfabe harfleriyle etiketleyin
- Belirli bir harf durumu kendine geri döndürüyorsa: döngü
- Başlangıç durumu "başlangıç" kelimesiyle veya bir eksi işaretiyle gösterilir
- Nihai durumlar şu şekilde etiketlenir: kelime "son" veya artı işaretleri

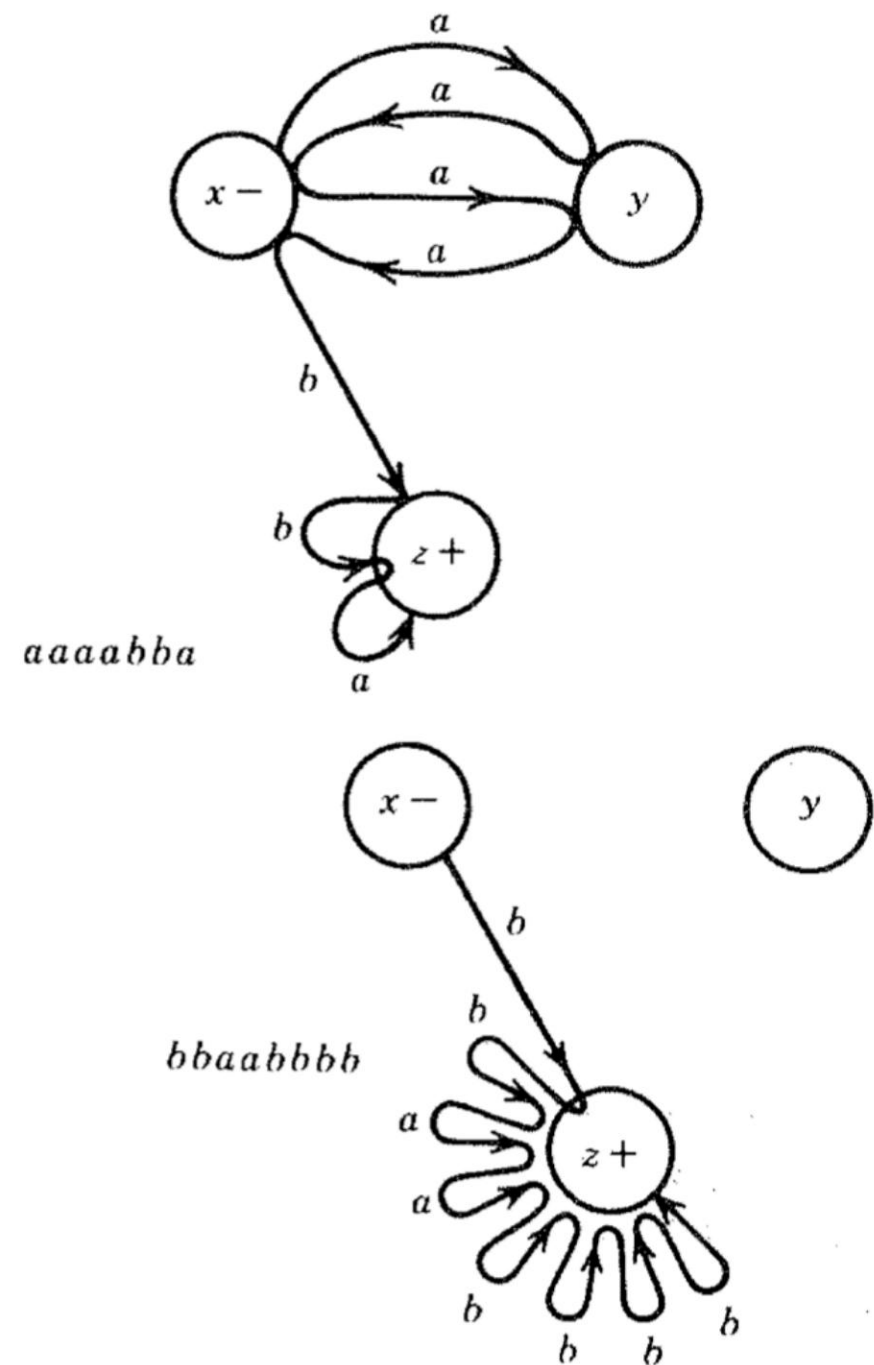
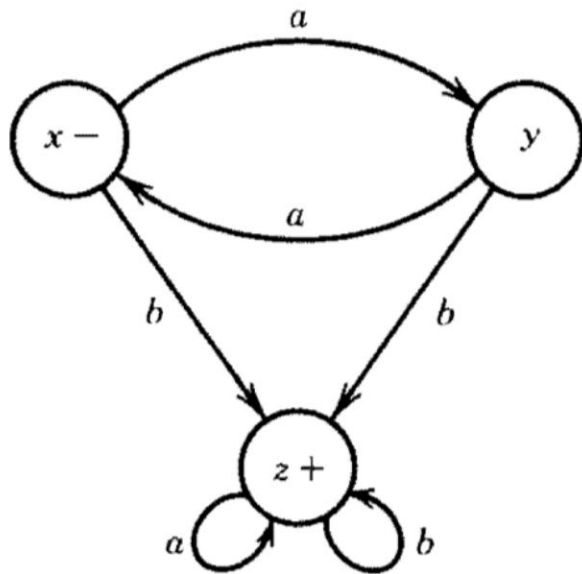


Harfler ve Geçiş Yolu

- Her giriş dizisi, bir yolu geçmek olarak yorumlanabilir
 - Başlangıç durumundan başlayın • Eyaletler arasında hareket edin (belki bazı eyaletleri birçok kez ziyaret edin) • Belirli bir dinlenme durumuna yerleşin • Bu bir son durumsa, yol başarıyla sona ermiştir
- Giriş dizisinin harfleri seyahat yönünü belirler • Harflerimiz bittiğinde durmalıyız

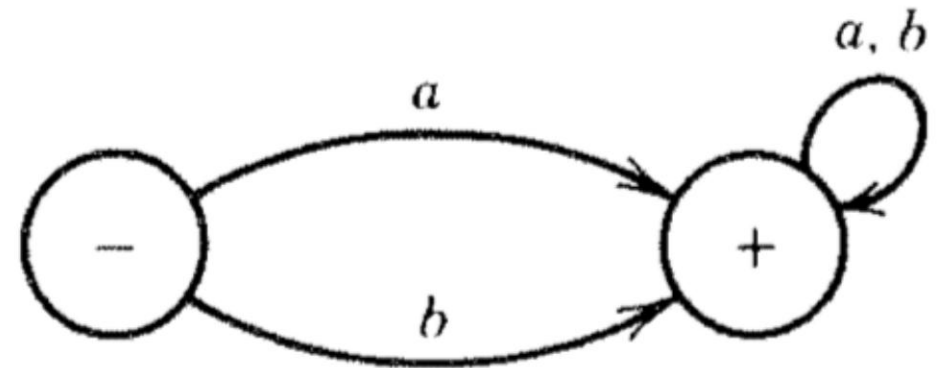
Örnek

- aaaabba ve bbaabbbb giriş dizeleri tarafından oluşturulan yollar



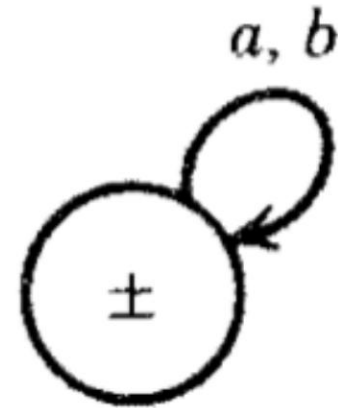
Örnek

- Bu makine tarafından kabul edilen dil, $\Lambda \cdot (a + b)(a + b)^* = (a + b)^+$ dışındaki tüm dizilerin kümesidir.



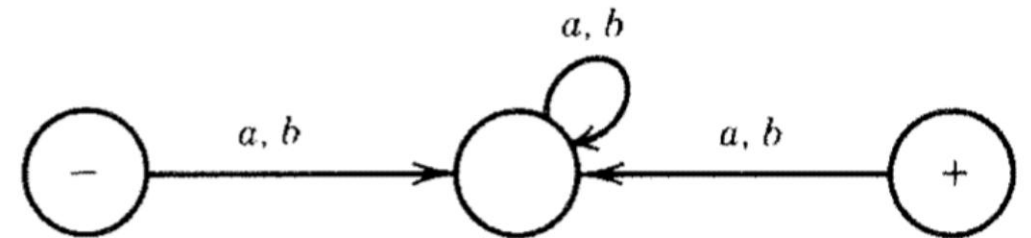
Örnek

- Tüm sözcükleri kabul eden birçok FA'dan biri •
- Aynı durum hem başlangıç durumu hem de son durumdur
- $(a + b)^*$



Örnek

- Aşağıdaki FA'lar vardır:
dil kabul etmeyin • İki tür vardır:
 - Nihai durumu olmayan FA'lar
 - Nihai durumu olan FA'lar
- başlangıç durumundan erişilemez



FA'lar ve Dilleri

- Bir dil bir RE tarafından tanımlandığında, dilde bulunan bazı rastgele sözcükleri üretmek kolaydır.
 - Ancak belirli bir harf dizisinin içinde olup olmadığını anlamak daha zordur. ifade tarafından tanımlanan dil
- FA ile durum tam tersidir!

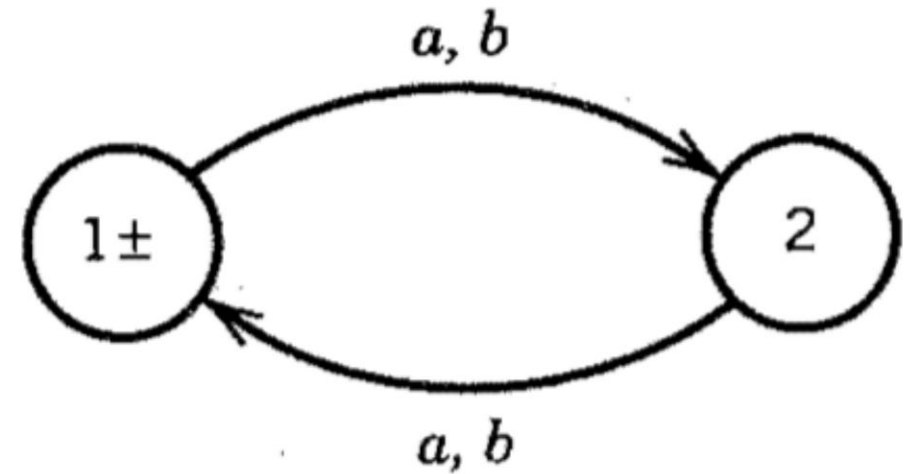
 - FA tarafından tanımlanan bir dil verildiğinde, makinenin kabul edeceğini önceden bildiğimiz bir grup kelimeyi yazmak kolay değildir.
- FA'yı iki farklı açıdan çalışma alıştırmaları yapmalıyız: • Bir dil verildiğinde, bunun için bir makine yapabilir miyiz? • Bir makine verildiğinde, onun dilini anlayabilir miyiz?

Örnek

- Çift sayılı harflerle $\{ab\}$ alfabesindeki tüm kelimelerin dilini kabul eden bir makine oluşturun • Matematikçi yaklaşım:
Soldan sağa doğru gidildikçe harflerin sayısını sayın
- Bilgisayar uzmanı: ?
 - Dizinin tam uzunluğuyla ilgilenmiyoruz • Bu sayı, gereksiz yere birçok hesaplama pahasına toplanan gereksiz bilgileri temsil ediyor
 - Bir Boole bayrağı kullanın, yalnızca bir depolama konumu kullanın. iki farklı değerden yalnızca birini içerebilir

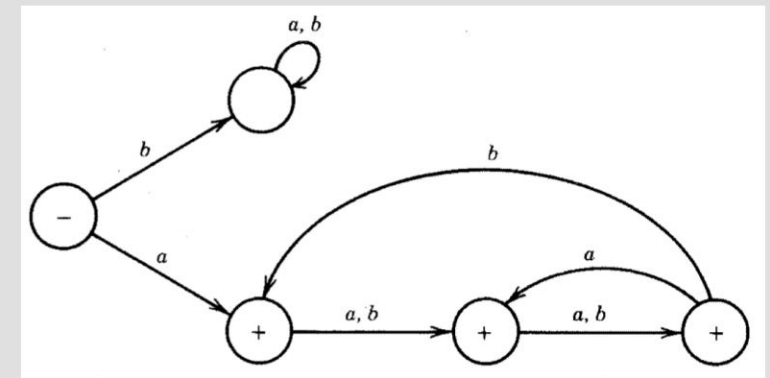
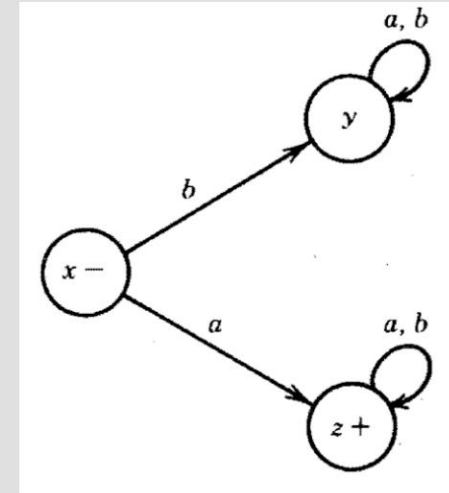
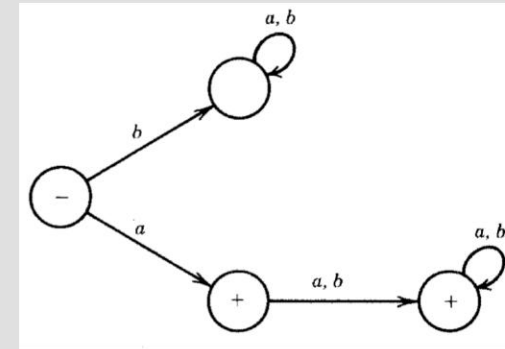
Örnek

- Harf sayısı çift olan $\{ab\}$ alfabesindeki tüm kelimelerin dilini kabul eden bir makine oluşturun



Örnek

- a harfi ile başlayan tüm dizeler olan dildeki tüm sözcükleri kabul eden bir FA oluşturun
- $a(a + b)^*$
- Belirli bir dil için benzersiz bir makine yoktur.
- Mümkün olan her dili kabul eden her zaman en az bir FA var mı?



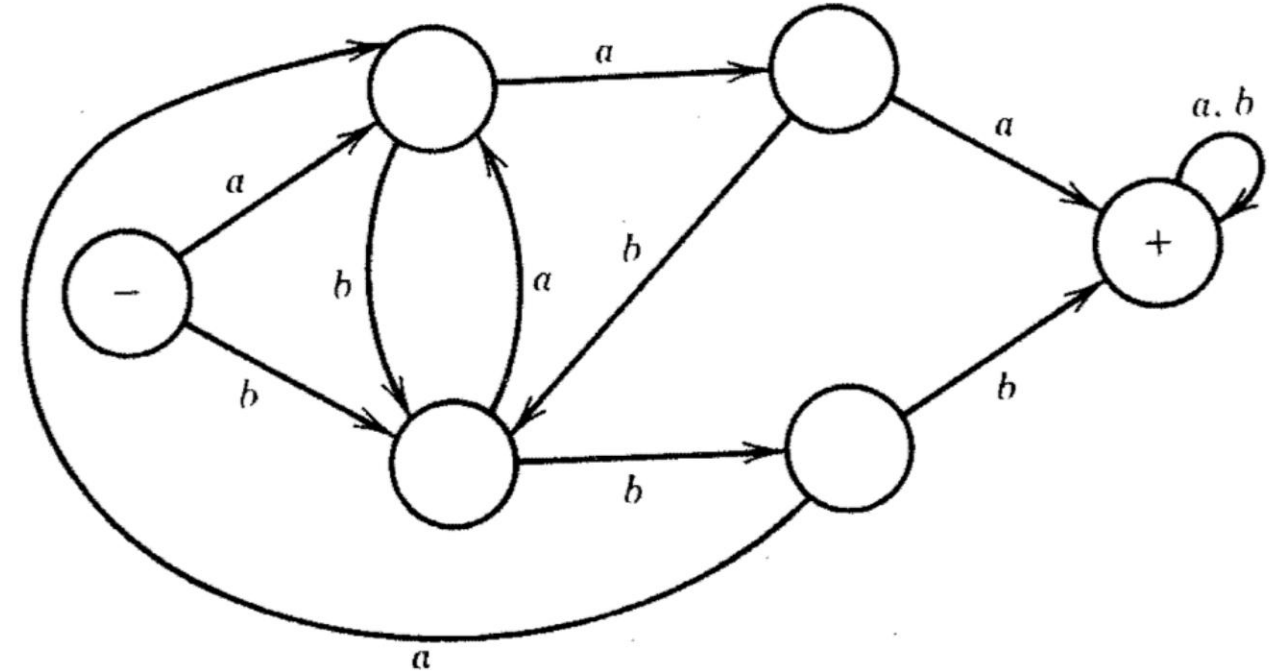
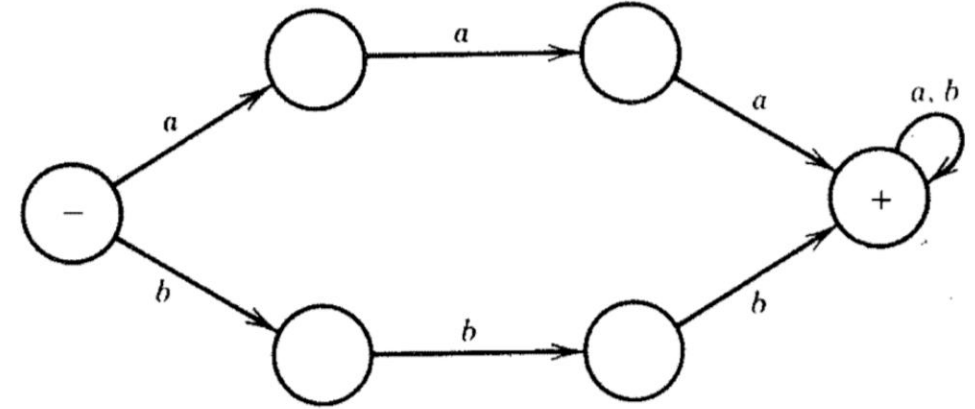
Örnek

- aaa veya bbb gibi üçlü harf içeren tüm sözcükleri ve yalnızca bu sözcükleri kabul eden bir FA oluşturun

- Anlayabiliriz

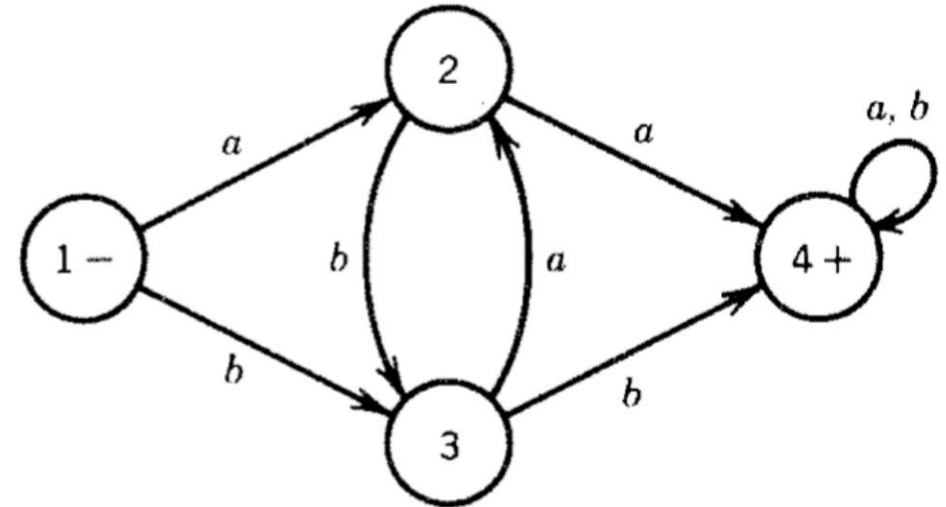
Bu FA'nın dili ve işleyişi çünkü nasıl inşa edildiğini gördük.

- Son resimden başlayıp anlamını yorumlamaya çalışsaydık..



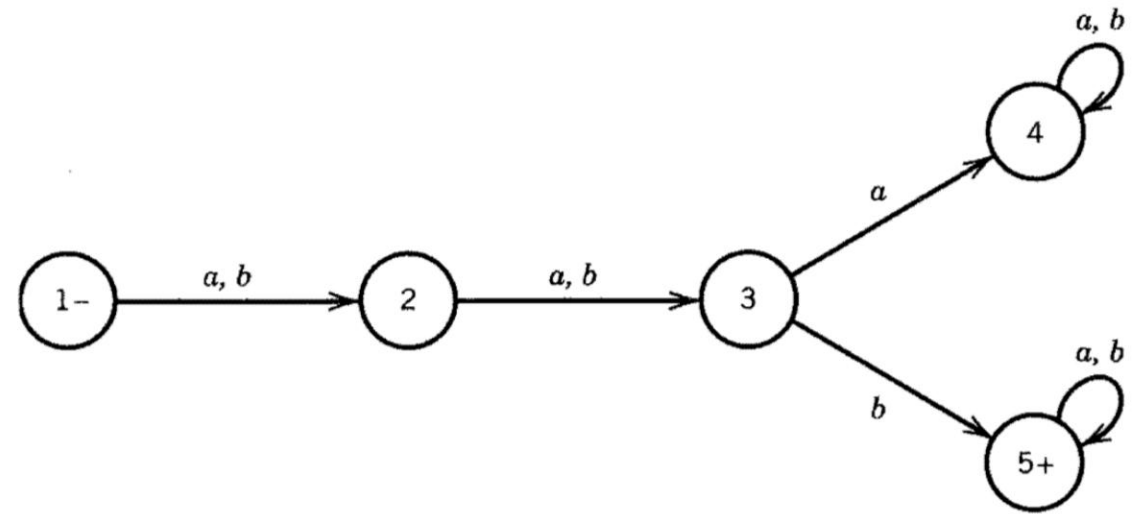
Örnek

- Bu makinenin hangi dili kabul ettiğini inceleyin •
ababa kabul edilmez • babbb kabul edilir
- Durum 4'e ulaşmanın iki yolu vardır • Durum 2'den (sadece a'yı okuyun) • Durum 3'ten (sadece b'yi okuyun)
- Çift harfli dizeler: • $(a + b)^*(aa + bb)$
 $(a + b)^*$



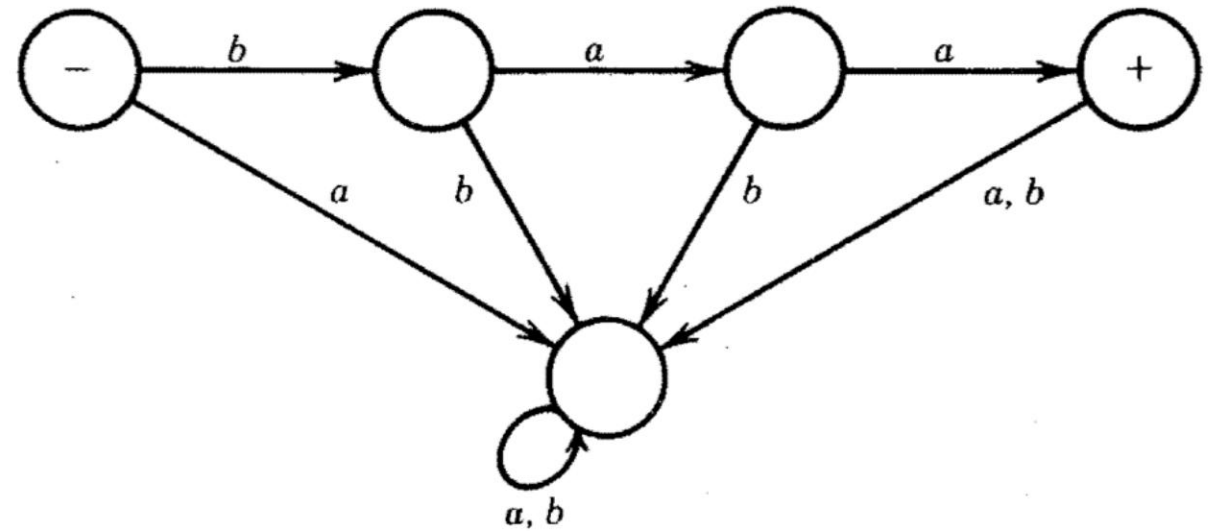
Örnek

- Bu makine tüm kelimeleri kabul edecek üçüncü harf olarak b ile ve diğer tüm kelimeleri reddet
- Bu dili tanımlayan bazı Res'ler: • $(aab + abb + bab + bbb)(a + b)^*$ • $(a + b)(a + b)(b)(a + b)^*$



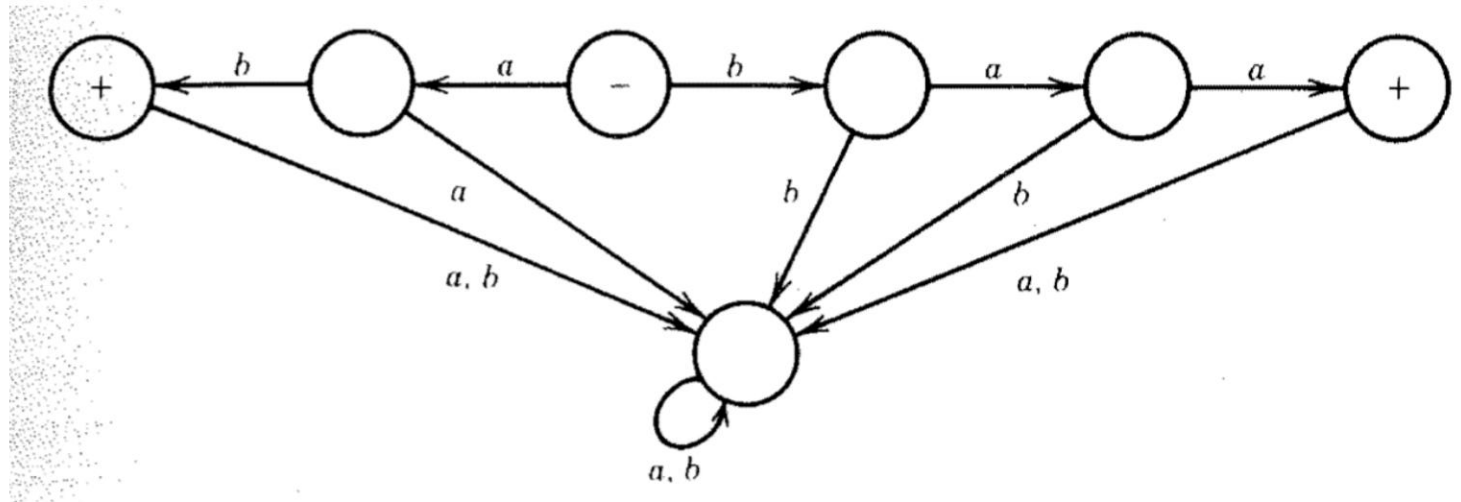
Örnek

- Yalnızca kelimeyi kabul eden bir FA evet
- $L = \{baa\}$



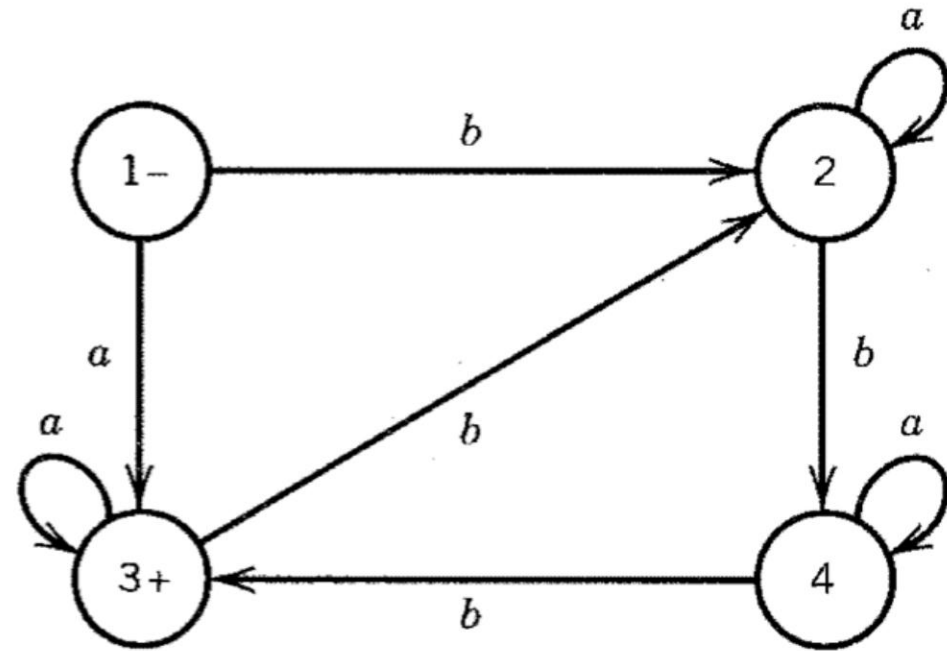
Örnek

- FA yalnızca iki diziye kabul eder: baa ve ab
- Büyük makine, küçük dil



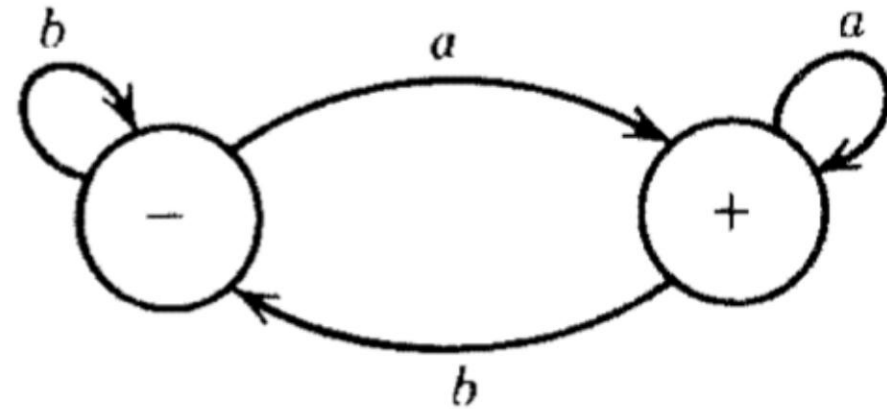
Örnek

- aa^* biçimindeki tüm sözcükler
- $a^*(a^*ba^*ba^*ba^*)^*(a + a^*ba^*ba^*ba^*)$
- Tek amacı
Son faktör, Λ 'nin bir olasılık olmadığını garanti etmektir.



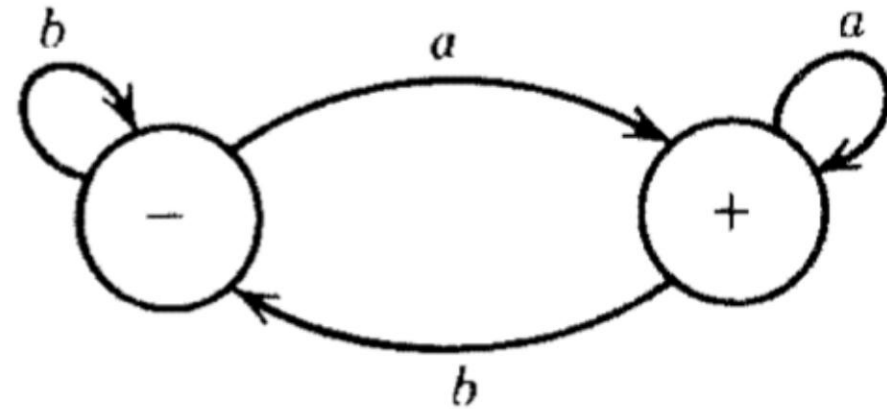
Örnek

- $(a + b)^*a$



Örnek

- $(a + b)^*a$



Örnek: Çift-Çift

- $[aa + bb + (ab + ba)(aa + bb)^*(ab + ba)]^*$

