CS–2006–30

On the definition of stochastic λ–transducers

Stavros Konstantinidis and Nicolae Santean

Technical Report 30

David R. Cheriton School of Computer Science
University of Waterloo

2006

On the definition of stochastic λ-transducers[1]

Stavros Konstantinidis[2]
Department of Mathematics and Computing Science
Saint Mary's University
Halifax, Nova Scotia B3H 3C3, Canada
s.konstantinidis@smu.ca

Nicolae Santean
David R. Cheriton School of Computer Science
University of Waterloo
200 University Avenue West
Waterloo, Ontario, Canada N2L 3G1
nsantean@cs.uwaterloo.ca

Abstract. We propose a formal definition for the general notion of stochastic transducer, called stochastic λ-transducer. Our definition is designed with two objectives in mind: (i) to extend naturally the established notion of stochastic automaton with output – as defined in the classic books of Paz (1971) and Starke (1972) – by permitting pairs of input-output words of different lengths; (ii) to be compatible with the more general notion of weighted transducer so that one can apply tools of weighted transducers to address certain computational problems involving stochastic transducers. The new transducers can be used to model stochastic input-output processes that cannot be modeled using classic stochastic automata with output.

Key words: Probabilistic transducer, Probabilistic automaton, Stochastic transducer, Stochastic automaton, Stochastic transduction, Weighted transducer, Transducer, Automaton.

1 Introduction

Many data processes are modeled via certain stochastic (probabilistic) systems that describe the desired input-output relationships of the process. In many cases these systems are, or can be, represented by specific finite-state transducers. In this paper we propose a formal definition for the general notion of stochastic transducer, called stochastic λ-transducer. Our definition is designed with two objectives in mind:

1. to extend naturally the established notion of stochastic automaton with output – as defined in the classic books of Paz (1971) and Starke (1972) – by permitting pairs of input-output words of different lengths;
2. to be compatible with the more general notion of weighted transducer so that one can apply algorithmic tools of weighted transducers to address certain computational problems involving stochastic transducers.

---

[1] Research supported by the Natural Sciences and Engineering Research Council of Canada.

[2] Corresponding author.

In the case of classic stochastic automata with output, transitions are of the form $(p, a/b, q)$, where p and q are states and a, b are input and output alphabet symbols. Our definition extends this concept by allowing a and/or b to be equal to λ, where λ is the empty word. In effect this means that, for a given input word, a stochastic λ-transducer could output a word of different length. This capability allows one to model stochastic input-output processes that cannot be modeled using classic stochastic automata with output – see for example the Zigangirov channel in Sections 2 and 3. The price for the extra capability is a slight increase in the constraints to be obeyed when defining transitions, and that the correctness proof is not immediate – correctness here means that the constraints on defining transitions must ensure that, for any given input word, the probability that at least one of the possible words will be outputted is equal to 1.

In the next section we present the definition of stochastic λ-transducers and discuss how these objects operate. In Section 3, we demonstrate the validity of our definition in terms of two meaningful examples. Section 4 contains the proof of correctness of our definition, and the last section discusses some applications and directions for future research.

2 Definition of Stochastic (Probabilistic) λ-Transducers

For any countable set D, we denote by ProbMeas(D) the set of all (discrete) probability measures on D. Obviously, for any element μ in ProbMeas(D) we have that

$$\sum_{z \in D} \mu(z) = 1 \text{ and } 0 \leq \mu(z) \leq 1, \text{ for any } z \in D.$$

Let Σ and Δ be alphabets, and let λ be the empty word over either of the two alphabets – this creates no confusion here. The expressions Σ_λ and Δ_λ denote $\Sigma \cup \{\lambda\}$ and $\Delta \cup \{\lambda\}$, respectively. A *total stochastic transduction* (or *total probabilistic transduction*) is a mapping

$$\tau : \Sigma^* \to \text{ProbMeas}(\Delta^*);$$

that is, for every word w in Σ^*, the function $\tau_w = \tau(w)$ is a probability measure on Δ^*, which implies

$$\sum_{u \in \Delta^*} \tau_w(u) = 1.$$

As in the case of ordinary (non-stochastic) transductions [1], stochastic ones are also meant to model input-output processes, for some input alphabet Σ and output alphabet Δ. Moreover, in a stochastic transduction τ the quantity $\tau_w(u)$ is equal to

$$\Pr\{\text{output} = u \mid \text{input} = w\},$$

namely the probability that the output word will be u given that the input word is w. Next we give our definition of stochastic λ-transducer.

The definition

A *stochastic λ-transducer $\mathcal{C}$* is defined by a tuple $(\Sigma, \Delta, \Phi, \varphi, T, \beta)$ consisting of the input and output alphabets Σ and Δ, respectively, a finite nonempty set of states Φ, a probability measure φ on Φ for the choice of the start state, a finite set T of transitions (labeled edges) of the form $(p, x/y, q)$, where p and q are states and $x \in \Sigma_\lambda$ and $y \in \Delta_\lambda$, and a function β that assigns a probability value in $(0, 1]$ to every transition. In addition, Conditions (1), (2), and (3) must be satisfied as explained below. To be compatible with [7], we use the notation $H_{p,x}(q, y)$ for the value $\beta(p, x/y, q)$ – we shall switch to the matrix notation of [5] when we get to the correctness proof of our definition. This value is the probability that the λ-transducer will follow the transition from p to q with label x/y, given that the current state is p and the non-consumed part of the input word starts with x. Moreover, we define $H_{p,x}(q,y)$ to be 0 when there is no transition $(p, x/y, q)$. In mathematical notation, for fixed p and x, and for any pair (q,y) in $\Phi \times \Delta_\lambda$, we have

$$H_{p,x}(q,y) = \Pr\{\text{output} = y, \text{next-state} = q \mid \text{input-starts-with } x, \text{current-state} = p\}.$$

For technical reasons, we assume that the λ-transducer appends the special symbol $\$ \notin \Sigma$ at the end of every input word, and that $(p, \$/\lambda, p)$ is a transition, for each state p. The following conditions must be satisfied, for all states p and input symbols $a \in \Sigma$.

$$\sum_{q\in\Phi, y\in\Delta_\lambda} (H_{p,a}(q,y) + H_{p,\lambda}(q,y)) = 1. \tag{1}$$

$$H_{p,\$}(p,\lambda) = 1 - \sum_{q\in\Phi, y\in\Delta_\lambda} H_{p,\lambda}(q,y). \tag{2}$$

Another requirement is that the λ-transducer contains no closed set of states K such that for all states p in K

$$\sum_{q\in\Phi, y\in\Delta_\lambda} H_{p,\lambda}(q,y) = 1. \tag{3}$$

A set of states K is *closed*, if for any two states p and q in K there is a path from p to q, and for every states p in K and q in $\Phi \setminus K$ there is no path from p to q [5]. The above requirement ensures that the λ-transducer will ultimately consume the entire input word, that is, it will never enter a closed set of states in which the transitions consume no input.

The operation

We discuss now concepts related to the operation of a stochastic λ-transducer $\mathcal{C}$. A $\mathcal{C}$-*event*, or simply *event* when $\mathcal{C}$ is understood, is an expression ζ of the form $(x_1/y_1)p_1 \cdots (x_n/y_n)p_n$, where $n \geq 1$, and each p_i is a state, and each pair (x_i/y_i) is in $\Sigma_\lambda \times \Delta_\lambda$, with x_n possibly being equal to \$, and $x_n \neq \lambda$. This event describes a possible path that the λ-transducer can follow on the input $x_1 \cdots x_n$ starting from some state in Φ. If that state is p_0, say, then the probability of the event is defined to be

$$H_{p_0}(\zeta) = H_{p_0,x_1}(p_1,y_1)\, H_{p_1,x_2}(p_2,y_2) \cdots H_{p_{n-1},x_n}(p_n,y_n).$$

The expected probability of the event ζ when the start state is not specified is

$$H(\zeta) = \sum_{p_0\in\Phi} \varphi(p_0) H_{p_0}(\zeta).$$

As in the classic definition of stochastic automata with output [5, 7] we make no assumptions about final states. Of course, one can specify that some states of a stochastic λ-transducer are final; however, then it is not clear what the stochastic meaning of the accepting versus non-accepting paths would be.

As with ordinary transducers, stochastic ones admit graph representations. An example is shown in Figure 1. Let's assume that $\Sigma = \{a, b\}$. Given the input b at state 1 the λ-transducer could output

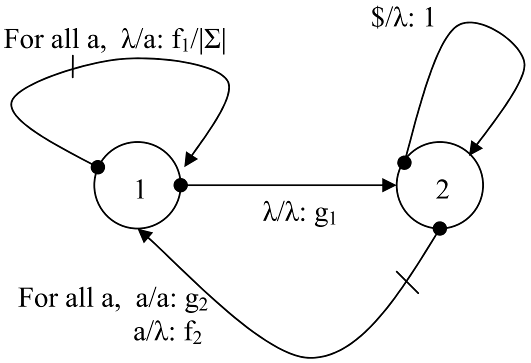

Figure 1: The Zigangirov channel. A stroke on an arrow indicates multiple transitions as indicated in the label of the arrow. A label of the form "$x/y : t$" between two states p and q represents the fact $H_{p,x}(q, y) = t$. We assume that $\Sigma = \Delta$ and $f_1 + g_1 = f_2 + g_2 = 1$ – see also Section 3 for further explanations.

the word ab using one of the following four possible events

$$\begin{aligned}
\zeta_1 &= (\lambda/a)1(\lambda/b)1(\lambda/\lambda)2(b/\lambda)1(\lambda/\lambda)2(\$/\lambda)2 \\
\zeta_2 &= (\lambda/a)1(\lambda/\lambda)2(b/\lambda)1(\lambda/b)1(\lambda/\lambda)2(\$/\lambda)2 \\
\zeta_3 &= (\lambda/a)1(\lambda/\lambda)2(b/b)1(\lambda/\lambda)2(\$/\lambda)2 \\
\zeta_4 &= (\lambda/\lambda)2(b/\lambda)1(\lambda/a)1(\lambda/b)1(\lambda/\lambda)2(\$/\lambda)2
\end{aligned}$$

One can verify that the probability of the event ζ_3 starting form state 1 is $H_1(\zeta_3) = (f_1/2)g_1^2 g_2$.

The correctness

For each nonempty word w that possibly ends with \$, we define

Z_w to be the set of all $\mathcal{C}$-events $(x_1/y_1)p_1 \cdots (x_n/y_n)p_n$ with $x_1 \cdots x_n = w$.

In addition, for any word u over Δ, let

$Z_{w,u}$ be the set of $\mathcal{C}$-events as above such that $w = x_1 \cdots x_n$ and $u = y_1 \cdots y_n$.

Theorem 1 *Let $\mathcal{C}$ be a stochastic λ-transducer. For any given state i_0 and any nonempty word w over Σ that possibly ends with \$, the event probability function H_{i_0} is indeed a probability measure on the set of $\mathcal{C}$-events Z_w, that is,*

$$\sum_{\zeta \in Z_w} H_{i_0}(\zeta) = 1.$$

The proof of Theorem 1 is given in Section 4. The theorem says that being in any state and for any given input w, the probability that one of the possible events in Z_w will occur is equal to 1. This establishes the correctness of our constraints (1), (2) and (3). When the start state is not specified then it is easy to see that again

$$\sum_{\zeta \in Z_w} H(\zeta) = 1.$$

We discuss now the total stochastic transduction $|\mathcal{C}|$ specified by $\mathcal{C}$. This is defined as follows, where w is any input word in Σ^* and u is any output word in Δ^*:

$$|\mathcal{C}|_w(u) = \sum_{\zeta \in Z_{w\$,u}} H(\zeta)$$

The quantity $|\mathcal{C}|_w(u)$ is the probability that the output is u, given that the input is w, and is equal to the sum of the probabilities of all the possible events in $Z_{w\$,u}$. The following result establishes the fact that $|\mathcal{C}|$ is indeed a total stochastic transduction.

Theorem 2 *Let $\mathcal{C}$ be a stochastic λ-transducer. For each word w over Σ, the function $|\mathcal{C}|_w$ is a probability measure on Δ^*, that is,*

$$\sum_{u \in \Delta^*} |\mathcal{C}|_w(u) = 1.$$

Theorem 2 follows from Theorem 1 when we observe the following facts, where w is an input word in Σ^* and i_0 is any state.

- $Z_{w\$} = \cup_u Z_{w\$,u}$.
- If $u \neq u'$ then $Z_{w\$,u} \cap Z_{w\$,u'} = \emptyset$.
- $\sum_{u \in \Delta^*} \sum_{\zeta \in Z_{w\$,u}} H_{i_0}(\zeta) = \sum_{\zeta \in Z_{w\$}} H_{i_0}(\zeta)$.

3 Two Examples

In this section we demonstrate the relevance of our definition with two meaningful examples. The first example is the stochastic channel of Zigangirov [9] – revisited recently in [2] – which permits insertions and deletions of symbols to occur in an input word. The second example is the important concept of stochastic (or probabilistic) automaton with output, as presented in [5, 7], in which any transition label is of the form a/b, where a and b are symbols in the input and output alphabets, respectively – this concept had been defined by Shannon [6] as well to model communication channels that permit substitutions of symbols in an input word by different symbols. In the literature of stochastic processes, the term channel is normally used in the general intuitive sense of a process that outputs a word with a certain probability in response to a given input word. In this context, our definition of stochastic λ-transducer provides a formal method for defining various stochastic channels. We note that, in the literature of formal languages and coding theory, one can also find the concept of non-stochastic (that is, combinatorial) channel – see [3], for instance.

The Zigangirov channel is defined as follows [9]: The channel receives input symbols $a_1, a_2, \ldots$ in Σ. The input $a_1a_2\cdots$ can be written as $\lambda a_1 \lambda a_2 \lambda \cdots$. For each λ-position and for each symbol a_i, the channel can make changes as follows.

- For each λ-position, we have the quantities

  $$\Pr\{\text{no insertion}\} = g_1 \text{ and } \Pr\{\text{insertion of } i \text{ symbols}\} = g_1 f_1^i,$$

  such that $f_1 + g_1 = 1$, and $\Pr\{\text{inserting } u\} = \Pr\{\text{inserting } v\}$, for any words u and v of the same length. Hence, for any word u of length i, we have that $\Pr\{\text{inserting } u\} = g_1 f_1^i / |\Sigma|^i$. This implies that, for each λ-position, $\Pr\{\text{insertion of a nonempty word}\} = f_1$.

- For each input symbol a, we have the quantities

  $$\Pr\{\text{no deletion}\} = g_2 \text{ and } \Pr\{\text{deletion of } a\} = f_2,$$

  such that $f_2 + g_2 = 1$.

In Figure 1 we show that stochastic λ-transducers can be used to model the Zigangirov channel. We assume here that the state 1 is the start state, that is, any input word to the λ-transducer is processed starting at state 1.

The next example demonstrates that our definition of stochastic λ-transducer is a natural extension of the concept of stochastic automaton (with output) defined in the classic books [5, 7]. Indeed, in [7] for instance, a stochastic automaton consists of the alphabets Σ and Δ, the state set Φ, and a function H that maps any pair (p, a) in $\Phi \times \Sigma$ onto $H_{p,a}$ which is a probability measure on $\Phi \times \Delta$, that is, $H_{p,a}(q, b)$ is the probability that the automaton will go to state q and output the symbol b, given that the current state is p and the input symbol is a. Moreover, for any p and a,

$$\sum_{q\in\Phi, b\in\Delta} H_{p,a}(q, b) = 1.$$

Thus, at each step, the automaton consumes exactly one input symbol and outputs exactly one output symbol. The book [7] does not use the concept of event that we use here, but extends the function H to words such that, for any words $w = a_1 \cdots a_n \in \Sigma^n$, $u = b_1 \cdots b_n \in \Delta^n$, and state word $p = p_1 \cdots p_n \in \Phi^n$, for some $n \geq 0$, we have that

$$H_{p_0,w}(p, u) = H_{p_0,a_1}(p_1, b_1) \cdots H_{p_{n-1},a_n}(p_n, b_n).$$

One can verify that our definition of stochastic λ-transducer reduces to the classic definition of stochastic automaton when we omit any component that involves the empty word λ. We note that the correctness of the classic definition is not an issue, that is, it is not difficult to see that, for any word w of length n and for any state p_0, the sum of the probabilities $H_{p_0,w}(p, u)$, for all p and u, is equal to 1. On the other hand, in our extended version of stochastic machine, the proof of Theorem 1 is not immediate – see Section 4.

4 Proof of Theorem 1

For the proof of Theorem 1, we use the notation of the previous section as well as the matrix representation of stochastic λ-transducers (following [5]). Let s be the number of states in Φ. Without loss of generality, assume that $\Phi = \{1, \ldots, s\}$. For any x in $\Sigma_\lambda \cup \{\$\}$ and y in Δ_λ, we define the $s \times s$ matrix $A(x/y)$ such that each entry (i, j) of the matrix is

$$[A(x/y)]_{i,j} = H_{i,x}(j, y).$$

Thus, the entry (i, j) of the matrix $A(x/y)$ is the probability that the λ-transducer will go to state j and output y, given that the current state is i and the input word to be consumed starts with x. These matrices are sufficient to define the stochastic λ-transducer in question. We also need the following notation

$$\begin{aligned} A_x &= \sum_y A(x/y), \\ P_i(x) &= \sum_{j=1}^{s} [A_x]_{i,j}, \text{ for any state } i. \end{aligned}$$

By definition, for any symbol a in $\Sigma \cup \{\$\}$, we have

$$P_i(a) = \sum_{j=1}^{s} \sum_{y} H_{i,a}(j,y),\ P_i(\lambda) = \sum_{j=1}^{s} \sum_{y} H_{i,\lambda}(j,y),$$

which implies that $P_i(a) + P_i(\lambda) = 1$, for any state i. Thus the matrix $A_x + A_\lambda$ is *stochastic* (each row sums to 1) and each of A_x and A_λ is *sub-stochastic* (each row sums to at most 1).

In the next lemma I_s denotes the $s \times s$ unit matrix, that is, the unique matrix with the property $I_s A = A I_s = A$, for all $s \times s$ matrices A.

Lemma 1 *The matrix $I_s - A_\lambda$ is invertible, which implies that the following equality is valid*

$$I_s + A_\lambda + A_\lambda^2 + \cdots = (I_s - A_\lambda)^{-1}.$$

Proof. It is sufficient to show that all the eigenvalues, say $t_1, \ldots, t_s$, of A_λ are such that $|t_i| < 1$. This ensures that the sum $\sum_{r=0}^{\infty} A_\lambda^r$ is well-defined and equal to $(I_s - A_\lambda)^{-1}$. Recall that the eigenvalues of an $s \times s$ matrix A are the s roots of the equation $\det(A - tI_s) = 0$, where the unknown variable is t.

We shall view the λ-transducer $\mathcal{C}$ as a labeled graph. Let $\mathcal{C}_\lambda$ be the graph that results from $\mathcal{C}$ when we keep all states in Φ and only the transitions of the form $(i, \lambda/y, j)$. Then, let $\mathcal{C}'_\lambda$ be the graph that results when we add the following elements in $\mathcal{C}_\lambda$:

- a new state 0;
- the transition $(0, \lambda/\lambda, 0)$ with weight $\beta(0, \lambda/\lambda, 0) = 1$;
- for each state $i \in \Phi$, the transition $(i, \lambda/\lambda, 0)$ with weight $\beta(i, \lambda/\lambda, 0) = 1 - P_i(\lambda)$

It is evident that $\mathcal{C}'_\lambda$ has a single closed set of states which is equal to $\{0\}$. Let $\mathcal{C}''_\lambda$ be the graph that results from $\mathcal{C}'_\lambda$ when we replace, for every pair of states (i, j), the set (if nonempty) of transitions $(i, \lambda/y_1, j), (i, \lambda/y_2, j), \ldots$ with a single transition (i, g, j) where g is the sum of the weights of these transitions. Note that, for fixed i, the sum of g's corresponding to all j's is equal to $P_i(\lambda)$. It is evident that $\mathcal{C}''_\lambda$ contains a single closed set of states, which is $\{0\}$. Let B be the $(s+1) \times (s+1)$ incidence matrix of $\mathcal{C}''_\lambda$, that is, the matrix whose entry (i, j) is equal to the weight q of the edge (i, q, j) of $\mathcal{C}''_\lambda$. Then the structure of B is as follows

$$B = \left[\begin{array}{rcccc} 1 & 0 & & \cdots & 0 \\ 1 - P_1(\lambda) & & & & \\ \vdots & & \Large A_\lambda & & \\ 1 - P_s(\lambda) & & & & \end{array}\right]$$

Obviously B is a stochastic matrix. Moreover, for every variable t,

$$\det(B - tI_{s+1}) = (1 - t) \cdot \det(A_\lambda - tI_s).$$

This implies that the eigenvalues of B are 1 and $t_1, \ldots, t_s$ (which are the eigenvalues of A_λ). As B is the incidence matrix of a graph with a single closed set of states, the eigenvalue 1 is unique (see page 98 of [5]), that is, each $|t_i|$ is not equal to 1. Moreover, as A_λ is sub-stochastic, each $|t_i|$ is less than 1, as required. $\square$

Lemma 2 *For any state i_0 and any input symbol a in $\Sigma \cup \{\$\}$, we have that*

$$\sum_{\zeta \in Z_a} H_{i_0}(\zeta) = 1.$$

Proof. For any integer $r \geq 0$ and $a \in \Sigma \cup \{\$\}$, let $Z_a^{(r)}$ be the set of all events of the form

$$\zeta = (\lambda/y_1)i_1 \cdots (\lambda/y_r)i_r(a/y_{r+1})i_{r+1}.$$

Then,

$$Z_a = \cup_{r=0}^{\infty} Z_a^{(r)} \text{ and } \sum_{\zeta \in Z_a} H_{i_0}(\zeta) = \sum_{r=0}^{\infty} \sum_{\zeta \in Z_a^{(r)}} H_{i_0}(\zeta).$$

Now fix the terms $y_1, \ldots, y_{r+1}$ and consider all $\zeta = (\lambda/y_1)i_1 \cdots (\lambda/y_r)i_r(a/y_{r+1})i_{r+1}$ in $Z_a^{(r)}$ where only the terms $i_1, \ldots, i_{r+1}$ are arbitrary. Then the sum of $H_{i_0}(\zeta)$ over all these ζ's is equal to

$$f_{i_0} A(\lambda/y_1) \cdots A(\lambda/y_r) A(a/y_{r+1}) g,$$

where g is the $s \times 1$ matrix $(1, \ldots, 1)^{\mathrm{T}}$, and f_{i_0} is the $1 \times s$ matrix $(0, \ldots, 1, \ldots, 0)$ with a single 1 at position i_0. Suppose that $\Delta_\lambda = \{c_0, \ldots, c_m\}$. If now we allow the terms $y_1, \ldots, y_{r+1}$ to be arbitrary, then

$$\sum_{\zeta \in Z_a^{(r)}} H_{i_0}(\zeta) = f_{i_0}(A(\lambda/c_0) + \cdots + A(\lambda/c_m))^r (A(a/c_0) + \cdots + A(a/c_m)) g,$$

which is equal to $f_{i_0} A_\lambda^r A_a g$. Hence,

$$\begin{aligned}
\sum_{\zeta \in Z_a} H_{i_0}(\zeta) &= \sum_{r=0}^{\infty} (f_{i_0} A_\lambda^r A_a g) \\
&= f_{i_0} (\sum_{r=0}^{\infty} A_\lambda^r) A_a g \\
&= f_{i_0} (I_s - A_\lambda)^{-1} A_a g \text{ [using Lemma 1].}
\end{aligned}$$

It follows now that $A_a g = (I_s - A_\lambda) g$, as $P_i(a) = 1 - P_i(\lambda)$ for all states i. Hence,

$$\sum_{\zeta \in Z_a} H_{i_0}(\zeta) = f_{i_0} (I_s - A_\lambda)^{-1} (I_s - A_\lambda) g = 1,$$

as required. □

Proof of Theorem 1. We use induction on the length of w. The base case where the length of w is 1 follows from Lemma 2. Assume the statement holds for all words of length up to n. Let w be a word of length $n+1$. Then $w = ux$ for some word u of length n and symbol x in $\Sigma \cup \{\$\}$. Each event ζ in Z_w, when viewed as a word, can be written uniquely as $\zeta = \zeta_1 \zeta_2$ where ζ_1 is in Z_u and ζ_2 is in Z_x. Moreover,

$$H_{i_0}(\zeta) = H_{i_0}(\zeta_1) \cdot H_{\ell(\zeta_1)}(\zeta_2),$$

where $\ell(\zeta_1)$ is the last state in the event ζ_1. Conversely, each $\zeta_1 \in Z_u$ and $\zeta_2 \in Z_x$ defines the event $\zeta_1\zeta_2$ in Z_w. Hence,

$$\begin{aligned}\sum_{\zeta\in Z_w} H_{i_0}(\zeta) &= \sum_{\zeta_1\in Z_u}\sum_{\zeta_2\in Z_x} H_{i_0}(\zeta_1)H_{\ell(\zeta_1)}(\zeta_2)\\ &= \sum_{\zeta_1\in Z_u}(H_{i_0}(\zeta_1)\cdot\sum_{\zeta_2\in Z_x} H_{\ell(\zeta_1)}(\zeta_2))\\ &= \sum_{\zeta_1\in Z_u} H_{i_0}(\zeta_1) \text{ [using Lemma 2]}\\ &= 1 \text{ [using the induction hypothesis].}\end{aligned}$$

□

5 Discussion

As demonstrated in Section 3, our definition of stochastic λ-transducer constitutes a natural generalization of the classic definition of stochastic automaton with output [5, 7], and can be used to model, for instance, channels involving insertions and deletions of symbols. Recall that the second objective of our definition is to be compatible with the more general notion of weighted transducer – see for example [4]. Using algorithmic tools for these objects we can address certain computational problems involving stochastic λ-transducers.

For example, the maximum likely event problem is as follows. Let $\mathcal{C}$ be a stochastic λ-transducer and let u be a given word. We wish to compute an event

$$\zeta = (x_1/y_1)p_1\cdots(x_n/y_n)p_n$$

of $\mathcal{C}$ such that $u = y_1\cdots y_n$ and, for any event $\zeta' = (x'_1/y'_1)p'_1\cdots(x'_m/y'_m)p'_m$ with $u = y'_1\cdots y'_m$, we have that $H(\zeta) \geq H(\zeta')$. This problem can be solved by reducing it to the best alignment problem for weighted transducers, [4], as follows.

1. Let $\mathcal{C}_1$ be the weighted transducer resulting when we replace every weight t of $\mathcal{C}$ with $-\log t$.
2. Let $\mathcal{C}_2$ be the weighted transducer that results when we add in $\mathcal{C}_1$ a single start state σ, say, and transitions (σ, c, q) for every state q of $\mathcal{C}_1$, such that c is equal to $-\log\varphi(q)$, where φ is the start state distribution of $\mathcal{C}$.
3. Let $\mathcal{C}_3$ be the weighted transducer that results by intersecting (or composing) $\mathcal{C}_2$ with the finite automaton that accepts the single word u such that in every event $\zeta = (x_1/y_1)p_1\cdots(x_n/y_n)p_n$ of $\mathcal{C}_3$, with p_n being a final state, we have that $y_1\cdots y_n = u$. This step is possible using a state Cartesian-product construction on automata – see for example [4].
4. Compute a shortest path in $\mathcal{C}_3$ from the start state σ to a final state and return the event specified by the path.

A more intricate problem is the maximum likely input problem. Again, let $\mathcal{C}$ and u be as before. We wish to compute a word w such that $|\mathcal{C}|_w(u) \geq |\mathcal{C}|_{w'}(u)$, for all input words w'. This problem is usually reduced to the problem of determinizing weighted automata [4].

The above problems are useful in situations where $\mathcal{C}$ is viewed as a channel and one wants to compute, for a given word u, the most likely input word that resulted into u via the channel. The channel could be, for example, an ordinary digital communications channel, or a typesetter conveying words from his/her mind into a computer [8]. In such cases, the required input words usually belong to a certain language (the dictionary), say L. When L is a regular language, we can always restrict $\mathcal{C}$ to a transducer whose domain (input part) is equal to L without affecting the weights.

There are a few directions for further research arising from this work. For example, (1) investigate state minimization methods for stochastic λ-transducers using as a guide the existing tools in [5, 7], (2) consider determinization methods specific to stochastic λ-transducers, and (3) define and study stochastic λ-transducers with final states.

References

[1] J. Berstel. *Transductions and context-free languages*. B. G. Teubner, Stuttgart, 1979.

[2] M. C. Davey, D. J. C. MacKay. Reliable communication over channels with insertions, deletions, and substitutions. *IEEE Trans. Info. Theory* **47** (2001), 687–698.

[3] S. Konstantinidis. Transducers and the properties of error-detection, error-correction and finite-delay decodability. *Journal Of Universal Computer Science* **8** (2002), 278-291.

[4] Mehryar Mohri. Edit-distance of weighted automata: general definitions and algorithms. *International Journal of Foundations of Computer Science* **14**(6) (2003), 957–982.

[5] A. Paz. *Introduction to probabilistic automata*. Academic Press, New York and London, 1971.

[6] C. Shannon, W. Weaver. *The mathematical theory of communication*. University of Illinois Press, Urbana, Chicago, London, 12th printing, 1971.

[7] P. Starke. *Abstract automata*. North-Holland, Academic Press, 1972.

[8] J. Xu. *Formalizations of error models with applications to spelling error correction*. MSc thesis, Department of Mathematics and Computing Science, Saint Mary's University, Canada, 2004.

[9] K. S. Zigangirov. Sequential decoding for a binary channel with drop-outs and insertions, *Probl. Pered. Inform.* **5** (1969), 23–30.