Biçimsel Diller ve Otomata Teorisi

Sunu V

Geçiş Çizgeleri

İZZET FATİH ŞENTÜRK

Girdilerdeki Kısıtlamayı Gevşetmek

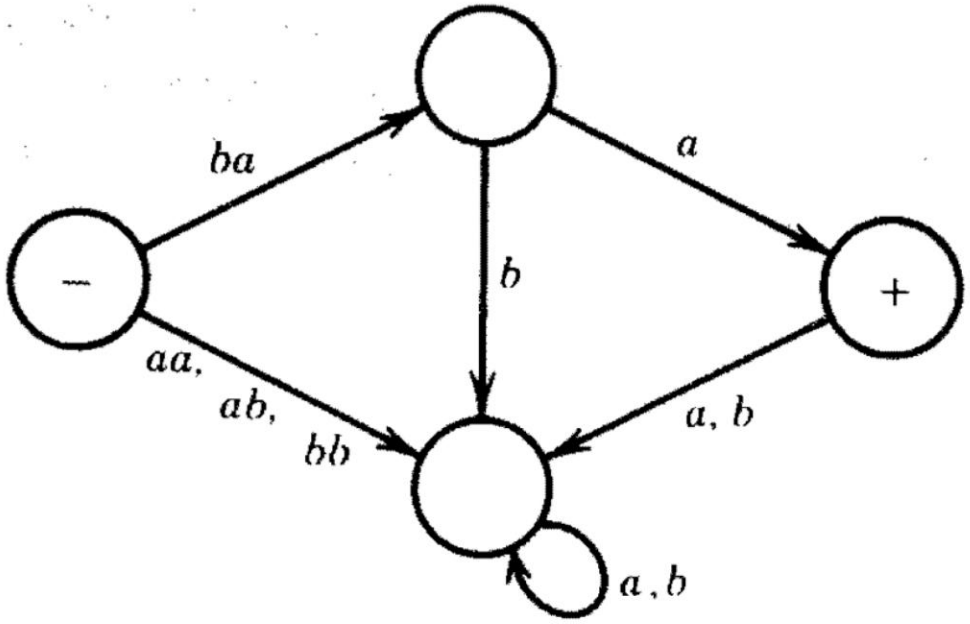
- Son bölümde, bir

  ANCAK

  - Yalnızca baa kelimesini kabul eder • Gerekli

  beş durum çünkü bir
  FA her seferinde bir harf okuyabilir

- Bir seferde bir veya iki harf okuyabilen daha güçlü bir makine tasarladığımızı varsayalım.



rahatlatıcı kısıtlama

girişler

- FA'ların bir varyasyonu
  - Biz terk ederiz

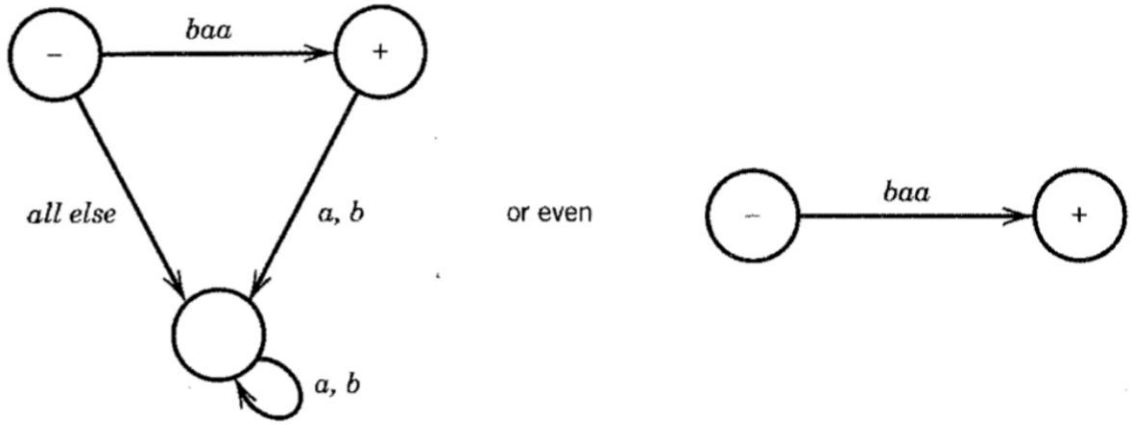
    kenarların bir seferde

    yalnızca bir harf yemesi gerekliliği

- Eğer bir şeyle ilgileniyorsanız

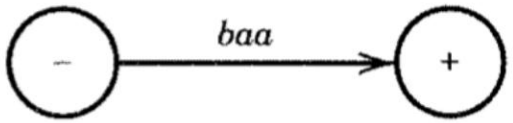
  sadece baa kelimesini kabul eden makine

  - Neden bir seferde sadece iki harf okuyasınız?



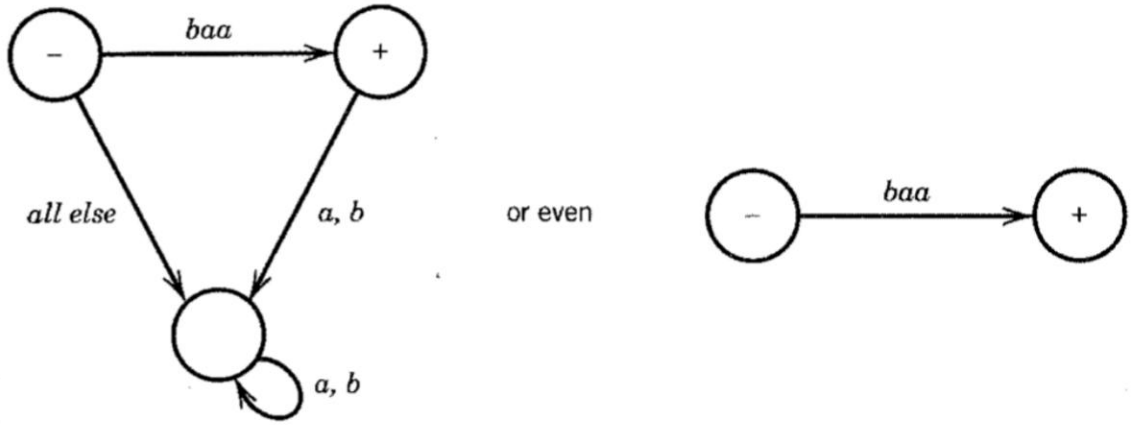
Girdilerdeki Kısıtlamayı Gevşetmek

- Resmi FA benzeri bir makine olarak yorumlayın
  - baa son duruma geç • Diğer tüm giriş dizileri hiçbir yerde bitmez
- Eksi durumundan başlarsak ve girişin ilk harfi a ise
  - Ne yapacağımız konusunda yönergemiz yok
- Giriş baabb ise
  - İlk üç harf bizi kabul durumuna götürür • Girilen harflerin devamını ne zaman okuruz? • FA kurallarına göre, giriş dizisi tamamen bitene kadar giriş harflerini okumayı durduramayız.



Çöp tenekesi durumu

- İzin verilen kenar geçişlerinden herhangi birini yapamadığımızda
  - Bir çöp kutusunun şunu belirttiğini varsayıyoruz: gitmeliyiz
- İki resmi eşdeğer kabul ediyoruz • Tamamen aynı dili kabul ediyorlar



Çöp tenekesi durumu

- Bir girdi bir makinede çalışırken ve henüz tam olarak okunmamış olsa bile kaçamayacağı bir duruma geldiğinde ne olduğunu açıklamak için yeni bir terim sunuyoruz.

- Tamamen okunmamış bir giriş dizisi, takip edebileceği bir çıkış kenarı olmadığı için ayrılamayacağı bir duruma (nihai veya başka türlü) ulaştığında • Girişin (veya makinenin) bu durumda kilitlendiğini söylüyoruz • Yürütme sonlandırılıyor • Giriş reddedildi

FA'lar ve Çökmeler

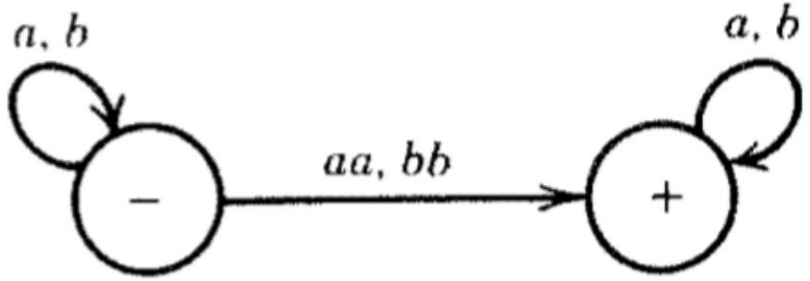
- Herhangi bir girişin çökmesi mümkün değildir.
  - Çünkü her durumda her zaman bir giden a-kenar ve bir giden b kenar vardır.
  - Okunmamış mektuplar kaldığı sürece ilerleme mümkündür

TG'ler ve Reddedilenler

- Bir girişi reddetmenin iki farklı yolu vardır • Son olmayan bir durumu sonlandıran bir yol izleyin • İşlenirken kilitlenme

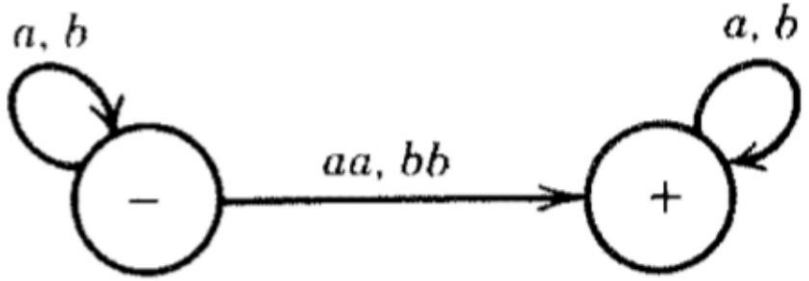
Alınacak Kararlar

- Aynı anda bir veya iki harf okuyabilen bir makine varsayalım .
  zaman
    - Çift harf içerebilen tüm kelimeleri tanır



- Bu makinede temel bir kuralı değiştirdik
    - Giriş dizisinden kaç harf okuyacağımıza karar vermeliyiz.
      her seferinde

Alınacak Kararlar



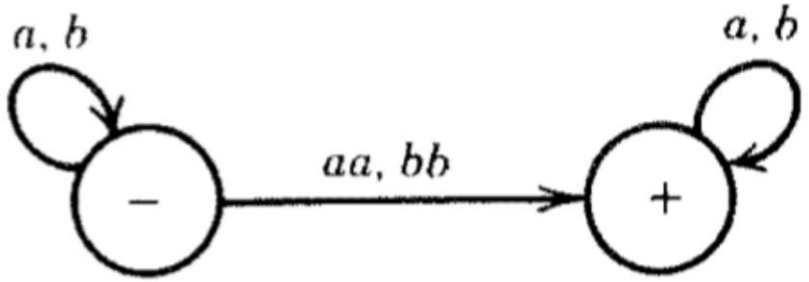
- Giriş dizisinin baa olduğunu varsayalım
- Bu dizinin nasıl kabul edildiğini görmek kolaydır: (b)(aa)
- Okunursa aynı diziyi reddederiz (b)(a)(a) • Ya (ba)

okursak)(a)? makine çöküyor

- Bu durum daha önce sahip olduğumuzdan tamamen farklı
  - Yollardan biri kabule , diğeri reddedilmeye götürür

giriş

Alınacak Kararlar



- Bu giriş dizisi, bu makinenin dilinin bir parçası mı yoksa olumsuzluk?

  - Makine operatörünün zekasına bağlı olamaz • Mutlak bir evet veya hayır olmalıdır • Aksi takdirde, dil iyi tanımlanmamıştır

Soyut Makinenin Tanımını Değiştirin

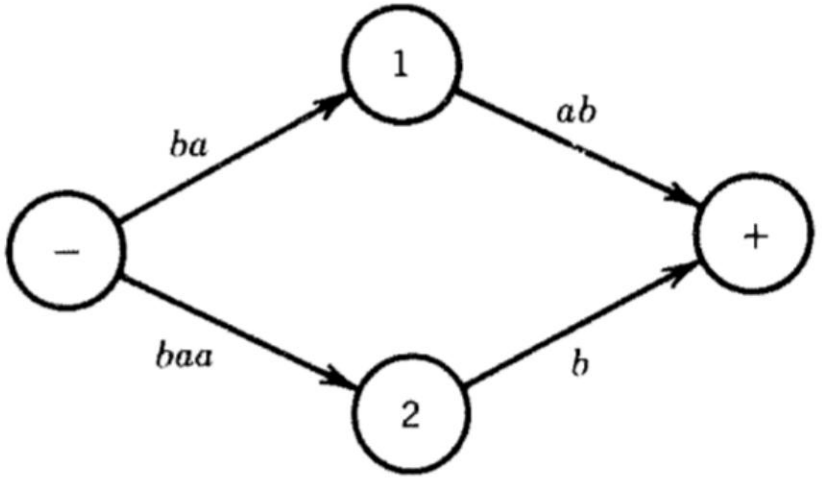
- Özetin tanımını değiştireceksek
  makine
  - Birden fazla harfin aynı anda okunmasını sağlamak için •

    Kabul tanımını da değiştirmeliyiz

- Bir dizgenin, son duruma gelmesi için işlenebilmesinin bir yolu varsa, makine tarafından kabul edildiğini söyleyeceğiz.

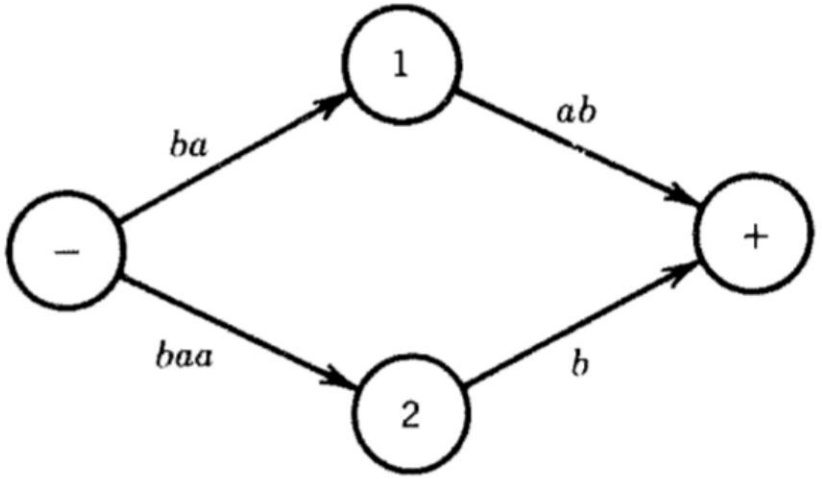
  - Bu dizgenin nihai duruma gelmediği durumlar da olabilir, ancak tüm hataları yok sayarız.

Soyut Makinenin Tanımını Değiştirin



- Aşağıdaki makineleri tasarlamak üzereyiz:
  - resimdeki herhangi bir kenar, herhangi bir alfabe harfi dizisiyle etiketlenebilir
- Bazı ek sonuçları dikkate almalıyız • Aşağıdaki sorunla karşılaşabiliriz..

Soyut Makinenin Tanımını Değiştirin



- Baab'ı iki farklı şekilde kabul edebiliriz • (baa)(b) • (ba)
  (ab)

- FA'larda, her giriş dizisi için benzersiz bir yolumuz vardı
  - Artık bazı dizgilerin hiç yolu yokken, bazılarının birden fazla yolu vardır.

Geçiş Grafikleri

- “Makine” tanımını aynı anda birden fazla girdi harfinin okunmasına izin verecek şekilde genişlettiğimizde pek çok zorluk gözlemledik • FA tanımını bırakıyoruz

- Bu yeni makinelere geçiş grafikleri diyoruz

Geçiş Grafiklerinin Tanımı

- TG, üç şeyin bir koleksiyonudur 1.
  En az biri en az biri olarak belirlenmiş sonlu bir durumlar kümesi
    başlangıç durumu (-) ve bazıları (belki hiçbiri) nihai durumlar (+) olarak belirlenir

  2. Giriş dizilerinin oluşturulduğu olası giriş harflerinden oluşan bir Σ alfabesi
     oluşur

  3. Giriş harflerinin belirtilen alt dizilerini okumaya dayalı olarak bazı durumlardan diğerlerine
     nasıl geçileceğini gösteren sonlu bir geçişler seti (kenar etiketleri) (boş dize Λ bile
     mümkündür)

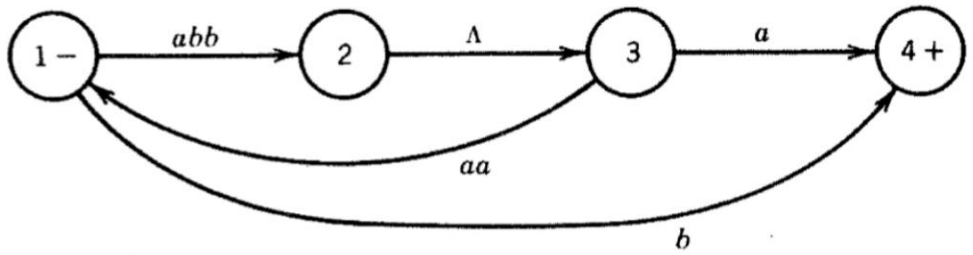
Geçiş Grafikleri

- Bazı eyaletlerin hiç avantajı olmayabilir ve bazılarının binlercesi olabilir (a, aa, aaa, aaaa, ...)
- Geçiş grafikleri 1957'de icat edildi • Geçiş

grafiğinde başarılı bir yol , bir başlangıç durumunda başlayan (birkaç tane olabilir) ve son durumda biten bir yol oluşturan bir dizi kenardır.

Geçiş Grafikleri

- Bu makine tarafından kabul
  edilen bir sözcük üretmek için yoldaki kenar
  etiketlerini birleştirin • (abb)(Λ)(aa)(b): abbaab



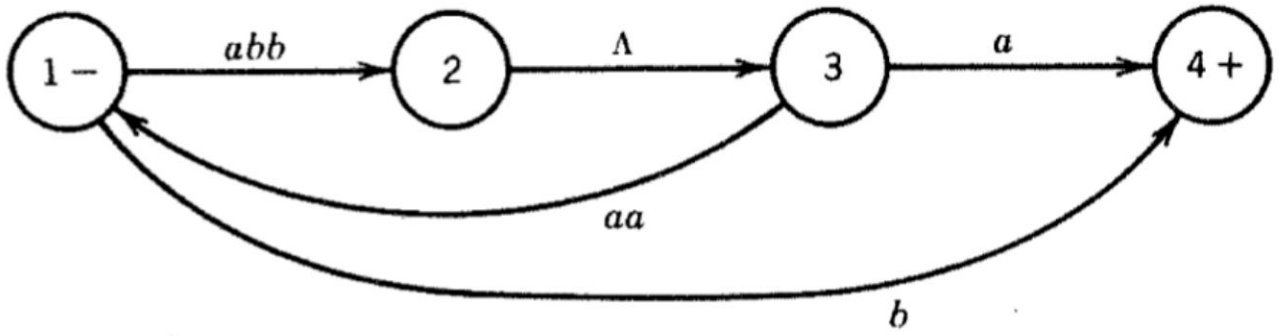
- Λ bedava yolculuk anlamına gelir
  - Harf tüketmeden • Takip etmemize gerek yok
    o kenar ama istersek yapabiliriz

Geçiş Grafikleri

• Belirli bir TG'de çalışacak belirli bir a'lar ve b'ler dizisi sunulursa • Sözcüğü alt dizilere nasıl ayıracağımıza karar vermeliyiz.

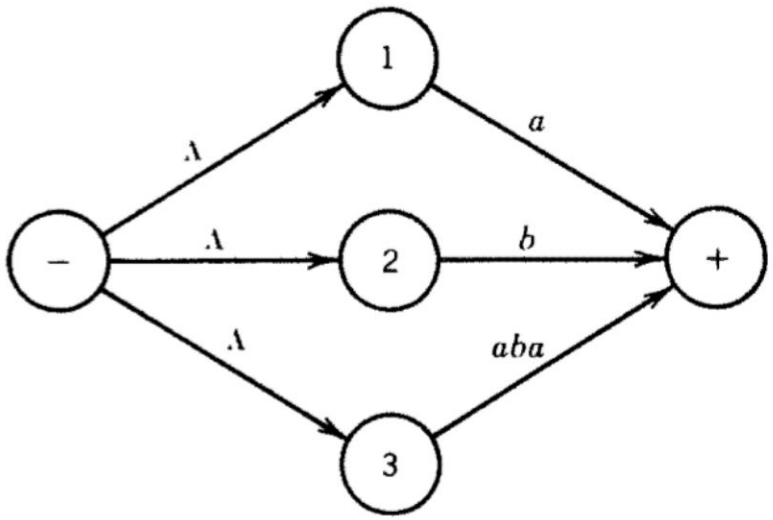
bir yoldaki kenarların etiketlerine karşılık

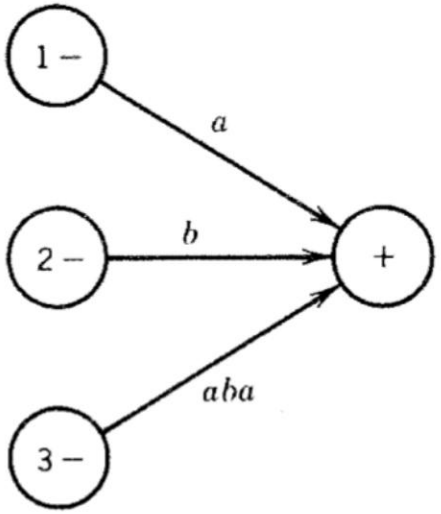
gelir • Aşağıdaki makinede abbab giriş dizesini göz önünde bulundurun



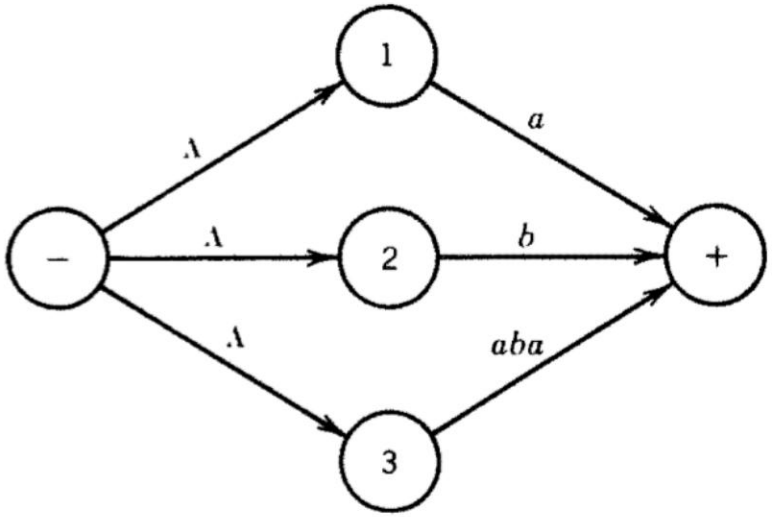
Geçiş Grafikleri

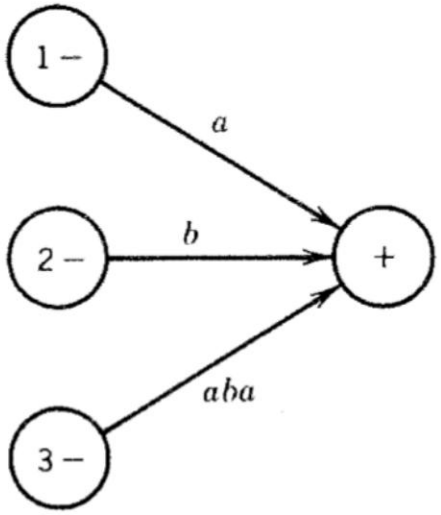
- Birden fazla başlangıç durumu olasılığı? • Bazı kenarların serbestçe geçilmesine izin verirsek (Λ)
  - Her zaman daha fazla başlangıç durumu sunabilir ve bunları Λ etiketli kenarlarla orijinal başlangıç durumuna bağlayabiliriz.





Geçiş Grafikleri





- İlk tarafından kabul edilen tüm diziler, birinci tarafından kabul edilir. ikinci ve tersi
- Sahip oldukları durum sayısı gibi farklılıklara rağmen eşdeğerdirler.

TG'ler ve FA'lar

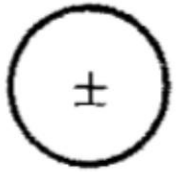
- Her FA aynı zamanda

bir TG'dir • Her TG, FA tanımını karşılamaz

Örnek

- Hiçbir şeyi kabul etmeyen bir TG, Λ boş dizisini bile

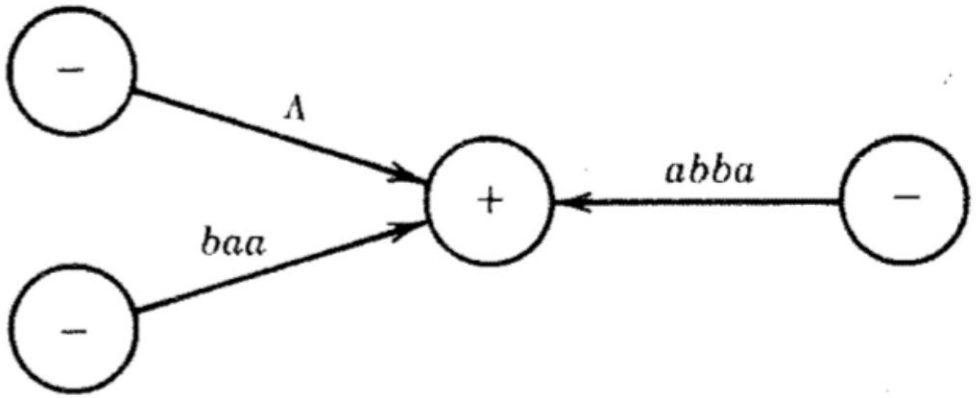
- Herhangi bir şeyi kabul edebilmek için nihai bir duruma sahip olması gerekir.



- Bu makine yalnızca Λ dizisini kabul eder.

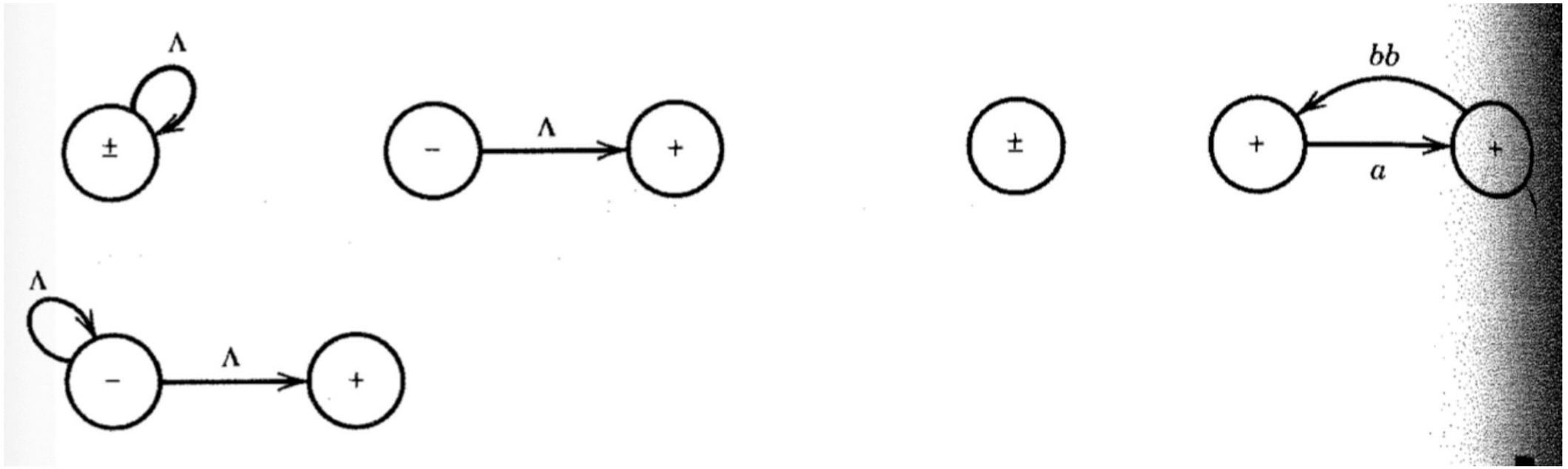
Örnek

- Bazı başlangıç durumlarının aynı zamanda son durum olduğu herhangi bir TG her zaman Λ dizesini kabul edecektir.
  - Bu aynı zamanda Fas için de geçerlidir.
- Λ kelimesini kabul eden başka TG'ler de vardır.
  - Anne ve babayı da kabul eder



Örnek

- Aşağıdaki TG'ler yalnızca Λ'yi kabul eder



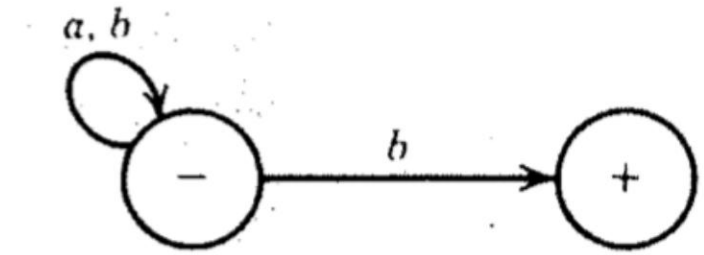
Örnek

- Girilen tüm harfleri birer birer okuyabilir ve sol tarafta kalabiliriz

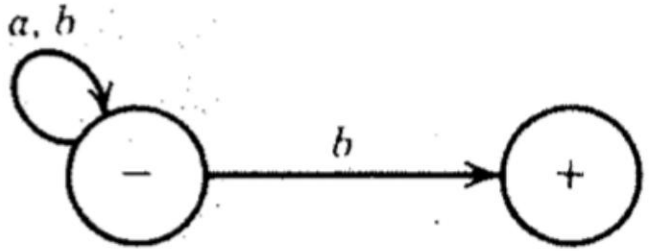
  - a b'yi – durumunda okuduğumuz zaman , iki kenar vardır.
    takip et

  - Son harf a b ise + durumuna geçmek için kullanabiliriz • Doğru durumda

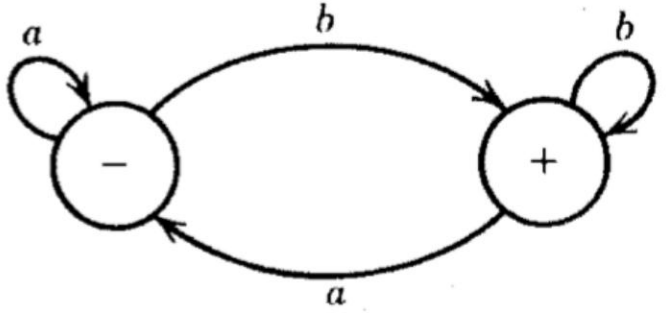
başka bir harf okumaya çalışırsak çarparız



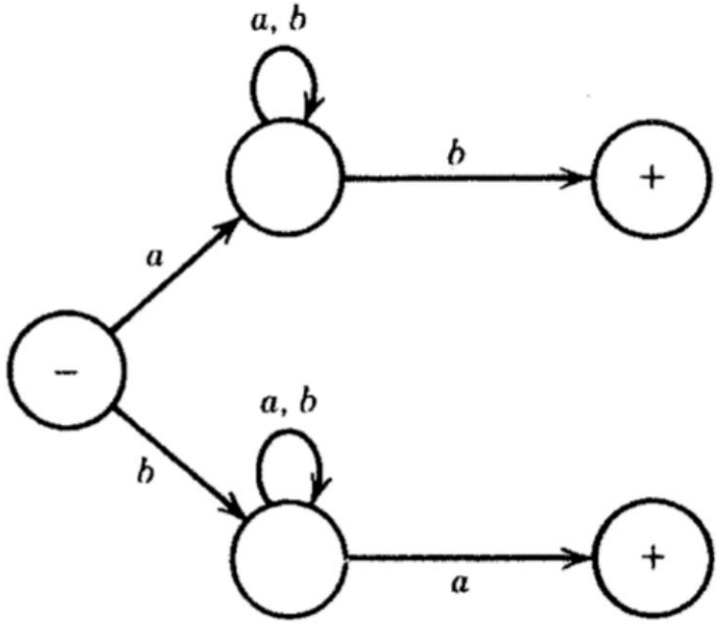
Örnek



- b ile biten tüm sözcükler bir şekilde kabul edilebilir • Bu TG tarafından kabul edilen dil, b ile biten tüm sözcüklerdir
- Bu dil için bir normal ifade: (a + b)*b
- Aynı dili kabul eden bir FA:

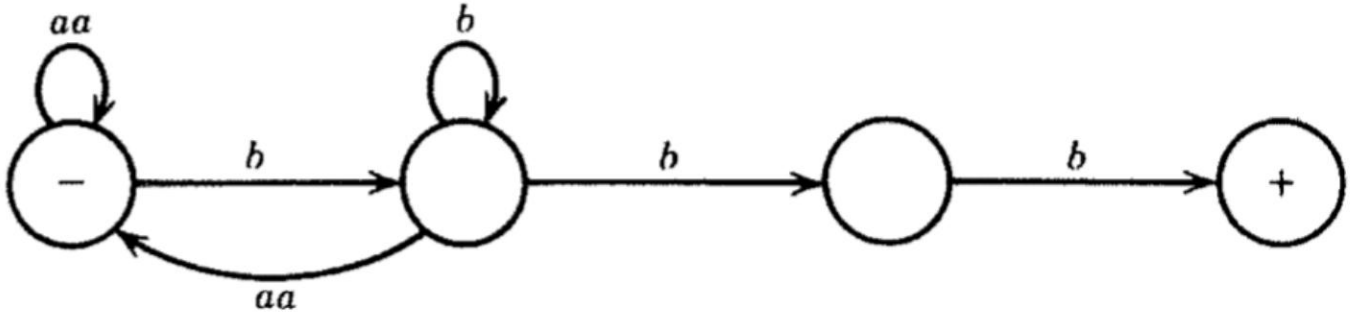


Örnek



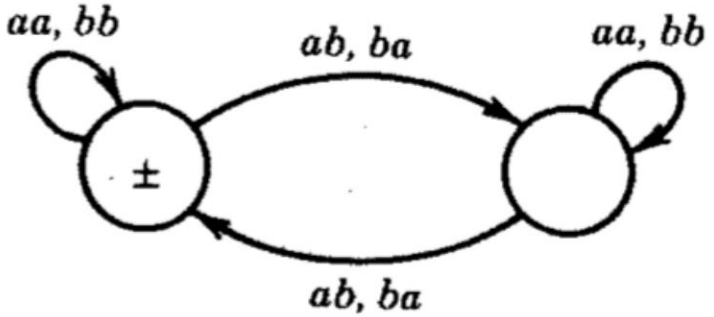
- Farklı harflerle başlayan ve biten tüm kelimelerin dili

Örnek



- a'ların yalnızca çift kümelerde geçtiği ve üç veya daha fazla b ile biten tüm kelimelerin dili

Örnek



- EVEN-EVEN: Çift olan tüm kelimelerin dili
  a'ların sayısı ve çift sayıda b'ler

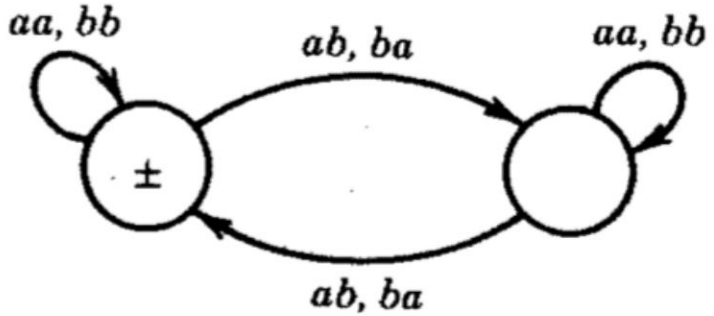
Çift-Çift: TG - FA - RE

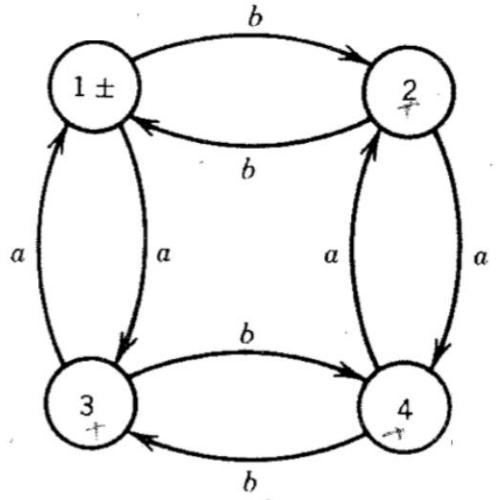
- Bu dilin tanımlarının üç örneğini inceledik

  - TG en anlaşılır olanıdır • TG'lerle

  ilgili pratik bir sorun mu var? Pek çok olası yol
  giriş dizesinin harflerini gruplandırma. Verilen dizgenin kabul edilip edilmeyeceğine karar vermek daha karmaşıktır.

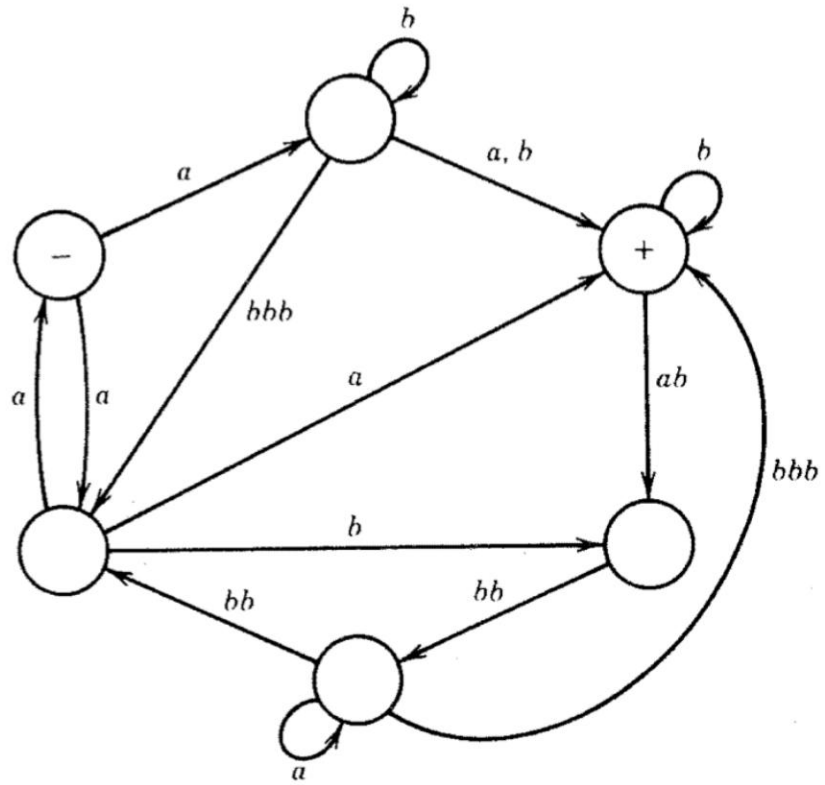
- [aa + bb + (ab + ba)(aa + bb)*(ab + ba)]*



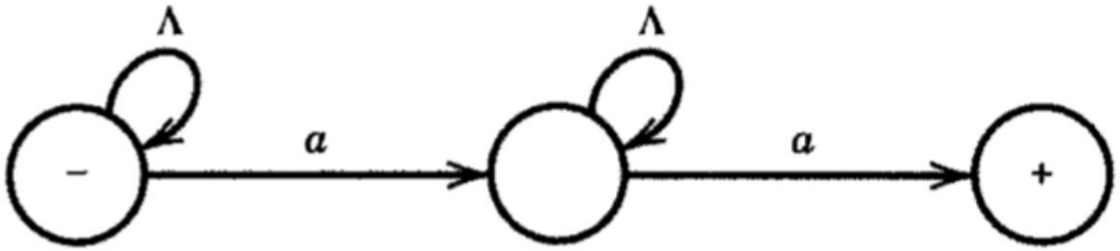


Örnek

- abbbabbabbabba kelimesi bu makine tarafından kabul ediliyor mu?



Örnek



- Bu makine tarafından kabul edilen tek kelime, tek kelimedir. aa
  - Ancak sonsuz sayıda farklı yolla kabul edilebilir • Λ-

ilmek kenarları hayatı zorlaştırabilir

- Λ-döngülerini kırptığımızda, grafik hala bir TG'dir ve aynı girdi dizisi setini kabul eder.
- Neden en başta Λ-döngülerine izin verdik?

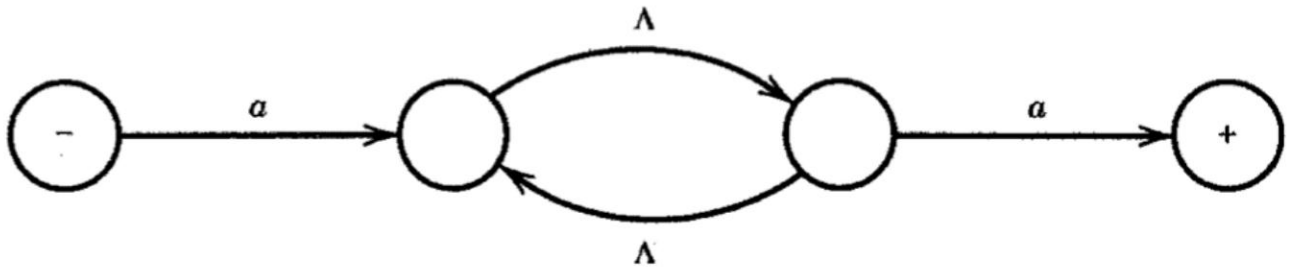
Λ Döngüler

- Tanım, Λ-döngüleri ile basit ve evrenseldir • Herhangi bir kenar, herhangi bir yerde, herhangi bir etiketle

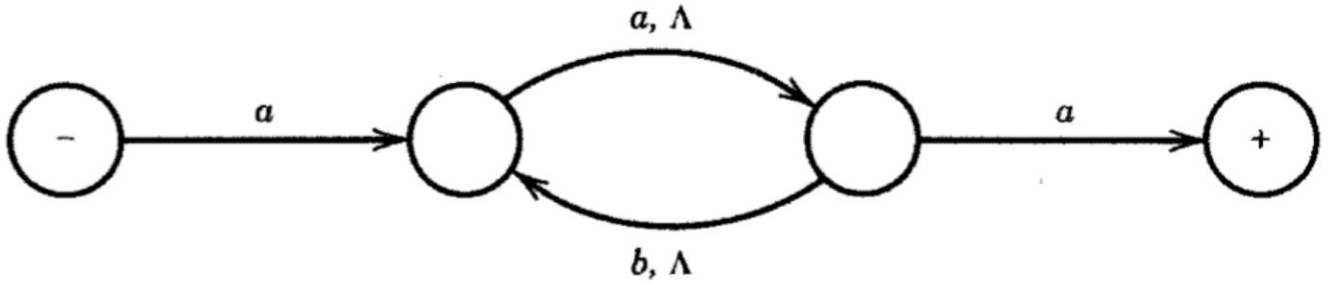
- Bölüm 7'de Λ-kenarların hiçbir zaman gerekli olmadığını göreceğiz. hiç
  - Λ-kenarlı bir TG tarafından kabul edilebilecek herhangi bir dil Λ-kenarları olmayan bazı farklı TG'ler tarafından kabul edilebilir

Λ Döngüler

- Λ-döngüsünün nasıl ortadan kaldırılacağı açıktır.



- Ancak bu makinede herhangi bir Λ seçeneği silinirse ortaya çıkan dil değişir



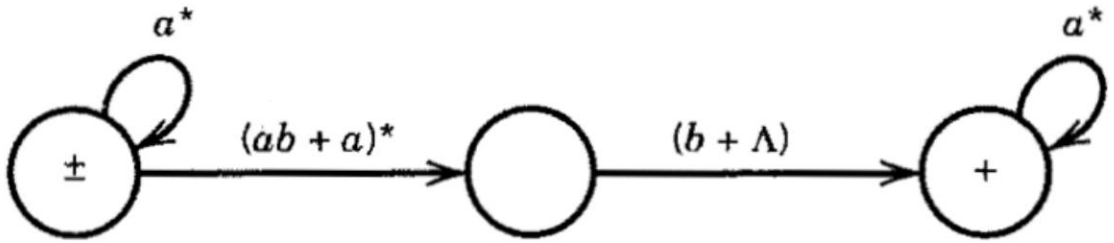
Genelleştirilmiş Geçiş Grafiği (GTG)

- GTG üç şeyden oluşan bir koleksiyondur
  1. En az birinin başlangıç durumu olduğu sonlu bir durumlar kümesi ve bazıları (belki hiçbiri) nihai durumlardır
  2. Giriş harflerinden oluşan bir Σ alfabesi 3.

  Her biri düzenli bir ifadeyle etiketlenmiş bazı durum çiftlerini birleştiren yönlendirilmiş kenarlar

Örnek

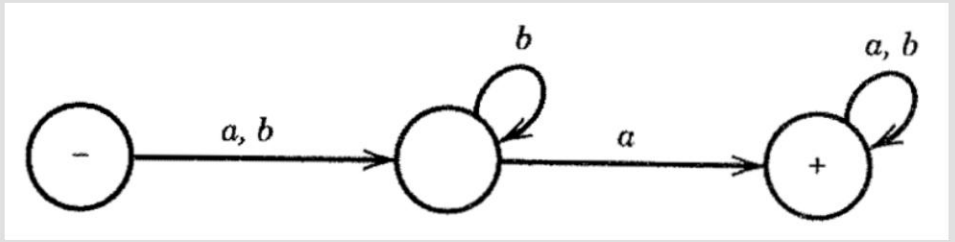
- Çift b olmadan tüm dizileri kabul eden makine

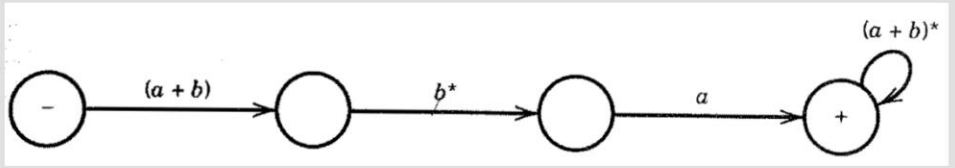


Kleene Star vs.

Döngü

Ortada döngü olmaması için b*
'den Λ'yi seçin





Belirsizlik

- Makinedeki nihai yol, yalnızca girdi tarafından belirlenmez.

- Yolun seçilmesinde insan tercihi bir faktör haline gelir •
Bu makine
kararsız
- Makine tüm belirlemeleri kendi yapmaz