# Disambiguation of Finite-State Transducers

**N. Smaili** and **P. Cardinal** and **G. Boulianne** and **P. Dumouchel**
Centre de Recherche Informatique de Montréal.
{nsmaili, pcardinal, gboulian, Pierre.Dumouchel}@crim.ca

## Abstract

The objective of this work is to disambiguate transducers which have the following form: $T = R \circ D$ and to be able to apply the determinization algorithm described in (Mohri, 1997). Our approach to disambiguating $T = R \circ D$ consists first of computing the composition $T$ and thereafter to disambiguate the transducer $T$. We will give an important consequence of this result that allows us to compose any number of transducers $R$ with the transducer $D$, in contrast to the previous approach which consisted in first disambiguating transducers $D$ and $R$ to produce respectively $D^{'}$ and $R^{'}$, then computing $T^{'} = R^{'} \circ D^{'}$ where $T^{'}$ is unambiguous. We will present results in the case of a transducer $D$ representing a dictionary and $R$ representing phonological rules.

**Keywords**: ambiguity, deterministic, dictionary, transducer.

## 1 Introduction

The task of speech recognition can be decomposed into several steps, where each step is represented by a finite-state transducer (Mohri et al., 1998). The search space of the recognizer is defined by the composition of transducers $T = A \circ C \circ R \circ D \circ M$. Transducer A converts a sequence of observations $O$ to a sequence of context-dependent phones.

Transducer $C$ converts a sequence of context-dependent phones to a sequence of context-independent phones. Transducer $R$ is a mapping from phones to phones which implements phonological rules. Transducer $D$ is the pronunciations dictionary. It converts a sequence of context-independent phones to a sequence of words. Transducer $M$ represents a language model: it converts sequences of words into sequences of words, while restricting the possible sequences or assigning a score to the sequences. The speech recognition problem consists of finding the path of least cost in transducer $O \circ T$, where $O$ is a sequence of acoustic observations.

The pronunciations dictionary representing the mapping from pronunciations to words can show an inherent ambiguity: a sequence of phones can correspond to more than one word, so we cannot apply the transducer determinization algorithm (an operation which reduces the redundancy, search time and possibly space). This problem is usually handled by adding special symbols to the dictionary to remove the ambiguity in order to be able to apply the determinization algorithm (Koskenniemi, 1990). Nevertheless, when we compose the dictionary with the phonological rules, we must take into account special symbols. This complicates the construction of transducers representing these rules and leads to size explosion. It would be simpler to compose the rules with the dictionary, then remove the ambiguity in the result and then apply the determinization algorithm.

## 2 Notations and definitions

Formally, a weighted transducer over a semiring $\mathbb{K} = (K, \oplus, \otimes, \bar{0}, \bar{1})$ is defined as a 6-tuple $T = (Q, I, \Sigma_1, \Sigma_2, E, F)$ where $Q$ is a finite set of states, $I \subseteq Q$ is a finite set of initial states, $\Sigma_1$ is the input alphabet, $\Sigma_2$ is the output alphabet, $E$ is a finite set of transitions and $F \subseteq Q$ is a finite set of final states. A transition is an element of $Q \times \Sigma_1 \times \Sigma_2 \times Q \times K$.
Transitions are of the form

$$t = (p(t), i(t), o(t), n(t), w(t)), t \in E$$

where $p(t)$ denotes the transition's origin state, $i(t)$ its input label, $o(t)$ its output label, $n(t)$ the transition's destination state and $w(t) \in K$ is the weight of $t$. The tropical semiring defined as $(R_+ \cup \infty, min, +, \infty, 0)$ is commonly used in speech recognition, but our results are applicable to the case of general semirings as well.

A path $\pi = t_1 \cdots t_n$ of $T$ is an element of $E^*$ verifying

$$n(t_{i-1}) = p(t_i) \text{ for } 2 \leq i \leq n.$$

We can easily extend the functions $p$ and $n$ to those paths:

$$p(\pi) = p(t_1), \qquad (1)$$
$$n(\pi) = n(t_n). \qquad (2)$$

We denote by $P(r, s)$ the set of paths whose origin is state $r$ and whose destination is state $s$. We can also extend the function $P$ to the sets $R \subset Q$ and $S \subset Q$:

$$P(R, S) = \bigcup_{r \in R,\ s \in S} P(r, s)$$

We can extend the functions $i$ and $o$ to the paths by taking the concatenations of the input and output symbols:

$$i(\pi) = i(t_1) \cdots i(t_n), \qquad (3)$$
$$o(\pi) = o(t_1) \cdots o(t_n). \qquad (4)$$

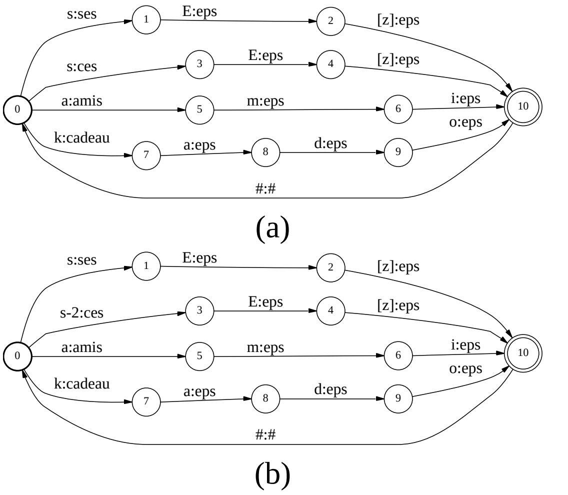
**Definition 1** *(unambiguous transducer, (Berstel, 1979))*
*A transducer $T$ is said to be unambiguous if for each $w \in \Sigma_1^*$, there exists at most one path $\pi$ in $T$ such that $i(\pi) = w$.*

**Definition 2** *(ambiguous paths)*
*Two paths $\pi$ and $\alpha$ are ambiguous if $\pi \neq \alpha$ and $i(\pi) = i(\alpha)$.*

**Remark 1** *: To remove the ambiguity between two paths $\pi$ and $\alpha$, it suffices to modify $i(\pi)$ by changing the first input label of the path $\pi$. This is done by introducing an auxiliary symbol such that: $i(\pi) \neq i(\alpha)$.*

Figure 1a shows an ambiguous transducer. It is ambiguous since for the input string "s e [z]", there are two paths representing the output strings $\{ces, ses\}$. In this figure, "eps" stands for epsilon or null symbol.

To disambiguate a transducer, we first group the ambiguous paths; we then remove the ambiguity in each group by adding auxiliary labels as shown in Figure 1b. Unfortunately, it is infeasible to enumerate all the paths in a cyclic transducer. However, in (Smaili, 2001) it is shown that cyclic transducers of the type studied in this work can be disambiguated by transforming to a corresponding acyclic subtransducer such that $T' \subset T$. This



(a)

(b)

Figure 1: *(a) Ambiguous transducer (b) Disambiguated transducer*

fundamental property is described in detail in section 2.1. Accordingly, we apply the appropriate transformation to the input transducer.

### 2.1 Fundamental Property

We are interested in the transducer $T = (Q,\ I,\ \Sigma,\ \Omega,\ E,\ F)$ with $\Sigma = \Sigma_0 \uplus \Sigma_1$ verifying the following property:

*Any cycle in $T$ contains at least a transition $t$ such that $i(t) \in \Sigma_1$.*

We denote by $E_0$ and $E_1$ the following sets: $E_0 = \{t \in E : i(t) \ \in \Sigma_0\}$ and $E_1 = \{t \in E : i(t) \ \in \Sigma_1\}$. Notice that $E = E_0 \uplus E_1$.

We can give a characterization of the ambiguous paths verifying the fundamental property. Before, let's make the following remark:

**Remark 2** *Any path $\pi$ in $T$ has the following form:*

$$\pi = f_0\ \pi_0\ f_1\ \pi_1 \cdots \pi_{n-1}\ f_n\ \pi_n$$

*with $\pi_i \in E_0^+$, $f_i \in E_1^+$ for $1 \leq i \leq n$, $f_0 \in E_1^*$ and $\pi_0 \in E_0^*$ if $n \geq 1$. If $n = 0$ then $\pi = f_0\ \pi_0$.*

**Proposition 1** *(characterization of ambiguous paths)*
*Let $\pi$ and $\alpha$ be two paths such that: $\pi = f_0\ \pi_0\ f_1\ \pi_1 \cdots \pi_{n-1}\ f_n\ \pi_n$ and $\alpha = g_0\ \alpha_0\ g_1\ \alpha_1 \cdots \alpha_{k-1}\ g_k\ \alpha_k$.*
*$\pi$ and $\alpha$ are ambiguous if and only if*

$$\begin{cases} k = n \\ \alpha_i \text{ and } \pi_i \text{ are ambiguous } (0 \leq i \leq n). \\ f_i \text{ and } g_i \text{ are ambiguous } (0 \leq i \leq n). \end{cases}$$

We will assume that the first transition's path belongs to $E_0$, *i.e.* $f_0 = \epsilon$. Recall that if we want to avoid cycles, we just have to remove from $T$ all transitions t $\in E_1$. According to Proposition 1, ambiguity needs to be removed only in paths that use transitions $t \in E_0$, namely the path $\pi_i$ that performs the decomposition given in Remark 2. Disambiguation consists only of introducing auxiliary labels in the ambiguous paths. We denote by $A_{src}$ the set of origin states of transitions belonging to $E_1$ and by $A_{dst}$ the set of destination states of transitions belonging to $E_2$.

$$A_{src} = \{p(t) : \ t \ \in \ E_1\}$$
$$A_{dst} = \{n(t) : \ t \ \in \ E_1\}$$

According to Proposition 1 and what precedes, it would be equivalent and simpler to disambiguate an acyclic transducer obtained from $T$ in which we have removed all $E_1$ transitions. Therefore, we introduce the operator $\Psi : \{T_{in}\} \longrightarrow \{T_{out}\}$ which accomplishes this construction.

Let $T = (Q, I, \Sigma_1, \Sigma_2, E, F)$. Then $\Psi(T) = (Q, I_1, \Sigma_1, \Sigma_2, E_T, F_1)$ where:

1. $I_1 = I \cup A_{dst} \cup \{i\}$, with $i \notin Q$.
2. $F_1 = F \cup A_{src} \cup \{f\}$, with $f \notin Q$.
3. $E_T = E \setminus E_1 \cup \{(i,\ q,\ \epsilon,\ \epsilon, 0),\ q\ \in I_1\} \cup \{(q,\ f,\ \epsilon,\ \epsilon, 0),\ q\ \in F_1\}$.

The third condition insures the connectivity of $\Psi(T)$ if $T$ is itself connected. It suffices to disambiguate the acyclic transducer $\Psi(T)$, then reinsert the transitions of $E_1$ in $\Psi(T)$. The set of paths in $\Psi(T)$ is then $\mathrm{P}(I_1,\ F_1)$.

### 2.2 Algorithm

Input:
$T = (Q,\ i,\ X,\ Y,\ E,\ F)$ is an ambiguous transducer verifying the fundamental property.

Output:
$T_1 = (Q,\ i,\ X \cup X_1,\ Y,\ E_T,\ F)$ is an unambiguous transducer, $X_1$ is the set of auxiliary symbols.

1. $T_{acyclic} \leftarrow \Psi(T)$.
2. Path $\leftarrow$ set of paths of $T_{acyclic}$.
3. Disambiguate the set Path (creating the set $X_1$).
4. $T_0$ $\leftarrow$ build the unambiguous transducer which has unambiguous paths.
5. $T_1$ $\leftarrow$ $\Psi^{-1}(T_0)$ (consists of reinserting in $T_0$ the transitions of $T$ which where removed).
6. return $T_1$

Now, we will study an important class of transducers verifying the fundamental property. This class is obtained by doing the composition of a transducer $D$ verifying the fundamental property with a transducer $R$. The composition of two transducers is an efficient algebraic operation for building more complex transducers. We give a brief definition of composition and the fundamental theorem that insures the invariance of the fundamental property by composition.

## 3 Composition

The transducer $T$ created by the composition of two transducers $R$ and $D$, denoted $T = R \circ D$, performs the mapping of word $x$ to word $z$ if and only if $R$ maps $x$ to $y$ and $D$ maps $y$ to $z$. The weight of the resulting word is the $\otimes$-product of the weights of $y$ and $z$ (Pereira and Riley, 1997).

**Definition 3 (Transitions)** *Let $t = (q,\ a,\ b,\ q_1,\ w_1)$ and $e = (r,\ b,\ c,\ r_1,\ w_2)$ be two transitions. We define the composition $t$ with $e$ by:*

$$t \circ e = ((q,\ r),\ a,\ c,\ (q_1,\ r_1),\ w_1 \otimes w_2).$$

Note that, in order to make the composition possible, we must have $o(t) = i(e)$.

**Definition 4 (Composition)** *Let $R = (Q_R, I_R, X, Y, E_R, F_R)$ and $S = (Q_S, I_S, Y, Z, E_S, F_S)$ be two transducers. The composition of $R$ with $S$ is a transducer $R \circ S = (Q, Q, X, Z, E, F)$ defined by:*

1. *$i = (i_R, i_S)$,*
2. *$Q = Q_R \times Q_S$,*
3. *$F = F_R \times F_S$,*
4. *$E = \{e_R \circ e_S : e_R \in E_R,\ e_S \in E_S\}$.*

Let $D = (Q_D, I_D, Y, Z, E_D, F_D)$ be a transducer verifying the fundamental property. We can write $Y = Y_0 \uplus Y_1$ where $Y_0 = \{i(t) \ :\ t \in E_0\}$ and $Y_1 = \{i(t) :\ t \in E_1\}$.

**Theorem 1 (Fundamental)** *Let $R = (Q_R, I_R, X, Y, E_R, F_R)$ verifying the following condition:*

*$(C)$ $\forall t \in E_R,\ o(t) \in Y_1 \Rightarrow i(t) \in Y_1$.*

*Then the transducer $T = R \circ D$ verifies the fundamental property.*

**Proof**:
Let $X_1 = \{i(t) : t \in E_R \text{ and } o(t) \in Y_1\} \subset Y_1$ and $X_0 = X \setminus X_1$. We will prove that any path in $T$ contains at least a transition $t$ such that $i(t) \in X_1$. Let $\pi$ be a cycle in $T$. Then, there exists two cycles $\pi_R$ and $\pi_D$ in $R$ and in $D$ respectively such that $\pi = \pi_R \circ \pi_D$. The paths $\pi_R$ and $\pi_D$ have the following form:

$$\begin{gathered}\pi_D = g_1 \cdots g_n,\\ \text{with } g_i \in E_D \text{ for } 1 \le i \le n;\\ \pi_R = f_1 \cdots f_n,\\ \text{with } f_i \in E_R \text{ for } 1 \le i \le n;\end{gathered}$$

$\pi = \pi_R \circ \pi_D = (f_1 \circ g_1) \cdots (f_n \circ g_n)$. There is an index $k$ such that $i(g_k) \in Y_1$ since $D$ verifies the fundamental property. We also necessarily have $i(g_k) = o(f_k)$ . According to condition $(C)$ of Theorem 1, we deduce that $i(f_k) \in Y_1$. Knowing that $f_k \in E_R$, we deduce that $i(f_k) \in X_1$, which implies $i(f_k \circ g_k) = i(f_k) \ \in X_1$.

### 3.1 Consequence

The restriction to the case $X = Y$ allows us to build a large class of transducers verifying the fundamental property. In fact, if two transducers $R = (Q_R, I_R, Y, Y, E_R, F_R)$ and $S = (Q_S, I_S, Y, Y, E_S, F_S)$ verify the condition $(C)$ of Theorem 1, then $S \circ R$ verifies the condition $(C)$, associativity of $\circ$ implies:

$$S \circ (R \circ D) = (S \circ R) \circ D.$$

Suppose that we have $m$ transducers $R_i$ ( $1 \le i \le m$ ) verifying the condition $(C)$ of Theorem 1 and that we want to reduce the size of the transducer:

$$T_m = R_m \circ R_{m-1} \cdots R_1 \circ D.$$

To this end, we proceed as follows: we add the auxiliary symbols to disambiguate the transducer; then we apply determinization and finally we remove the auxiliary labels. These three operations are denoted by $\psi$.

$$T_i = \begin{cases} \psi(D) \text{ if } i = 0.\\ \psi(R_i \circ \psi(T_{i-1})) \text{ if } i \ge 1.\end{cases}$$

The size of transducer $T_m$ can also be reduced by computing:

$$T_m = \psi(R_m \circ R_{m-1} \cdots R_1 \circ D).$$

The old approach:

$$T'_m = R'_m \circ R'_{m-1} \cdots R'_1 \circ D'.$$

has several disadvantages. The size of $R'_i$ for $1 \le i \le m$ increases considerably since the auxiliary labels introduced in each transducer have to be taken into account in all others. This fact limits the number of transducers that can be composed with $D$.

## 4 Application and Results

We will now apply our algorithm to transducers involved in speech recognition. Transducer $D$ represents the pronunciation dictionary and possesses the fundamental property. The set of transitions of $D$ is defined as

$$E = E_0 \uplus \{(f, \#, x, 0, w)\}$$

where $f$ is the unique final state of $D$, 0 is the unique initial state of $D$, $x$ is any symbol and # is a symbol representing the end of a word. All transitions $t \in E_0$ are such that $i(t) \neq \#$. Any path $\pi$ in $E_0^*$ is acyclic. The transducer $R$ representing a phonological rule is constructed to fulfill condition $(C)$ of the fundamental theorem. The transducer $D$ represents a French dictionary with 20000 words and their pronunciations. The transducer $R$ represents the phonological rule that handles *liaison* in the French language. This *liaison*, which is represented by a phoneme appearing at the end of some words, must be removed when the next word begins with a consonant since the *liaison* phoneme is never pronounced in that case. However, if the next word begins with a vowel, the liaison phoneme may or may not be pronounced and thus becomes optional.

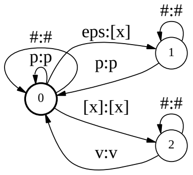


Figure 2: *Transducer used to handle the optional* liaison *rule.*

Figure 2 shows the transducer that handles this rule. In the figure, $p$ denotes all phonemes, $v$ the vowels and $[x]$ the liaison phonemes.

Table 1 shows the results of our algorithm using the dictionary and the phonological rule previously described.

| **Transducer** | **States** | **Transitions** |
|---|---|---|
| $D$ | 115941 | 136001 |
| $\psi(D)$ | 17607 | 42140 |
| $R \circ D$ | 115943 | 151434 |
| $\psi(R \circ D)$ | 17955 | 50769 |
| $R \circ \psi(D)$ | 17611 | 53209 |
| $\psi(R \circ \psi(D))$ | 17587 | 49620 |

Table 1: Size reduction on a French dictionary

As we can see in Table 1, the operator $\psi$ produces a smaller transducer in all the cases considered here.

## 5 Conclusion and future work

We have been able to disambiguate an important class of cyclic and ambiguous transducers, which allows us to apply the determinization algorithm (Mohri, 1997); and then to reduce the size of those transducers. With our new approach, we do not have to take into account the number of transducers $R_i$ and their auxiliary labels as was the case with the approach used before. Thus, new transducers $R_i$ such as phonological rules can be easily inserted in the chain.

The major disadvantage of our approach is that disambiguating a transducer increases its size systematically. Our future work will consist of developing a more effective algorithm for disambiguating an acyclic transducer.

## References

J. Berstel. 1979. *Transductions and Context-Free Languages.* Teubner Studienbucher, Stuttgart, Germany.

G. Boulianne, J. Brousseau, P. Ouellet, and P. Dumouchel. 2000. French large vocabulary recognition with cross-word phonology transducers. In *Proceedings ICASSP 2000*, June. Istanbul, Turkey.

S. Eilenberg. 1974-1976. *Automata, Language and Machines*, volume A-B. Academic Press, New York.

R. Kaplan and M. Kay. 1994. Regular models of phonological rule systems. *Computational linguistics*, 20(3):331–378.

K. Koskenniemi. 1990. Finite state parsing and disambiguation. In *Proceedings of the 13th International Conference on Computational Linguistics (COLING'90)*, volume 2. Helsinki, Finland.

M. Mohri, M. Riley, D. Hindle, A. Ljolje, and F. Pereira. 1998. Full expansion of context-dependent networks in large vocabulary speech recognition. In *Proceedings of the International Conference on Acoustics, Speech, and Signal*

*Proceesing(ICASSP' 98)*. Seattle, Washington.

M. Mohri. 1997. Finite-state transducers in language and speech processing. *Computational linguistics*, 23(2).

F. Pereira and M. Riley, 1997. *Speech recognition by composition of weighted finite automata*. Emmanuel Roche and Yves Schabes, Cambridge, Massachusetts, a bradford book, the mit press edition.

Nasser Smaili. 2001. *Désambiguïsation de transducteurs en reconnaissance de la parole*. Université du Québec à Montréal.