Edit-Distance of Weighted Automata: General Definitions and Algorithms

Mehryar Mohri

International Journal of Foundations of Computer Science, 2003

指導教授:陳嘉平
學生:陳柏含

Introduction

- This paper proposes a general *synchronization* algorithm for weighted transducers.

- The synchronization algorithm combined with ε-removal algorithm can be used to *normalize* weighted transducers with bounded delays.

Overview of Transducer

- Weighted finite-state transducer $T=(A, B, Q, I, F, E, \lambda, \rho)$ over a semring $\mathbb{K}$
  - A : input alphabet
  - B : output alphabet
  - Q : a finite set of states
  - I : a set of initial state, $I \subseteq Q$
  - F : a set of final state, $F \subseteq Q$
  - E : a finite set of transitions, $E \subseteq Q \times (A \cup \{\varepsilon\}) \times (B \cup \{\varepsilon\}) \times \mathbb{K} \times Q$
  - λ : initial state weight assignment $\lambda : I \rightarrow \mathbb{K}$
  - ρ : final state weight assignment $\rho : F \rightarrow \mathrm{K}$

- Given a transition e belong to E
  - $n[e]$: e's destination state
  - $i[e]$: input label of e
  - $o[e]$: output label of e
- A path $\pi = e_1 \ldots e_k$ is a sequence of consecutive transitions

Synchronization Algorithm

- The objective of the algorithm is to synchronize the consumption of non-ε symbols by the input and output taps of a transducers as much as possible.

- Definition: The delay of a path π is defined as the difference of length between its output and input labels: $d[\pi] = |o[\pi]| - |i[\pi]|$

- The delay of a path is thus simply the sum of the delays of its constituent transitions.

- A weighted transducer T is said to be *synchronized* if along any successful path of T the delay is zero or varies strictly monotonically.

- A trim transducer T is said to have bounded delays if the delay along all paths of T is bounded.

Trim Transducer

- A state q of a transducer is *accessible* if q can be reached from I.

- A state q of a transducer is *coaccessible* if a final state can be reached from q.

- A transducer T is *trim* if all state of T are both accessible and coaccessible.

- $d[T] \geqq 0$ denoted as the maximum delay in absolute value of a path in T.

- A transducer T has bounded delays iff the delay of any cycle in T is zero.
  - If T admit a cycle π with non-zero delay, then $d[T] \leqq |d[\pi^n]| = n|d[\pi]|$ is not bounded.

String Delay

- Define the *string delay* of a path $\sigma[\pi]$

$$\sigma[\pi] = \begin{cases} \text{suffix of } o[\pi] \text{ of length } |d[\pi]| & \text{if } d[\pi] \geq 0 \\ \text{suffix of } i[\pi] \text{ of length } |d[\pi]| & \text{otherwise} \end{cases}$$

Preliminary

- *car(x):* first symbol of string x if x is not empty, ε otherwise.
- *cdr(x):* the suffix of string x such that

  $$x = car(x)cdr(x)$$

- $T = (A, B, Q, I, F, E, \lambda, \rho)$: input transducer
- $T' = (A, B, Q', I', F', E', \lambda', \rho')$: synchronized transducer

- We first augment Q and F with a new state f and set:

  $$\rho[f] = \overline{1} \text{ and } E[f] = \varphi$$

Synchronization(T)

```
F′ ← Q′ ← E′ ← ∅
S ← i′ ← {(i, ε, ε) : i ∈ I}
while  S ≠ ∅
        do  p′ = (q, x, y) ← head(S); DEQUEUE(S)
            if  (q ∈ F and |x| + |y| = 0)
                then  F′ ← F′ ∪ {p′}; ρ′(p′) ← ρ(q)
            else  if  (q ∈ F and |x| + |y| > 0)
                      then  q′ ← (f, cdr(x), cdr(y))
                            E′ ← E′ ∪ (p′, car(x), car(y), ρ[q], q′)
                            if  (q′ ∉ Q′)
                                then  Q′ ← Q′ ∪ {q′}; ENQUEUE(S, q′)
            for  each e ∈ E[q]
                 do  if  (|x i[e]| > 0  and |y o[e]| > 0)
                         then  q′ ← (n[e], cdr(x i[e]), cdr(y o[e]))
                               E′ ← E′ ∪ {(p′, car(x i[e]), car(y o[e]), w[e], q′)}
                         else  q′ ← (n[e], x i[e], y o[e])
                               E′ ← E′ ∪ {(p′, ε, ε, w[e], q′)}
                     if  (q′ ∉ Q′)
                         then  Q′ ← Q′ ∪ {q′}; ENQUEUE(S, q′)
return T′
```

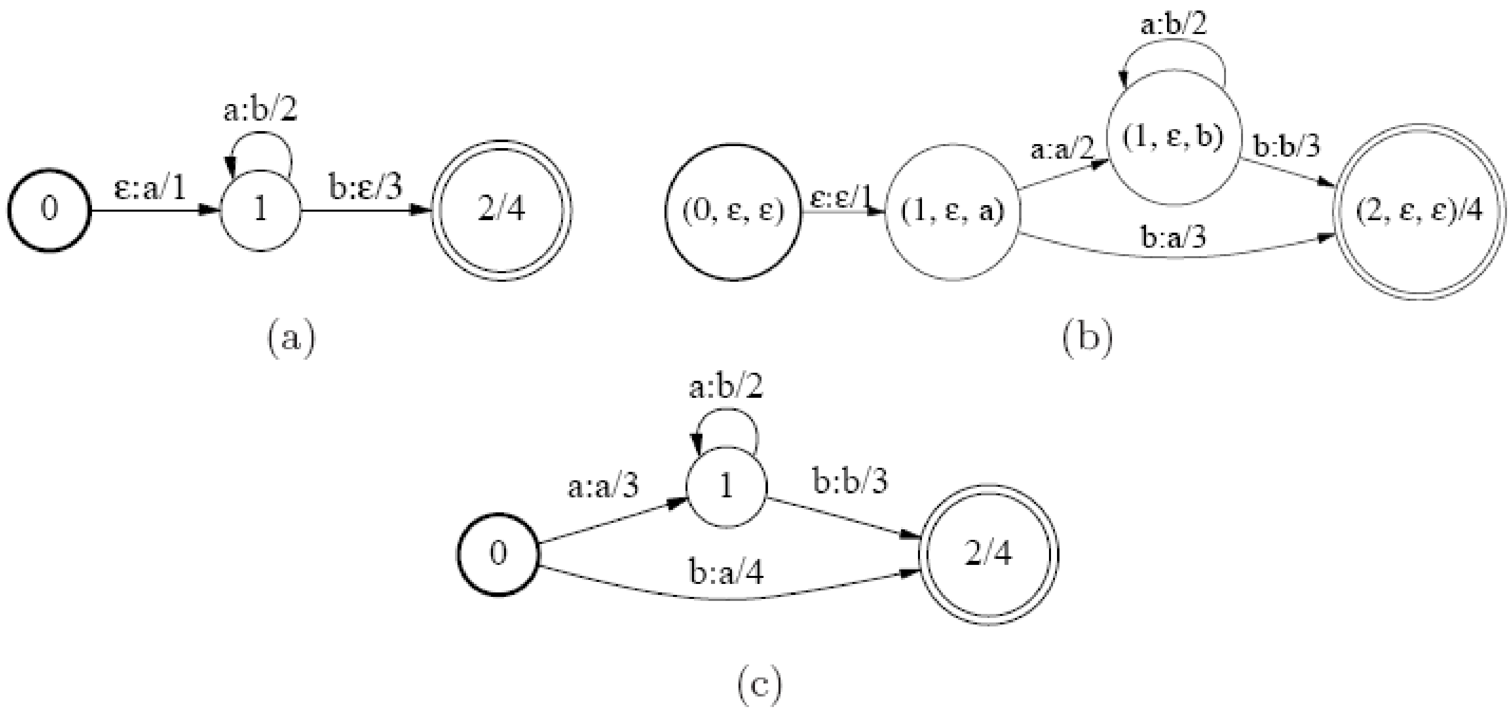

Figure 3: (a) Weighted transducer T_1 over the tropical semiring. (b) Equivalent synchronized transducer T_2. (c) Synchronized weighted transducer T_3 equivalent to T_1 and T_2 obtained by ϵ-removal from T_2.

- The synchronized algorithm presented terminates with any input weighted transducer T with bounded delays.

- Let T be a synchronized transducer and assume that T has no ε-transition. Then, T is *normalized*