Electronic Notes in Theoretical Computer Science 46 (2001)
URL: http://www.elsevier.nl/locate/entcs/volume46.html 16 pages

Image Representation using Distributed Weighted Finite Automata

Y. Sivasubramanyam [1,2] and Kamala Krithivasan [3]

Department of Computer Science and Engineering
Indian Institute of Technology, Madras
Chennai, India

Abstract

Weighted finite automata (WFA) define real functions, in particular, grayness functions of graytone images. Inference algorithm that converts an arbitrary function (graytone image) into a WFA that can regenerate it is given in [7]. In this paper we define the theoretical construct of Cooperating Distributed Weighted Finite Automata with n-components(n-WFA) and study the power of this construct in various modes of acceptance. We give an inference algorithm and the de-inference algorithm for the n-WFA.

1 Introduction

Weighted finite automata (WFA) have been introduced in [7]. They compute real functions of n-variables [6], more precisely functions $[0,1]^n \to \mathbb{R}$. For $n = 2$ such a function can be interpreted as the grayscale function of an image. For a theoretical study of WFA see [6]. In [7] an inference algorithm for the WFA is given, that for a function (image) given in table (pixel) form finds a WFA with a small number of states that approximates the given function. A recursive algorithm that infers a relatively small WFA which provides a good approximation of any given real life image has been given in [8]. [9,10] give a comprehensive treatment of WFA and their applications to image compression.

Distributed computing plays a major role in this era of computing. The theory of grammar systems is a grammatical model for the distributed computation. A grammar system is a set of grammars working in unison, according to a specified protocol, to generate one language. The grammar systems can be

[1] The financial support from the Department of Science and Technology, India is acknowledged

[2] Email: siva@cs.iitm.ernet.in

[3] Email: kamala@iitm.ernet.in

either sequential(Cooperating Distributed) or parallel(Parallel Communicating) in nature.A comprehensive treatment of grammar systems and a survey of the recent developments in this area can be found in [2]. The notion of sequential grammar systems was extended to automata in [1,12].

In this paper we define a new theoretical construct namely, the Cooperating Distributed Weighted Finite Automata with n-components (n-WFA) which is a collection of weighted finite automata working sequentially to accept the input string. Here the protocol followed is that of the Cooperating Distribution. We study the power of this construct in various modes of acceptance. We also give an inference algorithm and the de-inference algorithm for the n-WFA and illustrate with examples how images are represented using n-WFA.

When the images are represented using n-WFA the weight matrices will be sparse and hence the amount of storage required will be small. This might give a better compression ratio. Also the inferencing and de-inferencing algorithm will be faster in most cases as the matrix computations involved in the inference and de-inference algorithms are much faster than in the classical case.

In Section 2 we give the preliminary definitions of the weighted finite automata and its applications to digital images. In Section 3 we introduce the construct of Cooperating distributed weighted finite automata and study the acceptance power of this construct in various modes of acceptance. In this section we also deal with the representation of gray-scale images using n-WFA and give an inference algorithm and the de-inference algorithm for the n-WFA. Section 4 deals with the conclusions of this paper.

2 Weighted Finite Automata and Gray-Scale Images

In this section we give the basic definitions needed for this paper.

Definition 2.1 *A* **weighted finite automaton** *M [10] is specified by*

(i) *Q a finite set of states.*

(ii) *Σ a finite set of alphabets.*

(iii) *$W_\alpha : Q \times Q \longrightarrow \mathbb{R}$ for all $\alpha \in \Sigma \bigcup \{\epsilon\}$, the weights of edges labeled α.*

(iv) *$I : Q \longrightarrow (-\infty, \infty)$, the initial distribution.*

(v) *$F : Q \longrightarrow (-\infty, \infty)$, the final distribution.*

Here W_α is an $n \times n$ matrix where $n =\mid Q \mid$. I is considered to be an $1 \times n$ row vector and F is considered to be an $n \times 1$ column vector. When representing the WFAs as figure, we follow a format similar to FSAs. Each state is represented by a node in a graph. The initial distribution and final distribution of each state is written as a tuple inside the state. A transition labeled α is drawn as a directed arc from state p to q if $W_\alpha(p, q) \neq 0$. The weight of the edge is written in brackets on the directed arc. The notation $I_q(F_q)$ is used to refer to the initial(final) distribution of state q. $W_\alpha(p,q)$ refers to the weight of the transition from p to q. $W_\alpha(p)$ refers to the p^{th} row vector of the weight matrix W_α. It gives the weights of all the transitions from state p labeled α in a vector form. Also W_x refers to the product $W_{\alpha_1} \cdot W_{\alpha_2} \cdots W_{\alpha_k}$ where $x = \alpha_1\alpha_2\cdots\alpha_k$.

Definition 2.2 *A WFA is said to be* **deterministic** *if its underlying FSA is deterministic.*

Definition 2.3 *A WFA M defines a function $f : \Sigma^* \longrightarrow \mathbb{R}$, where for all $x \in \Sigma^*$ and $x = \alpha_1\alpha_2\cdots\alpha_k$,*

$$f(x) = I \cdot W_{\alpha_1} \cdot W_{\alpha_2} \cdots W_{\alpha_k} \cdot F$$

where the operation '$\cdot$' is matrix multiplication.

Definition 2.4 *A* **path** *P of length k is defined as a tuple $(q_0q_1\cdots q_k, \alpha_1\alpha_2\cdots\alpha_k)$ where $q_i \in Q, 0 \le i \le k$ and $\alpha_i \in \Sigma, 1 \le i \le k$ such that α_i denotes the label of the edge traversed while moving from q_{i-1} to q_i.*

Definition 2.5 *The* **weight** *of a path P is defined as*

$$W(P) = I_{q_0} \cdot W_{\alpha_1}(q_0,q_1) \cdot W_{\alpha_2}(q_1,q_2) \cdots W_{\alpha_k}(q_{k-1},q_k) \cdot F_{q_k}$$

The function $f : \Sigma^* \longrightarrow \mathbb{R}$ represented by a WFA M can be equivalently defined as follows

$$f(x) = \sum_{\text{P is a path of M labeled x}} W(P),\ x \in \Sigma^*.$$

Definition 2.6 *A function $f : \Sigma^* \longrightarrow \mathbb{R}$ is said to be* **average preserving** *if*

$$f(w) = \frac{1}{m}\sum_{\alpha\in\Sigma} f(w\alpha)$$

for all $w \in \Sigma^*$ where $m = \mid \Sigma \mid$.

Definition 2.7 *A WFA M is said to be* **average preserving** *if the function that it represents is average preserving.*

The general condition to check whether a WFA is average preserving is given in [10]. A WFA M is average preserving if and only if

$$\sum_{\alpha\in\Sigma} W_\alpha \cdot F = mF,$$

where $m = \mid \Sigma \mid$.

Definition 2.8 *A WFA is said to be* **i-normal** *[13] if the initial distribution of every state is 0 or 1 i.e. $I_{q_i} = 0$ or $I_{q_i} = 1$ for all $q_i \in Q$.*

Definition 2.9 *A WFA is said to be* **f-normal** *[13] if the final distribution of every state is 0 or 1 i.e. $F_{q_i} = 0$ or $F_{q_i} = 1$ for all $q_i \in Q$.*

Definition 2.10 *A WFA is said to be* **I-normal** *if there is only one state with non-zero initial distribution.*

Definition 2.11 *A WFA is said to be* **F-normal** *if there is only one state with non-zero final distribution.*

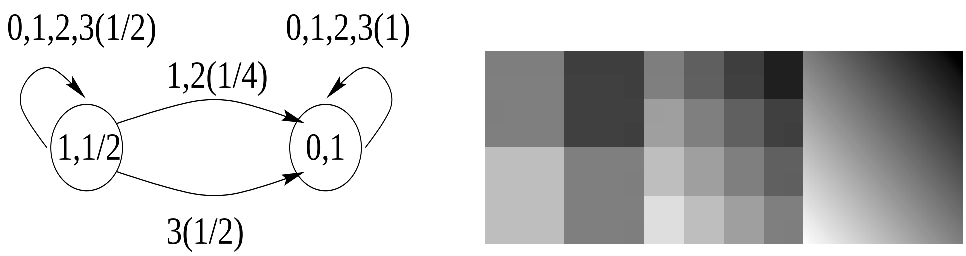
2.1 Representation of Gray-Scale Images using WFA

A gray-scale digital image of finite resolution consists of 2^m by 2^m pixels (typically $7 \leq m \leq 11$) each of which takes a real value (practically digitized to a value between 0 and $2^m - 1$, typically $m = 8$). By a multi-resolution image, we mean a collection of compatible 2^n by 2^n resolution images for $n = 0, 1, \cdots$. We will assign to each pixel at 2^n by 2^n resolution a word of the length n over the alphabet $\Sigma = \{0, 1, 2, 3\}$. A word x of length less than k will address a sub-square of resolution $2^{k'}$ by $2^{k'}$ where $k' < k$.

Then we can define our finite resolution image as a function $f_I : \Sigma^k \longrightarrow \mathbb{R}$, where $f_I(x)$ gives the value of the pixel at address x. A multi-resolution image is a function $f_I : \Sigma^* \longrightarrow \mathbb{R}$. It is shown in that for compatibility, the function f_I should be average preserving i.e.

$$f_I(x) = \frac{1}{4}[f_I(x0) + f_I(x1) + f_I(x2) + f_I(x3)].$$

A WFA M is said to represent a multi-resolution image if the function f_M represented by M is the same as the function f_I of the image.



WFA 2x2 4x4 128x128

Example 2.12 Consider the 2 state WFA shown in figure [10]. The $I = (1, 0)$ and $F = (\frac{1}{2}, 1)$ and the weight matrices are

$$W_0 = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & 1 \end{pmatrix}, \ W_1 = \begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & 1 \end{pmatrix}, \ W_2 = \begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & 1 \end{pmatrix}, \text{ and } W_3 = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ 0 & 1 \end{pmatrix}.$$

Then we can calculate the values of pixels as follows. $f(03)$ = sum of weights all paths labeled 03.

$$f(03) = 1 \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} + 1 \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot 1 = \frac{1}{8} + \frac{1}{4} = \frac{3}{8}$$

similarly for $f(123)$ we have $f(123) = \frac{1}{16} + \frac{1}{8} + \frac{1}{8} + \frac{1}{4} = \frac{9}{16}$. The images obtained by this WFA are shown for resolutions 2×2, 4×4 and 128×128 in the above figure.

Thus we have seen how WFAs can be used for representing gray-scale images.

3 Cooperating Distributed Weighted Finite Automata

3.1 Definitions

Definition 3.1 *A Cooperating Distributed Weighted Finite Automata with n-components , n-WFA is a 5-tuple $\Gamma = (Q, \Sigma, W_\alpha, I, F)$ where,*

(i) *Q is an n-tuple $(Q_1, Q_2, \cdots, Q_n)$ where each Q_i is the set of states corresponding to the i^{th} component.*

(ii) *Σ is the finite set of alphabet.*

(iii) *W_α is an n-tuple $(W^1_\alpha, W^2_\alpha, \cdots, W^n_\alpha)$ of weight matrices (weights of edges labeled α for each $\alpha \in \Sigma \cup \{\epsilon\}$ where each $W^i_\alpha : Q_{union} \times Q_{union} \longrightarrow \mathbb{R},\ 1 \leq i \leq n$*

(iv) *$I : Q_{union} \longrightarrow (-\infty, \infty)$ is the initial distribution.*

(v) *$F : Q_{union} \longrightarrow (-\infty, \infty)$ is the final distribution.*

where $Q_{union} = \cup_i Q_i$.

Each of the component WFA of the n-WFA is of the form $M_i = (Q_i, \Sigma, W^i_\alpha)$, $1 \leq i \leq n$. Note that here Q_i *'s* need not be disjoint.

Each of W^i_α is an $m \times m$ matrix where $m =\mid Q_{union} \mid$ and these matrices are sparse matrices. I is considered to be an $1 \times m$ row vector and F is considered to be an $m \times 1$ column vector. When representing the n-WFAs as a figure, we follow the format similar to that of the WFAs. The transitions from one component to another are indicated by dotted lines.

Definition 3.2 *A n-WFA is said to be* **deterministic** *if each of its component WFA is deterministic.*

Definition 3.3 *A n-WFA M defines a function $f : \Sigma^* \longrightarrow \mathbb{R}$, where for all $x \in \Sigma^*$ and $x = \alpha_1\alpha_2\cdots\alpha_k$,*

$$f(x) = I \cdot W^{i_1}_{\alpha_1} \cdot W^{i_2}_{\alpha_2} \cdots W^{i_k}_{\alpha_k} \cdot F$$

where the operation '$\cdot$' is matrix multiplication and $1 \leq i_1, i_2, i_k \leq n$.

The definitions of a path, weight of a path and average preserving function of n-WFA are defined in similar terms as those of a WFA.

We consider different modes of acceptance depending on the number of steps the system has to go through in each of the n-components. The different modes of acceptance are t-mode, *-mode, $\leq k$-mode, $\geq k$-mode, and $= k$-mode. Description of each of the above modes of acceptance is as follows: **t-mode acceptance:** The automaton which has a state with the non-zero initial distribution begins the processing of the input string. Suppose that the system starts from the component i. In the component i the system follows its transition function given by it's weight matrix W^i_α as any "stand alone" WFA. The control is transfered from the component i to component j only if the system arrives at a state $q \notin Q_i$ and $q \in Q_j$. The selection of j is nondeterministic if q belongs to more than one Q_j. This process is repeated and we accept the string if the system after reading the entire string reaches any one of the states which has a non-zero final state distribution. It does not matter in which component the system is in.

Definition 3.4 *The instantaneous description of the n-WFA (ID) in the t-mode is given by a 3-tuple (q, w, i) where $q \in Q_{union},\ w \in \Sigma^*,\ 1 \le i \le n$.*

In this ID of the n-WFA, q denotes the current state of the whole system, w the portion of the input string yet to be read and i the index of the component in which the system is currently in.

The transition between the ID's is defined as follows:

(i) $(q, aw, i) \vdash (q', w, i)$ iff $W^i_a(q, q') \neq 0$ where $q \in Q_i,\ q' \in Q_{union},\ a \in \Sigma \cup \{\epsilon\},\ w \in \Sigma^*, 1 \le i \le n$

(ii) $(q, w, i) \vdash (q, w, j)$ iff $q \in Q_j - Q_i$

Let $\vdash^*$ be the reflexive and transitive closure of $\vdash$ (when we consider as a graytone picture $\Sigma = \{0, 1, 2, 3\}$ and w is the address of a pixel).

Definition 3.5 *The language accepted by the n-WFA $\Gamma = (Q, \Sigma, W_\alpha, I, F)$ working in t-mode is defined as follows,*

$$L_t(\Gamma) = \left\{ w \in \Sigma^* \;\middle|\; \begin{array}{l} (q_0, w, i) \vdash^* (q_f, \epsilon, j) \textit{ for some } q_f \textit{ with non-zero} \\ \textit{final distribution },\ 1 \le j, i \le n \textit{ and } q_0 \in Q_i \\ \textit{also } f_\Gamma(w) = \textit{weight of the string } w \textit{ and } f_\Gamma(w) > 0 \end{array} \right\}$$

***-mode acceptance:** The automaton which has a state with the non-zero initial distribution begins the processing of the input string. Suppose the system starts the processing from the component i. Unlike the termination mode(t-mode), here there is no restriction. The automaton can transfer the control to any other component at any time if possible, i.e, if there is some j such that $q \in Q_j$ then the system can transfer the control to the component j. The selection is done nondeterministically if there is more than one j. The instantaneous description and the language accepted by the system in *-mode can be defined analogously. The language accepted in *-mode is denoted as $L_*(\Gamma)$.

$=k$-mode($\le k$-mode, $\ge k$-mode) acceptance: The component which has a state with the non-zero initial distribution begins the processing of the input string. Suppose the system starts the processing from the component i. The system transfers the control to the other component j only after the completion of exactly k($k'(k' \le k)$, $k'(k' \ge k)$) number of steps in the component i, i.e, if there is a state $q \in Q_j$ then the transition from component i to the component j takes place only if the system has already completed $k(k'(k' \leq k), k'(k' \geq k))$ steps in component i. If there is more than one choice for j the selection is done nondeterministically.

The instantaneous description of n-WFA in the above three modes of derivations and the language generated by the them are defined as follows,

Definition 3.6 *The instantaneous description of the n-WFA (ID) is given by a 4-tuple (q, w, i, j) where $q \in Q_{union}$, $w \in \Sigma^*$, $1 \leq i \leq n$, j is a non negative integer.*

In this ID of the n-WFA, q denotes the current state of the whole system, w the portion of the input string yet to be read; i the index of the component in which the system is currently in, and j denotes the number of steps for which the system has been in the ith component. We accept the strings only if the n-WFA reaches a state with non-zero final distribution in some component i after reading the string and provided it has completed k steps in the component i in the case of $= k$-mode of acceptance (it has completed some $k'(k' \leq k)$ steps in the component i in the case of $\leq k$-mode of acceptance or it has completed some $k'(k' \geq k)$ steps in the component i in the case of $\geq k$-mode of acceptance). The languages accepted by the n-WFA Γ in the respective modes are denoted as $L_{=k}(\Gamma), L_{\leq k}(\Gamma)$ and $L_{\geq k}(\Gamma)$.

3.2 Power of acceptance of different modes

Notation: The family of languages accepted by WFA is denoted by $\mathcal{L}(WFA)$.

Theorem 3.7 *For any n-WFA Γ working in t-mode, the function defined by it can be defined by a single WFA.*

Proof Let $\Gamma = (Q, \Sigma, W_\alpha, I, F)$ be a n-WFA working in t-mode where $W_\alpha = (W_\alpha^1, W_\alpha^2, \cdots, W_\alpha^n)$ and the components have states $Q_1, Q_2, \cdots, Q_n$. Consider the WFA $M = (Q', \Sigma, W'_\alpha, I', F')$ where,

$$Q' = \{[q, i] \mid q \in Q_{union},\ 1 \leq i \leq n\} \cup \{q'_0\}$$

$$I' : Q' \longrightarrow (-\infty, \infty) \text{ is an } \textbf{I-normal} \text{ initial distribution such that}$$

$$I'(q'_0) = 1 \text{ and } I'(q) = 0 \text{ for all other } q \in Q'$$

$$F' : Q' \longrightarrow (-\infty, \infty) \text{ is such that } F'(q'_0) = 0$$

$$\text{and } F'([q, i]) = F(q),\ q \in Q_{union}, 1 \leq i \leq n.$$

W'_α ,the weight matrices are defined as follows,

(i) $W'_\epsilon(q'_0, [q_0, i']) = I(q_0)$ such that $q_0 \in Q_{i'}$

(ii) for each q_k such that $W_a^i(q_j, q_k) \neq 0, a \in \Sigma \cup \{\epsilon\}, 1 \leq i \leq n$,
  (a) if $q_k \in Q_i$ then $W'_a([q_j, i], [q_k, i]) = W_a^i(q_j, q_k)$
  (b) if $q_k \in Q_j - Q_i$ then $W'_a([q_j, i], [q_k, j] = W_a^i(q_j, q_k)$

The construction of WFA clearly shows that

$$L(M) \;=\; L_t(\Gamma)$$
$$\text{and so } L_t(\Gamma) \in \mathcal{L}(WFA).$$

Moreover for any string $w = a_1a_2\cdots a_k \in \Sigma^*$ let $P = (q_0q_1\cdots q_k, a_1a_2\cdots a_k)$ be a path of length k in the n-WFA Γ. The weight of this path P is $W(p) = I_{q_0}\cdot W^{i_1}_{a_1}(q_0,q_1)\cdot W^{i_2}_{a_2}(q_1,q_2)\cdots W^{i_k}_{a_k}(q_{k-1},q_k)F_{q_k}, 1 \le i_1, i_2, \cdots i_k \le n$. The path followed by this string w in the WFA M, is given $P' = (q'_0q_0\cdots q_k, a_1a_2\cdots a_k)$ and the weight of this path P' is $W'(p') = I_{q'_0}\cdot I_{[q_0,i]}\cdot W'_{a_1}([q_0,i],[q_1,i])\cdot W'_{a_2}([q_1,i],[q_2,j])\cdots W'_{a_k}([q_{k-1},i'][q_k,j'])F_{[q_k,j']}$. By the above construction it is clear that $W(P) = W'(P')$ and so the function f_Γ defined by the n-WFA is equal to the function f_M defined by the WFA M. i.e. $f_\Gamma(w) = f_M(w)$ for $w \in \Sigma^*$.

Theorem 3.8 *For any n-WFA Γ working in *-mode, we have $L_*(\Gamma) \in \mathcal{L}(WFA)$. Also the function defined by the n-WFA Γ working in $*$-mode can be defined by a single WFA.*

Proof Let $\Gamma = (Q,\Sigma,W_\alpha,I,F)$ be a n-WFA working in *-mode where $W_\alpha = (W^1_\alpha, W^2_\alpha,\cdots,W^n_\alpha)$ and the components have states $Q_1,Q_2,\cdots,Q_n$. Consider the WFA $M = (Q',\Sigma,W'_\alpha,I',F')$ where,

$$Q' \;=\; \{[q,i] \mid q \in Q_{union},\ 1 \le i \le n\} \cup \{q'_0\}$$
$$I' \;:\; Q' \longrightarrow (-\infty,\infty) \text{ is a } \textbf{I-normal} \text{ initial distribution such that}$$
$$I'(q'_0) \;=\; 1 \text{ and } I'(q) \;=\; 0 \text{ for all other } q \in Q'$$
$$F' \;:\; Q' \longrightarrow (-\infty,\infty) \text{ is such that } F'(q'_0) \;=\; 0 \text{ and}$$
$$F'([q,i]) \;=\; F(q),\ q \in Q_{union},\ 1 \le i \le n.$$

W'_α ,the weight matrices are defined as follows,

(i) $W'_\epsilon(q'_0,[q_0,i]) = I(q_0)$ such that $q_0 \in Q_i, 1 \le i \le n$,

(ii) for each q_y such that $W^i_a(q_s,q_y) \ne 0, a \in \Sigma\cup\{\epsilon\}, 1 \le i \le n$,
$W'_a([q_s,i],[q_y,j] = W^i_a(q_s,q_y),\ 1 \le j \le n$ and$q_y \in Q_j$

The construction of the WFA clearly shows that

$$L(M) = L_*(\Gamma)$$
$$\text{and so } L_*(\Gamma) \in \mathcal{L}(WFA).$$

Also for any string $w \in \Sigma^*$ we have $f_\Gamma(w) = f_M(w)$ where f_Γ is the function defined by the n-WFA, Γ and f_M is the function defined by the WFA, M. $\square$

Theorem 3.9 *For any n-WFA Γ, $n \ge 1$ working in $= k$-mode, we have $L_{=k}(\Gamma) \in \mathcal{L}(WFA)$.*
The function defined by the n-WFA Γ working in $= k$-mode can defined by a single WFA.

Proof Let $\Gamma = (Q, \Sigma, W_\alpha, I, F)$ be a n-WFA working in $= k$-mode where $W_\alpha = (W_\alpha^1, W_\alpha^2, \cdots, W_\alpha^n)$ and the components have states $Q_1, Q_2, \cdots, Q_n$. Consider the WFA $M = (Q', \Sigma, W'_\alpha, I', F')$ where,

$$Q' = \{[q, i, j] \mid q \in Q_{union},\ 1 \leq i \leq n,\ 0 \leq j \leq k\} \cup \{q'_0\}$$
$$I' : Q' \longrightarrow (-\infty, \infty) \text{ is a } \textbf{I-normal} \text{ initial distribution such that}$$
$$I'(q'_0) = 1 \text{ and } I'(q) = 0 \text{ for all other } q \in Q'$$
$$F' : Q' \longrightarrow (-\infty, \infty) \text{ is such that } F'(q'_0) = 0 \text{ and}$$
$$F'([q, i, j]) = F(q) \text{ for all } q \in Q_{union}, 1 \leq i \leq n, 0 \leq j \leq k$$

W'_α the weight matrices are defined as follows,

(i) $W'_\epsilon(q'_0, [q_0, i', 0]) = I(q_0)$ such that $q_0 \in Q_{i'}$

(ii) for each q_y such that $W_a^i(q_s, q_y) \neq 0,\ q_s \in Q_i,\ a \in \Sigma \cup \{\epsilon\},\ 1 \leq i \leq n,\ 0 \leq j \leq k$
   (a) if $j < k$ then $W'_a([q_s, i, j-1], [q_y, i, j]) = W_a^i(q_s, q_y)$
   (b) if $j = k$ then $W'_\epsilon([q_s, i, k], [q_s, j', 0]) = 1, 1 \leq j' \leq n$ and $q_s \in Q_{j'}$.

The construction of WFA clearly shows that

$$L(M) = L_{=k}(\Gamma)$$
$$\text{and so } L_{=k}(\Gamma) \in \mathcal{L}(WFA).$$

Also for any string $w \in \Sigma^*$ we have $f_\Gamma(w) = f_M(w)$ where f_Γ is the function defined by the n-WFA, Γ and f_M is the function defined by the WFA, M.□

Theorem 3.10 *For any n-WFA Γ in $\leq k$-mode, we have $L_{\leq k}(\Gamma) \in \mathcal{L}(WFA)$. The function defined by the n-WFA Γ working in $\leq k$-mode can be defined by a single WFA.*

Proof Let $\Gamma = (Q, \Sigma, W_\alpha, I, F)$ be a n-WFA working in $\leq k$-mode where $W_\alpha = (W_\alpha^1, W_\alpha^2, \cdots, W_\alpha^n)$ and the component states are $Q_1, Q_2, \cdots, Q_n$. Consider the WFA $M = (Q', \Sigma, W'_\alpha, I', F')$ where,

$$Q' = \{[q, i, j] \mid q \in Q_{union},\ 1 \leq i \leq n, 0 \leq j \leq k\} \cup \{q'_0\}$$
$$I' : Q' \longrightarrow (-\infty, \infty) \text{ is a } \textbf{I-normal} \text{ initial distribution such that}$$
$$I'(q'_0) = 1 \text{ and } I'(q) = 0 \text{ for all other } q \in Q'$$
$$F' : Q' \longrightarrow (-\infty, \infty) \text{ such that } F'(q'_0) = 0 \text{ and}$$
$$F'([q, i, k']) = F(q) \text{ for all } q \in Q_{union},\ 1 \leq i \leq n,\ 1 \leq k' \leq k$$

W'_α the weight matrices are defined as follows,

(i) $W'_\epsilon(q'_0, [q_0, i', 0]) = I(q_0)$ such that $q_0 \in Q_{i'}$

(ii) for each q_y such that $W_a^i(q_s, q_y) \neq 0,\ q_s \in Q_i,\ a \in \Sigma \cup \{\epsilon\},\ 1 \leq i \leq n,$ $0 \leq j \leq k+1$
   (a) if $j - 1 < k$ then
   $W'_a([q_s, i, j-1], [q_y, i, j]) = W_a^i(q_s, q_y)$ where $q_y \in Q_i, 1 \leq i \leq n$

$W'_a([q_s, i, j-1], [q_y, i'', 0]) \; = \; W^i_a(q_s, q_y)$ where $q_y \in Q_{i''}, 1 \leq i, i'' \leq n, i \neq i''$

(b) if $j - 1 = k$ then $W'_\epsilon([q_s, i, j-1], [q_s, j', 0]) \; = \; 1, \; 1 \leq j' \leq n$ and $q_s \in Q_{j'}$.

The construction of WFA clearly shows that,

$$L(M) \; = \; L_{\leq k}(\Gamma)$$
$$\text{and so } \; L_{\leq k}(\Gamma) \; \in \; \mathcal{L}(WFA)$$

Moreover for any string $w \in \Sigma^*$ we have $f_\Gamma(w) \; = \; f_M(w)$ where f_Γ is the function defined by the n-WFA, Γ and f_M is the function defined by the WFA, M. □

Theorem 3.11 *For any n-WFA Γ in $\geq k$-mode, we have $L_{\geq k}(\Gamma) \in \mathcal{L}(WFA)$. The function defined by the n-WFA Γ working in $\geq k$-mode can be defined by a single WFA.*

Proof Let $\Gamma \; = \; (Q, \Sigma, W_\alpha, I, F)$ be a n-WFA in $\geq k$-mode where $W_\alpha \; = \; (W^1_\alpha, W^2_\alpha, \cdots, W^n_\alpha)$ and the component states $Q_1, Q_2, \cdots, Q_n$.
Consider the WFA $M \; = \; (Q', \Sigma, W'_\alpha, I', F')$ where, $Q' \; = \; \{[q, i, j] \mid q \in Q_{Union}, \; 1 \leq i \leq n, 0 \leq j \leq k\} \cup \{[q, i] \mid q \in Q_{union}, \; 1 \leq i \leq n\} \cup \{q'_0\}$

$$I' \; : \; Q' \; \longrightarrow \; (-\infty, \infty) \text{ is a } \textbf{I-normal} \text{ initial distribution such that}$$
$$I'(q'_0) \; = \; 1 \text{ and } I'(q) \; = \; 0 \text{ for all other } q \in Q'$$
$$F' \; : Q' \; \longrightarrow \; (-\infty, \infty) \text{ such that } F'(q'_0) \; = \; 0 \text{ and}$$
$$F'([q, i, j]) \; = \; F'([q, i]) \; = \; F(q) \text{ for all } q \in Q_{union}$$

W'_α the weight matrices are defined as follows,

(i) $W'_\epsilon(q'_0, [q_0, i', 0]) \; = \; I(q_0)$ such that $q_0 \in Q_{i'}$

(ii) for each q_y such that $W^i_a(q_s, q_y) \; \neq \; 0, \; q_s \in Q_i, \; a \in \Sigma \cup \{\epsilon\}, \; 1 \leq i \leq n, \; 0 \leq j \leq k+1$

(a) if $j - 1 < k$ then $W'_a([q_s, i, j-1], [q_y, i, j]) \; = \; W^i_a(q_s, q_y), \; q_y \in Q_i$

(b) if $j - 1 = k$ then
$W'_a([q_s, i, j-1], [q_y, i]) \; = \; W^i_a(q_s, q_y), \; q_y \in Q_i$
$W'_a([q_s, i, j-1], [q_y, j', 0]) \; = \; W^i_a(q_s, q_y), \; 1 \leq j' \leq n, \; j' \neq i,$ and $q_y \in Q_{j'}$

(c) $W'_a([q_s, i], [q_y, i]) \; = \; W^i_a(q_s, q_y), \; q_y \in Q_i$

(d) $W'_a([q_s, i], [q_y, j', 0]) \; = \; W^i_a(q_s, q_y), \; 1 \leq j' \leq n, \; j' \neq i,$ and $q_y \in Q_{j'}$

The construction of WFA clearly shows that,

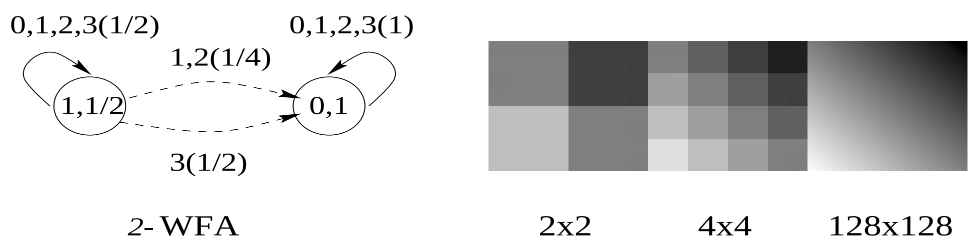
$$L(M) \; = \; L_{\geq k}(\Gamma)$$
$$\text{So } L_{\geq k}(\Gamma) \; \in \; \mathcal{L}(WFA)$$

Moreover for any string $w \in \Sigma^*$ we have $f_\Gamma(w) \; = \; f_M(w)$ where f_Γ is the function defined by the n-WFA, Γ and f_M is the function defined by the WFA, M.□

Thus we find for the n-WFA the different modes of acceptance are equivalent and the function defined by an n-WFA can be defined by a single WFA. The n-WFA accepts only those languages accepted by the WFA. In what follows, we use only $*$-mode of computations for the image representation.

3.3 Representation of Gray-Scale Images using n-WFA

We know that a WFA can be used to represent a gray-scale image. Similarly a n-WFA can be used to represent a gray-scale image.
A n-WFA M is said to represent a multi-resolution image if the function f_M represented by M is the same as the function f_I of the image.



2-WFA computing the linear grayness function

Example 3.12 Consider the 2-WFA shown in figure. The $I = (1, 0)$ and $F = (\frac{1}{2}, 1)$ and the weight matrices corresponding to the 2 components are as follows

$$W_0^1 = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & 0 \end{pmatrix}, W_1^1 = \begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & 0 \end{pmatrix}, W_2^1 = \begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & 0 \end{pmatrix}, W_3^1 = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ 0 & 0 \end{pmatrix},$$

$$W_0^2 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}, W_1^2 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}, W_2^2 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \text{ and } W_3^2 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$

In the figure the dotted lines correspond to the change in the control from one component to another. Then we can calculate the values of pixels as follows. $f(13)$ = sum of weights of all paths labeled 13.

$$f(13) = 1 \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} + 1 \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot 1 + 1 \cdot \frac{1}{4} \cdot 1 \cdot 1 = \frac{1}{8} + \frac{1}{4} + \frac{1}{4} = \frac{5}{8}$$

similarly for $f(123)$ we have $f(123) = \frac{1}{16} + \frac{1}{8} + \frac{1}{8} + \frac{1}{4} = \frac{9}{16}$. The images obtained by this 2-WFA are shown for resolutions 2×2, 4×4 and 128×128 in the above figure.

Thus we have seen how n-WFAs can be used for representing gray-scale images. Though the number of matrices in the n-WFA are more than the usual WFA the advantage of using the n-WFAs is that most of the matrices are sparse matrices and thus the matrix computations are much faster than in the usual WFA case.

3.4 Inferencing and De-Inferencing

In this subsection we give algorithms for inferencing and de-inferencing of a n-WFA.

3.4.1 Inferencing

Let $\mathcal{I}$ be a digital gray scale multi resolution image given by the average preserving function $f : \Sigma^* \longrightarrow \mathbb{R}$. We construct an average preserving m-WFA M such that $f_M = f$. During the construction

- N is the index of the last state created,
- L denotes the index of the component in which the state is in $1 \leq L \leq m$,
- i is the index of the first unprocessed state,
- $\gamma : Q_{union} \longrightarrow \Sigma^*$ is a mapping of the states to subsquares,
- ϕ_p is the image represented by the state p and
- f_w represents the subimage at the subsquare labeled w

Algorithm 1 Infer_m-WFA

Input *: Image $\mathcal{I}$ given by an average preserving function, $f : \Sigma^* \longrightarrow \mathbb{R}$ and m-the number of components of the m-WFA to be constructed*

Output *: m-WFA M representing the image $\mathcal{I}$*

Begin

(i) *Set* $N \longleftarrow 0, i \longleftarrow 0, \textit{component} \longleftarrow 1, L \longleftarrow 1, F([q_0, L]) = f(\epsilon)$ *and* $\gamma([q_0, L]) \longleftarrow \epsilon$

(ii) *Process* q_i*, i.e. for* $w = \gamma(q_i, L])$ *and each* $\alpha \in \{0, 1, 2, 3\}$ *do*
begin for
  (a) *If there are* $c_0, c_1, \cdots, c_N$ *such that*
  $f_{w\alpha} = c_0\phi_0 + c_1\phi_1 + \cdots + c_N\phi_N$*, where* $\phi_j = f_{[q_j,s]}$ *for some* $s, 1 \leq s \leq m, 0 \leq j \leq N$
  then set $W_\alpha^L([q_i, L], [q_j, s]) \longleftarrow c_j$*, for* $0 \leq j \leq N$
  (b) *else*
  if component $\leq m$ *then*
  begin elseif(then)
  $\gamma([q_{N+1}, L+1]) \longleftarrow w\alpha, F_{[q_{N+1}, L+1]} \longleftarrow f(w\alpha)$
  $W_\alpha^L([q_i, L], [q_{N+1}, L+1]) \longleftarrow 1$ *and* $N \longleftarrow N+1$
  component $\longleftarrow$ *component+1*
  end elseif(then)
  else

$\gamma([q_{N+1}, L]) \longleftarrow w\alpha, F_{[q_{N+1},L]} \longleftarrow f(w\alpha)$
$W_{\alpha}^{L}([q_i, L], [q_{N+1}, L]) \longleftarrow 1$ *and* $N \longleftarrow N + 1$
end elseif
end for

(iii) *Set* $i \longleftarrow i + 1$, *if* $i \leq N$ *then go to Step (ii)*

(iv) *Set* $I(q_0) = 1, I(q_j) = 0$ *for* $j = 1, \cdots, N$, *where* I *is the initial distribution of* M

end

The 3-WFA infered from the diminishing triangles figure using the above algorithm is given below.

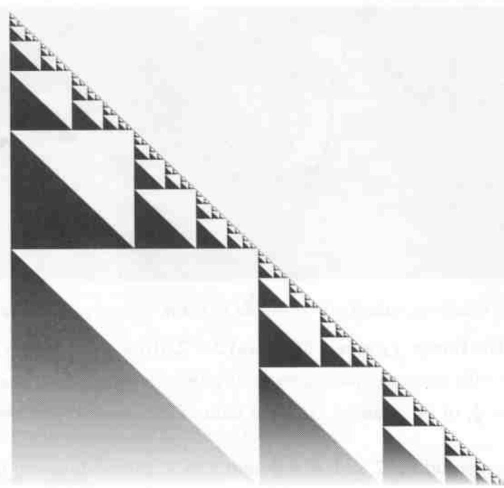
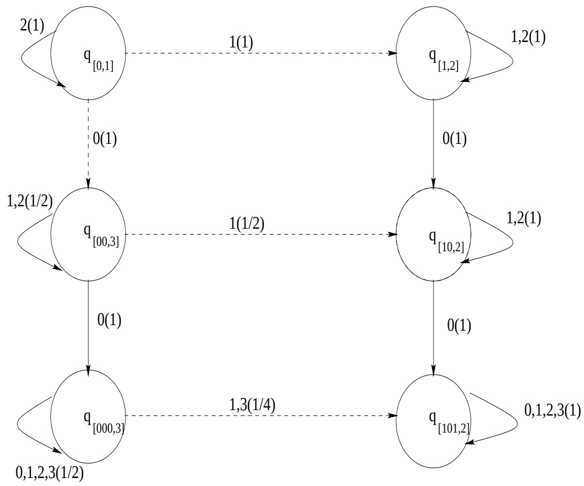


Figure 13.6: The diminishing triangles.



The 3-WFA infered from the given diminishing triangles figure

The initial distribution of the state $q_{[0,1]}$ is 1 and at all other states the initial distribution is 0. The final distribution at each state is the average intensity of the image of that state.

3.4.2 De-Inferencing

Assume, we are given a m-WFA $M(I, F, W_0^i, W_1^i, W_2^i, W_3^i), 1 \leq i \leq m$ and we want to construct a finite resolution approximation of the multi-resolution image represented by M. Let the image to be constructed be $\mathcal{I}$ of resolution $2^k \times 2^k$. Then for all $x \in \Sigma^k$, we have to compute $f(x) = I.W_x.F$. The algorithm is as follows. The algorithm computes $\phi_p(x)$ for $p \in Q$ for all $x \in \Sigma^i, 0 \leq i \leq k$. Here ϕ_p is the image of state p.

Algorithm 2 De_Infer_m-WFA
Input :*WFA* $M = (I, F, W_0^i, W_1^i, W_2^i, W_3^i), 1 \leq i \leq m$.

Output :$f(x)$, *for all* $x \in \Sigma^k$.

begin

(i) *Set* $\phi_p(\epsilon) \longleftarrow F_p$ *for all* $p \in Q_{union}$

(ii) *For* $j = 1, 2, \cdots, k$, *do the following*
begin

(iii) *For all* $p \in Q_{union}$, $x \in \Sigma^{j-1}$ *and* $\alpha \in \Sigma$ *compute*
$\phi_p(\alpha x) \longleftarrow \sum_{q \in Q_{union}} W_\alpha^i(p, q) \cdot \phi_q(x)$, *where the state* p *belongs to the component* i
end for

(iv) *For each* $x \in \Sigma^k$, *compute*

$$f(x) = \sum_{q \in Q} I_q \cdot \phi_q(x).$$

(v) *Stop*

end

The time complexity of this de-inferencing algorithm is $O(n^2 4^k)$, where n is the total number of states in the m-WFA and $4^k = 2^k \cdot 2^k$ is the number of pixels in the image. We know that $f(x)$ can be computed either by summing the weights of all the paths labeled x or by computing $I \cdot W_x \cdot F$. Finding all paths labeled of length k takes $k \cdot (4k)^n$ time. Since $n \gg k$ we prefer the matrix multiplication over this.

4 Conclusions

In this paper we have defined a new theoretical construct namely, the distributed weighted finite automata and studied the power of this construct in the various modes of acceptance. We have shown that the power of this construct is no more than the classical weighted finite automata in all modes of acceptance and hence proved that all the modes of acceptance are equivalent. We have used this construct for the representation of gray scale images and have given an inferencing and a de-inferencing algorithm for the distributed weighted finite automata. The weight matrices produced for this construct using the inference algorithm are mostly sparse matrices which occupy less space and thus the matrix computations involved in inferencing and de-inferencing are much faster when compared to the usual weight martices in the WFA.

References

[1] M. Sakthi Balan. Distributed Processing in Automata. Master's thesis, Indian Institute of Technology, Madras, April 2000.

[2] E. Csuhaj-Varju, J. Dassow, J. Keleman, and Gh. Paun. *Grammar Systems: A Grammatical Approach to Distribution and Cooperation*. Gordon and Breach, London, 1994.

[3] Vesa Halava and Tero Harju. Languages accepted by integer weighted finite automata. Technical Report TR-216, TUCS, 1998.

[4] John E. Hopcroft and Jeffrey D. Ullman. *Introduction to Automata Theory, Languages and Computation.* Addison-Wesley, 1979.

[5] K. Culik II and S. Dube. Rational and affine expressions for image description. *Discrete Applied Mathematics*, 41:85–120, 1993.

[6] K. Culik II and J. Karhumaki. Automata computing real functions. *SIAM J. on Computing*, 23:789–814, 1994.

[7] K. Culik II and J. Kari. Image compression using weighted finite automata. *Computer and Graphics*, 17:305–313, 1993.

[8] K. Culik II and J. Kari. Image-data compression using edge-optimizing algorithm for wfa inference. *Journal of Information Processing and Management*, 30:829–838, 1994.

[9] K. Culik II and J. Kari. Inference algorithms for wfa and image compression. In Yuval Fisher, editor, *Fractal Image Compression, Theory and Application*, chapter 13, pages 243–258. Springer-Verlag, 1994.

[10] K. Culik II and J. Kari. *Handbook of Formal languages*, chapter 10. Digital Images and Formal Languages. Springer-Verlag, 1997.

[11] Karel Culik II and Jarrko Kari. Finite state transformation of images. *Comput. & Graphics*, 20(1):125–135, 1996.

[12] K. Krithivasan, M. Sakthi Balan, and P. Harsha. Distributed processing in automata. *International Journal of Foundations of Computer Science*, 10(4):443–464, 1999.

[13] S. V. Ramasubramanian and Kamala Krithivasan. Finite automata and digital images. *IJPRAI*, 14(4):501–524, 2000.

[14] Derick Wood. *Theory of Computation*. Harper & Row, Publishers, Inc, 1987.