Weighted Automata

and
Monadic Second Order Logic

Nadia Labai and Johann A. Makowsky

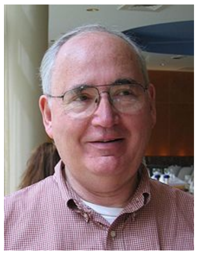
Faculty of Computer Science,
Technion - Israel Institute of Technology,
Haifa, Israel

http://www.cs.technion.ac.il/~janos
e-mail: {janos,nadia}@cs.technion.ac.il

****

Probabilistic Automata

Michael Rabin (1931 -)



Dana Scott (1932 -)

Azaria Paz (19?? -)

- Dana Scott and Michael Rabin. *Finite Automata and Their Decision Problems* IBM Journal of Research and Development. 3 (2), 1959, pp. 114–125.
- Michael O. Rabin. *Probabilistic Automata* Information and Control 6, 1963, pp. 230-245
- Jack W. Carlyle, Azaria Paz: *Realizations by Stochastic Finite Automata* J. Comput. Syst. Sci. 5(1), 1971, pp. 26-40
- Azaria Paz. *Introduction to Probabilistic Automata* cademic Press Inc., 1971.

Probabilistic automata (Rabin 1963)

A vector $\alpha = (\alpha_1, \ldots, \alpha_r) \in \mathbb{R}^r$ is stochastic if each $\alpha_i \geq 0$ and $\sum_i \alpha_i = 1$.

A matrix $\mu \in \mathbb{R}^{r \times r}$ is row-stochastic (column-stochasttic) if each row-vector (column-vector) is stochastic. μ is doubly stochastic if it is both row- and column-stochastic.

A Probabilistic Automaton (PA) A of size r is given by:

- A set $\{\mu_\sigma : \sigma \in \Sigma\}$ of $r \times r$ doubly stochastic matrices;
- Two stochastic vectors $\lambda, \gamma \in \mathcal{F}^r$.
- A defines a function $f_A : \Sigma^\star \to \mathbb{R}$

$$f_A(w) = f_A(\sigma_1 \circ \sigma_2 \circ \ldots \circ \sigma_n) = \lambda \mu_{\sigma_1} \mu_{\sigma_2} \cdot \ldots \cdot \mu_{\sigma_n} \gamma^t$$

- A function $f : \Sigma^\star \to \mathbb{R}$ is PA-recognizable if $f = f_A$ for some PA A.

Intuition behind probabilistic automata

- The automaton has r states.
- λ gives the probability λ_i that the automaton is in state i when reading the empty word.
- μ_σ is the transition matrix for the transition when reading σ..
- γ gives the probability γ_i that state i is an accepting state.

Multiplicity automata (Schutzenberger, 1961)

Marcel-Paul Schützenberger (1920–1996)

- M.P.Schützenberger. *On the definition of a family of automata*
  Information and Control, Volume 4 (2-3), 1961, pp. 245-270
- M.P.Schützenberger. *Finite counting automata*
  Information and Control, Volume 5 (2), 1962, pp. 91–107

Multiplicity automata (Schutzenberger, 1961)

A Multiplicity Automaton (MA) A of size r over a field $\mathcal{F}$ is given by:

- A set $\{\mu_\sigma : \sigma \in \Sigma\}$ of $r \times r$ matrices over $\mathcal{F}$;
- Two vectors $\lambda, \gamma \in \mathcal{F}^r$.
- A defines a function $f_A : \Sigma^\star \to \mathcal{F}$

$$f_A(w) = f_A(\sigma_1 \circ \sigma_2 \circ \ldots \circ \sigma_n) = \lambda \mu_{\sigma_1} \mu_{\sigma_2} \cdot \ldots \cdot \mu_{\sigma_n} \gamma^t$$

- A function $f : \Sigma^\star \to \mathcal{F}$ is MA-recognizable if $f = f_A$ for some MA A.

Probabilistic automata (PA) and Multiplicity automata (MA) where introduced independently, generalizing the developments described in the famous paper by M. Rabin and D. Scott (1959).

Word functions and power series

Let $\mathcal{F}$ be a field (or semi-ring) and Σ an alphabet.

We can view Σ as a set of non-commutative indeterminates and $\Sigma^\star$ is its set of monomials.

A function $f : \Sigma^\star \to \mathcal{F}$ the defines a power series

$$S_f(w) = \sum_{w \in \Sigma^\star} f(w)w$$

A power series is rational if it can be obtained from polynomials by addition, multiplication, external products and the star-operation.

Regular languages and power series

We define a language $L(f) = \{w \in \Sigma^\star : f(w) \neq 0\}$.

$L(f)$ is FA-recognizable if there is a determinsitic finite automaton A which accepts $L(f)$.

Theorem: (Kleene-Schützenberger)

In the case of $\mathcal{F} = \mathbb{Z}_2$ the following are equivalent:

(i) $L(f)$ is FA-recognizable;

(ii) $L(f)$ is regular;

(iii) $S_f(w)$ is rational.

MA-Recognizable word functions

A function $f : \Sigma^\star \to \mathcal{F}$ is MA-recognizable if there exists an MA A such that $f_A = f$.

Theorem: (Schützenberger 1961)

For arbitrary semi-rings $\mathcal{F}$ the following are equivalent:

(i) f MA-recognizable

(ii) $S_f(w)$ is rational

Is there an analogue for **regular expressions** for MA over $\mathcal{F}$?

Hankel matrices in classical mathematics

(over a field $\mathcal{F}$)

Let $f : \mathcal{F} \to \mathcal{F}$ be a function.

A finite or infinite matrix $H(f) = h_{i,j}$ over a field $\mathcal{F}$ is a Hankel matrix for f if $h_{i,j} = f(i + j)$.

Hankel matrices have many applications in:
numeric analysis, probability theory and combinatorics.

- Padé approximations
- Orthogonal polynomials
- Probability theory (theory of moments)
- Coding theory (BCH codes, Berlekamp-Massey algorithm)
- Combinatorial enumerations
  (Lattice paths, Young tableaux, matching theory)

Hankel matrices over words

Let Σ be a finite alphabet and $\mathcal{F}$ be a field and let $f : \Sigma^\star \to \mathcal{F}$ be a function on words.

A finite or infinite matrix $H(f) = h_{u,v}$ indexed over the words $u, v \in \Sigma^\star$ is a Hankel matrix for f if $h_{u,v} = f(u \circ v)$. Here $\circ$ denotes concatenation.

Hankel matrices over words have applications in

- Formal language theory and stochastic automata,
  J. Carlyle and A. Paz 1971
- Learning theory (exact learning of queries).
  A.Beimel, F. Bergadano, N. Bshouty, E. Kushilevitz, S. Varricchio 1998
  J. Oncina 2008
- Definability of picture languages.
  O. Matz 1998, and D. Giammarresi and A. Restivo 2008

Hankel matrices for graphs

If we want to define Hankel matrices for (labeled) graphs,

what plays the role of concatenation?

- Disjoint union
  Used by Freedman, Lovász and Schrijver, 2007, for characterizing multiplicative graph parameters over the real numbers

- k-unions (connections, connection matrices)
  Used by Freedman, Lovász, Schrijver and Szegedy, 2007ff, for characterizing various forms and partition functions.

- Joins, cartesian products, generalized sum-like operations
  used by Godlin, Kotek and JAM to prove non-definability.

Multiplicity Automata and Hankel matrices (over a field)

THEOREM: (J. Carlyle and A. Paz 1971)

For a function $f : \Sigma^\star \to \mathcal{F}$ the following are equivalent:

(i) f is MA-recognizable;

(ii) S_f is rational

(iii) the Hankel matrix $H(f)$ has finite rank over $\mathcal{F}$.

This is an **ALGEBRAIC** **characterization of MA-recognizability**.

The Büchi-Elgot-Trakhtenbrot Theorem (around 1960)

A word w of size n over an alphabet Σ can be considered as a structure

$$\mathfrak{A}_w = \langle [n], <_{nat}, P_\sigma, (\sigma \in \Sigma) \rangle$$

where $P_\sigma : \sigma \in \Sigma$ is a partition of $[n]$ into possibly empty sets.

THEOREM: (R. Büchi, C. Elgot and B. Trakhtenbrot)

The following are equivalent:

(i) L is FA-recognizable;

(ii) L is regular;

(iii) The class $\{\mathfrak{A}_w : w \in L\}$ of structures is **definable in Monadic Second Order Logic**.

Is there an analogue for MA-recognizability ?

MSOLEVAL$_{\mathcal{F}}$

MSOLEVAL$_{\mathcal{F}}$ consists of those functions mapping relational structures into $\mathcal{F}$ which are definable in Monadic Second Order Logic MSOL.

The functions in MSOLEVAL$_{\mathcal{F}}$ are represented as terms associating with each τ-structure $\mathcal{A}$ a polynomial $p(\mathcal{A}, \bar{X}) \in \mathcal{F}[\bar{X}]$.

Similarily, CMSOLEVAL$_{\mathcal{F}}$ is obtained by replacing MSOL by Monadic Second Order Logic with modular counting CMSOL.

MSOLEVAL$_{\mathcal{F}}$ is defined inductively:

(i) monomials are products of constants in $\mathcal{F}$ and indeterminates in $\bar{X}$ and the product ranges over elements a of $\mathcal{A}$ which satisfy an MSOL-formula $\phi(a)$.

(ii) polynomials are then defined as sums of monomials where the sum ranges over unary relations $U \subset A$ satisfying an MSOL-formula $\psi(U)$.

MSOLEVAL$_{\mathcal{F}}$ was first studied in a sequence of papers on graph polynomials by J.A.M. variably co-authored with B. Courcelle, B. Godlin, T. Kotek, U. Rotics, B. Zilber.

We procced now by examples of word functions in MSOLEVAL.

Examples of word functions in MSOLEVAL, I

Let $\Sigma = \{0, 1\}$ and $w \in \Sigma^*$ be represented by the structure

$$\mathcal{A}_w = \langle [\ell(w)], <, P_0, P_1 \rangle.$$

Counting occurrences:

(i) The function $\sharp_1(w)$ counts the number of occurences of 1 in a word w can be written as

$$\sharp_1(w) = \sum_{i \in [n] : P_1(i)} 1.$$

(ii) The polynomial $X^{\sharp_1(w)}$ can be written as

$$X^{\sharp_1(w)} = \prod_{i \in [n] : P_1(i)} X.$$

Examples of word functions in MSOLEVAL, II

Let L be a regular language defined by the MSOL-formula ϕ_L.

The polynomial

$$\sharp_L(w) = \sum_{u \in L : \exists v_1, v_2 (w = v_1 \circ u \circ v_2)} X^{\ell(u)}$$

is the generating function of the number of (contiguous) occurences of words $u \in L$ in a word w of size i.

It can also be written as

$$\sharp_L(w) = \sum_{U \subseteq [n] : \psi_L(U)} \prod_{i \in U} X,$$

where $\psi_L(U)$ says that U is an interval and ϕ_L^U, the relativization of ϕ_L to U holds.

Examples of word functions in MSOLEVAL, III

Let $\text{int}(w) = \sum_{i=0}^{\ell(w)-1} 2^{-i} w[i]$.

$\text{int}(w)$ considers w as a rational number in $[0,1]$ written in binary and computes its value.

$\text{int}(w)$ can be written as

$$\text{int}(w) = \sum_{U \subset [\ell(w)] : \text{INIT}_1(U)} \prod_{i \in U} (2^{-1})$$

where $\text{INIT}_1(U)$ says that U is an initial segment of $\langle \ell(w), < \rangle$ where the last element is in P_1.

It should be clear that it is very *convenient* and *user friendly* to define word functions as terms in $\text{MSOLEVAL}_{\mathcal{F}}$.

Examples of word functions NOTin MSOLEVAL

(i) The function sqexp$(w) = 2^{\ell(w)^2} = \prod_{(x,y):x=x \wedge y=y} 2$ is not in MSOLEVAL because the product is over tuples, rather than elements.

(ii) The function dexp$(w) = 2^{2^{\ell(w)}}$ is not representable in MSOLEVAL$_{\mathcal{F}}$ due to a growth argument.

Characterizing functions defined by Multiplicity Automata

Main Theorem: (N. Labai and J.A.M., 2012)

Let $\mathcal{F}$ be a field, and $f : \Sigma^* \to \mathcal{F}$.

The following are equivalent:

(i) $f = f_A$ for some Multiplicity Automaton A over $\mathcal{F}$.

(ii) $f \in \mathrm{MSOLEVAL}_{\mathcal{F}}$

(iii) $f \in \mathrm{CMSOLEVAL}_{\mathcal{F}}$

(iv) $M(\circ, f)$ has finite rank.

Proof: (i) $\leftrightarrow$ (iv) is the Carlyle-Paz Theorem. (ii) $\leftrightarrow$ (iii) follows from CMSOL equals MSOL on words. (iii) $\to$ (iv) is the Finite Rank Theorem. (i) $\to$ (ii) is proven using matrix algebra and logic.

My co-workers (and former students)

Benny Godlin

Nadia Labai

Tomer Kotek

Previous attempts of characterizing MA-recognizable functions

Our Main Theorem is an analogue to the
Büchi-Elgot-Trakhtenbrot Theorem
for multiplicity automata.

There were previous attempts to prove such a theorem using a subset RMSOL of weighted MSOL-formulas rather than MSOL-definable functions.

Manfred Droste

Paul Gastin

- M. Droste and P Gastin,
  Weighted automata and weighted logic,
  TCS 380 (2007), pp. 69-86.
- M. Droste, W. Kuich and H. Vogler, eds.,
  Handbook of Weighted Automata,
  Springer 2009

Weighted RMSOL vs. MSOLEVAL

However, there are serious disadvantages in their approach.

(i) The definition of RMSOL is not a purely syntactic.

(ii) The formulas are hybrid objects, mixing constants from $\mathcal{F}$ and logical expressions. For instance $\forall x \cdot 2$ is a weighted formula (for $2 = 1 + 1$ in a field) which represents the function $2^{\ell(w)}$, and $\forall x \forall y \cdot 2$ is a weighted formula which represents the function $2^{2^{\ell(w)}}$.

(iii) Seemingly equivalent formulas can represent different functions: $\exists x P_1(x)$ represents the function $\sharp_1(w)$ but $\exists (P(x) \vee P(x))$ represents the function $2 \cdot \sharp_1(w)$.

(iv) Some of these disadvantages have been corrected in very recent papers by M. Droste and P. Gastin in the Handbook and B. Bollig, P. Gastin , B. Monmege and M. Zeitoun presented at ICALP 2010.

In contrast to these disadvantages, $\mathrm{MSOLEVAL}_{\mathcal{F}}$ has the following advantages:

(i) The expressions are natural and intuitive.

(ii) The expressions are defined for all formulas of MSOL without any restrictions.

(iii) If we replace formulas occurring in an expression by equivalent formulas, the word function it represents remains the same.

What else is in the paper?

- If the field $\mathcal{F}$ is replace by a semiring $\mathcal{S}$ a similar result holds.

  Instead of the finite rank condition of the Hankel matrix we have to require that the word function is in a finitely generated, stable semimodule.

- We also give a direct translation between RMSOL and MSOLEVAL which uses the syntactic restriction imposed in RMSOL.

More Details and Proofs

If time permits we now discuss the following:

- We discuss the Bilinear Decomposition Theorem for word functions in MSOLEVAL.
- We show how to derive the finite rank of the Hankel matrix using the Bilinear Decomposition Theorem.
- We show how to convert a word function f recognizable by a weighted automaton into an equivalent expression in MSOLEVAL representing f.

The Bilinear Decomposition Theorem

Let $f \in \mathrm{MSOLEVAL}_{\mathcal{F}}$ be a word function $\Sigma^* \to \mathcal{F}$.

We would like to compute $f(u \circ v)$ from $f(u)$ and $f(v)$ only.

Let us discuss two examples with $\Sigma = \{0, 1\}$ and $u, v, w \in \Sigma^*$.

- $\sharp_1(w)$ counts the number of 1's in w and is in $\mathrm{MSOLEVAL}_{\mathcal{F}}$.
- $b_1(w)$ counts the number of blocks of 1's in w. A *block of* 1*'s in* w is a maximal set of consecutive positions $i \in [\ell(w)]$ in the word w with $P_1(i)$.

Clearly we have

$$\sharp_1(u \circ v) = \sharp_1(u) + \sharp_1(v). \tag{1}$$

Computing $b_1(u \circ v)$, I

However

$$b_1(u \circ v) = \begin{cases} b_1(u) + b_1(v) - 1 & P_1(u[\ell(u)]) \text{ and } P_1(v[1]) \\ b_1(u) + b_1(v) & \text{else} \end{cases} \tag{2}$$

To handle the second case we introduce auxiliary functions from $\text{MSOLEVAL}_{\mathcal{F}}$.

(i) $f_1(w)$ counts the number of blocks of 1's in w which include the first position.

(ii) $\ell_1(w)$ counts the number of blocks of 1's in w which include the last position.

(iii) $i_1(w)$ counts the number of blocks of 1's in w which exclude the first and last position.

(iv) $c(w) = 1$, the constant function with value 1.

It is easily verified that they are really in $\text{MSOLEVAL}_{\mathcal{F}}$.

Computing $b_1(u \circ v)$, II

Clearly, we have

$$b_1(w) = f_1(w) + \ell_1(w) + i_1(w) - f_1(w)\ell_1(w) \tag{3}$$

Furthermore, we have $f_1(w), \ell_1(w) \in \{0, 1\}$ and

$$f_1(u \circ v) = f_1(u) \tag{4}$$
$$\ell_1(u \circ v) = \ell_1(v) \tag{5}$$
$$i_1(u \circ v) = i_1(u) + i_1(v) + \ell_1(u) + f_1(v) - \ell_1(u)f_1(v) \tag{6}$$
$$c(u \circ v) = 1 \tag{7}$$

Let $B(w) = (f_1(w), \ell_1(w), i_1(w), c(w))$.

Computing $b_1(u \circ v)$, III

Proposition

There are matrices $M^f, M^l, M^i, M^c \in \mathcal{F}^{4\times 4}$ such that

$$f_1(u \circ v) = B(u) \cdot M^f B(v)^{tr} \tag{8}$$

$$\ell_1(u \circ v) = B(u) \cdot M^l B(v)^{tr} \tag{9}$$

$$i_1(u \circ v) = B(u) \cdot M^i B(v)^{tr} \tag{10}$$

$$c(u \circ v) = B(u) \cdot M^c B(v)^{tr} \tag{11}$$

Computing $b_1(u \circ v)$, IV

Using the equations (3) - (7) one easily verifies that

$$M^f = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad M^l = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{12}$$

$$M^i = \begin{pmatrix} 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}, \quad M^c = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{13}$$

Conclusion

(i) The Hankel matrix $H(\sharp_1)$ of $\sharp_1$ has rank 2 due to equation (1)

(ii) The Hankel matrices $H(b_1), H(f_1), H(\ell_1)$ and $H(i_1)$ of b_1, f_1, ℓ_1, i_1 have rank at most 4.

Bilinear Decomposition Theorem (BDT) for Word Functions

The two examples are typical in the following sense:

Theorem:

Let $f \in \mathrm{MSOLEVAL}_{\mathcal{F}}$ be a word function $\Sigma^* \to \mathcal{F}$ of quantifier rank $q(f)$. There is a finite sequence $F = (g_1, \ldots, g_{\alpha(f)})$ of functions in $\mathrm{MSOLEVAL}_{\mathcal{F}}$ of size $\alpha(f)$ and for each g_i there is a matrix $M^{(i)} \in \mathcal{F}^{q(f) \times q(f)}$ such that

(i) $f \in F$ and

(ii) $g_i(u \circ v) = F(u) \cdot M^{(i)} F(v)^{tr}$.

$\alpha(f)$ actually only depends on $q(f)$ but grows very quickly.

The full proof is in B. Courcelle, J.A.M. and U. Rotics, DAM 2001.

The bilinear version was only formulated later in J.A.M., Annals of Pure and Applied Logic, 2004, but uses the same proof.

Here we merely note that F can be chosen to consist of all the functions in $\mathrm{MSOLEVAL}_{\mathbf{F}}$ of quantifier rank at most $q(f)$.

This is a rough estimate. The examples above show that F can often be much smaller.

Finite Rank Theorem for Word Functions

Using the BDT we get

Theorem (B. Godlin, T. Kotek and J.A.M., 2008)

Let $f \in \mathrm{MSOLEVAL}_{\mathcal{F}}$ be a word function $f : \Sigma^* \to \mathcal{F}$ with all the formulas of quantifier rank at most $q(f)$.

Then the Hankel matrix $H(f)$ has rank at most $\alpha(f)$.

Proof (of the Main Theorem), I

Let A be a weighted automaton of size r over $\mathcal{F}$ for words in Σ^* given by

(i) Two vectors $\alpha, \gamma \in \mathcal{F}^r$, and

(ii) for each $\sigma \in \Sigma$ a matrix $\mu_\sigma \in \mathcal{F}^{r \times r}$.

For a word $w = \sigma_1 \sigma_2 \ldots \sigma_{\ell(w)}$ the automaton A defines the function

$$f_A(w) = \alpha^T \mu_{\sigma_1} \cdot \ldots \cdot \mu_{\sigma_{\ell(w)}} \cdot \gamma. \tag{14}$$

We have to show that $f_A \in \mathrm{MSOLEVAL}_{\mathcal{F}}$.

To unify notation we define

$$M(i, j, \sigma) = (\mu_\sigma)_{i,j}. \tag{15}$$

Furthermore, the word w is given as a function $w : [\ell(w)] \to \Sigma$.

Proof, II

Using Equation 14 and matrix algebra we get

$$f_A(w) =$$

$$\sum_{\pi:[n+2]\to[r]} \alpha_{\pi(1)} \cdot [M(\pi(1), \pi(2), w(1)) \cdot \ldots \cdot M(\pi(n), \pi(n+1), w(n))] \cdot \gamma_{\pi(n+2)} =$$

$$\sum_{\pi:[n+2]\to[r]} \alpha_{\pi(1)} \cdot \left(\prod_{v\in[n]} M(\pi(v), \pi(v+1), w(v)) \right) \cdot \gamma_{\pi(n+2)} \tag{16}$$

Proof, III

To convert Equation (16) into an expression in MSOLEVAL(Σ) we use the following lemmas:

Let S be any set and $\pi : S \to [r]$ be a function. π induces a partition of S into sets $U_1^\pi, \ldots, U_r^\pi$ by $U_i^\pi = \{s \in S; \pi(s) = i\}$. Conversely, every partition $\mathcal{U} = (U_1, \ldots, U_r)$ of S induces a function $\pi_\mathcal{U}$ by setting $\pi_\mathcal{U}(s) = i$ for $s \in U_i$.

Lemma 1
Let $M(\pi)$ be any function depending on
pi.

$$\sum_{\pi:S\to[r]} M(\pi) = \sum_{\mathcal{U}} M(\pi_\mathcal{U}) = \sum_{U_1,\ldots U_r:Partition(U_1,\ldots,U_r)} M(\pi_\mathcal{U}) \qquad (17)$$

where $\mathcal{U}$ ranges over all partition of S into r sets $U_i : i \in [r]$.

Clearly, $Partition(U_1, \ldots, U_r)$ can be written in MSOL.

Proof, IV

To convert the factors $\alpha_{\pi(1)}$ and $\gamma_{\pi(n+2)}$ we proceed as follows:

Lemma 2
Let α_i be the unique value of the coordinate of α such that $1 \in U_i$. Similarily, let γ_i be the unique value of the coordinate of γ such that $n+2 \in U_i$.

$$\alpha_{\pi(1)} = \prod_{i=1}^{r} \prod_{1 \in U_i} \alpha_i \tag{18}$$

$$\gamma_{\pi(n+2)} = \prod_{i=1}^{r} \prod_{n+2 \in U_i} \gamma_i \tag{19}$$

Proof:
First we note that, as $\mathcal{U}$ is the partition induced by π, the restriction of π to U_i is constant for all $i \in [r]$. Next we note that the product ranging over the empty set gives the value 1. Q.E.D.

Proof, V

Similarily, to convert the factor $\prod_{v\in[n]} M(\pi(v), \pi(v+1), w(v))$ use following lemma:

Lemma 3
Let $M_{i,j,w(v)}$ be the unique value of the (i,j)-entry of the matrix $\mu_{w(v)}$ such that $v \in U_i$ and $v+1 \in U_j$.

$$\prod_{v\in[n]} M(\pi(v), \pi(v+1), w(v)) = \prod_{i,j=1}^{r} \prod_{v\in U_i, v+1\in U_j} M_{i,j,w(v)} \tag{20}$$

By writing $U_i(v)$ instead of $v \in U_i$ it is not difficult to see that the monomials of the Lemmas 1, 2 and 3 are indeed in MSOLEVAL$_{\mathcal{F}}$.

Using that MSOLEVAL$_{\mathcal{F}}$ is closed under products
and using Lemmas 1, 2 and 3 we complete the proof $\square$

Thank you for your attention !