Continuous weighted finite automata

Jarkko Kari[1], Alexandr Kazda[2] and Paula Steinby[1]

[1]University of Turku, Finland [2]Charles University in Prague, Czech Republic

Weighted finite automata (WFA) over $\mathbb{R}$ are finite automata with transitions labelled by real numbers. Weighted finite automata have many nice applications in natural language processing, image manipulation and similar areas.

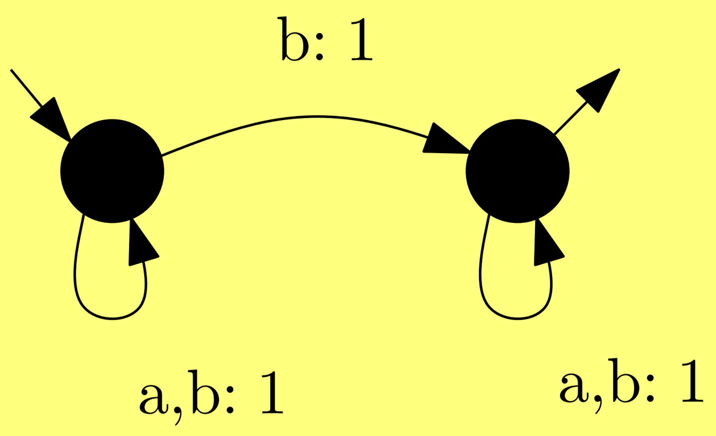
WFA provide a natural and intrinsic description for some self-similar real functions. We study continuity of WFA from the algorithmic point of view.

Weighted Finite Automata

A *weighted finite automaton* (WFA) $\mathcal{A}$ over the alphabet Σ is a set of states $1, 2, \ldots, n$ together with $n \times n$ matrices A_a, one for each $a \in \Sigma$, a row vector I and a column vector F (both of dimension n).

For v word of length m, let $F_{\mathcal{A}}(v) = I \cdot A_{v_1} \cdots A_{v_m} \cdot F$. This is the *word function* defined by $\mathcal{A}$.

In the example below, you can see the automation $\mathcal{A}$ over the alphabet $\Sigma = \{a, b\}$ such that $F_{\mathcal{A}}(v)$ is the number of b's in the word v. We can depict the automaton by a labeled graph (to the left) or by giving its matrices and vectors (to the right).



$$A_a = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad A_b = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$

$$I = \begin{pmatrix} 1 & 0 \end{pmatrix}, \quad F = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Given a WFA $\mathcal{A}$ we define its ω*-function* and (for $\Sigma = \{0, 1\}$) its *real function* as:

$$\begin{aligned} f_{\mathcal{A}}(w) &= \lim_{k \to \infty} F_{\mathcal{A}}(pref_k(w)) \\ \hat{f}_{\mathcal{A}}(x) &= f_{\mathcal{A}}(bin(x)), \end{aligned}$$

where $x \in [0, 1)$ and the function $bin : [0, 1) \to \Sigma^\omega$ returns the binary expansion that does not use 1^∞.

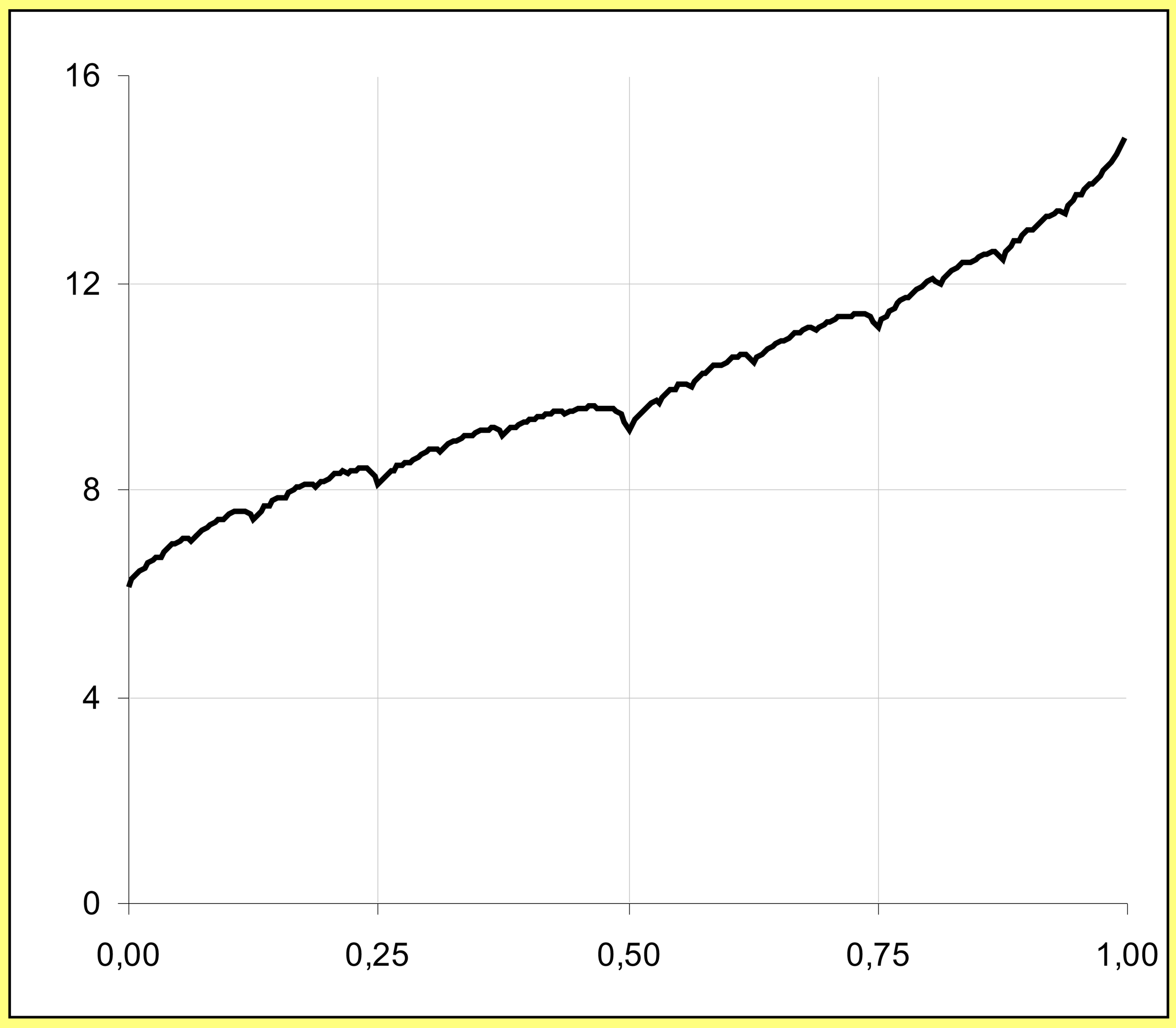
WFA: a specimen

Real functions defined by WFA can be quite complicated beasts. Consider the following automaton:

$$A_0 = \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & 3 \\ \frac{1}{3} & \frac{1}{3} & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad A_1 = \begin{pmatrix} \frac{2}{3} & 0 & 5 \\ -\frac{1}{3} & \frac{2}{3} & 6 \\ 0 & 0 & 1 \end{pmatrix}$$

$$I = (1, 0, 0), \qquad F = \begin{pmatrix} 10 \\ 6 \\ 1 \end{pmatrix}$$

This automation defines the continuous real function pictured below:



Average preserving WFA

An automaton $\mathcal{A}$ is called *average preserving* (ap), if

$$\sum_{a \in \Sigma} A_a F = |\Sigma| F.$$

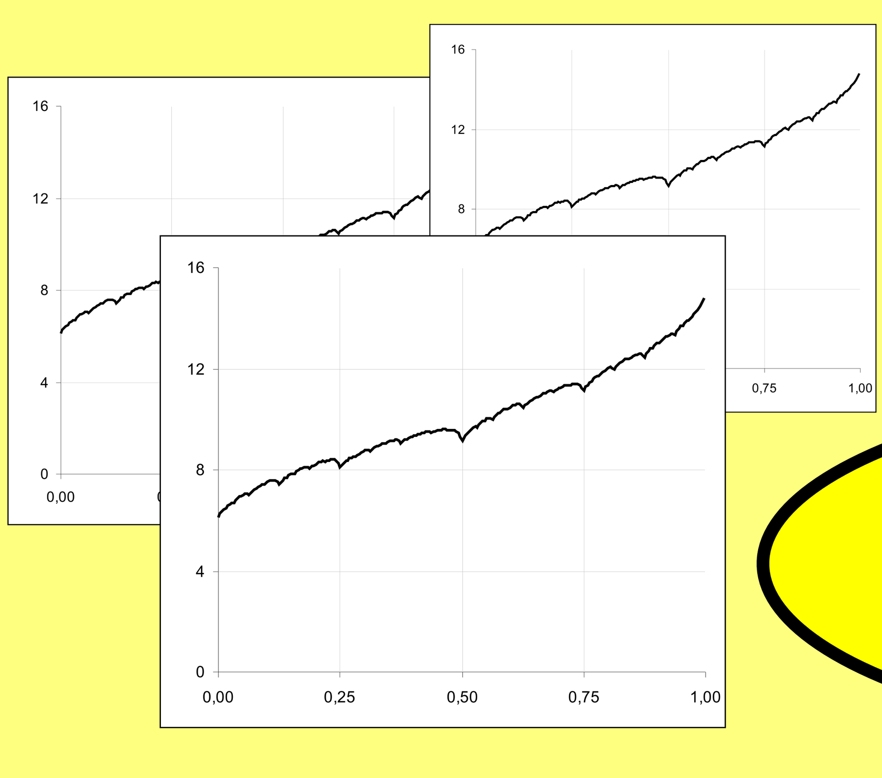
Ap automata are a quite rich class; any continuous ω-function definable by some WFA can be defined by an ap WFA.

Complexity of WFA problems

The Ap-WFA Uniform Continuity Problem

Input: An average preserving WFA $\mathcal{A}$ (with rational coefficients) over the binary alphabet $\Sigma = \{0, 1\}$.

Question: Are both $f_{\mathcal{A}}$ and $\hat{f}_{\mathcal{A}}$ everywhere continuous?



The Matrix Product Stability Problem

Input: A finite set $\{B_a | a \in \Sigma\}$ of $n \times n$ matrices with rational entries.

Question: Is it true for all $w \in \Sigma^\omega$ that

$$\lim_{k \to \infty} B_{w_1} B_{w_2} \cdots B_{w_k} = 0?$$

Complexity: Unknown, probably not recursive (although a semi-algorithm verifying stability exists).

Our result

Decision problems Matrix Product Stability and Ap-WFA uniform continuity can be algorithmically reduced to each other.