Linear Recurrence Relations for Graph Polynomials

Eldar Fischer* and Johann A. Makowsky**

Department of Computer Science,
Technion–Israel Institute of Technology, Haifa, Israel
{eldar,janos}@cs.technion.ac.il

For Boaz (Boris) Abramovich Trakhtenbrot
on the occasion of his 85th birthday.

Abstract. A sequence of graphs G_n is *iteratively constructible* if it can be built from an initial labeled graph by means of a repeated fixed succession of elementary operations involving addition of vertices and edges, deletion of edges, and relabelings. Let G_n be a iteratively constructible sequence of graphs. In a recent paper, [27], M. Noy and A. Ribò have proven linear recurrences with polynomial coefficients for the Tutte polynomials $T(G_i, x, y) = T(G_i)$, i.e.

$$T(G_{n+r}) = p_1(x, y)T(G_{n+r-1}) + \ldots + p_r(x, y)T(G_n).$$

We show that such linear recurrences hold much more generally for a wide class of graph polynomials (also of labeled or signed graphs), namely they hold for all the extended MSOL-definable graph polynomials. These include most graph and knot polynomials studied in the literature.

1 Introduction

Among Boaz' celebrated papers we find two papers dealing with Monadic Predicate Calculus and finite automata [30,31,21], and therein the theorem known today as the Büchi-Elgot-Trakhtenbrot Theorem. It states that the regular languages are exactly those sets of words which are definable in Monadic Predicate Calculus, also known as Monadic Second Order Logic MSOL.

There are innumerous papers dealing the importance of Monadic Predicate Calculus for algorithmic questions. One of the crucial properties of MSOL is the fact that the MSOL-theories of two structures determine uniquely the MSOL-theory of the disjoint union of the two structures, and also of many other sum-like

* Partially supported by a Grant of the Fund for Promotion of Research of the Technion–Israel Institute of Technology.

** Partially supported by a Grant of the Fund for Promotion of Research of the Technion–Israel Institute of Technology and an ISF Grant 2007–2009.

A. Avron et al. (Eds.): Trakhtenbrot/Festschrift, LNCS 4800, pp. 266–279, 2008.

compositions of the two structures. This is generally known as the Feferman-Vaught theorem for Monadic Second Order Logic MSOL.[1] In [23] many algorithmic applications of this property of MSOL are discussed. We present here yet another application of Monadic Predicate Calculus to algorithmic questions, namely to graph polynomials. A sequence of graphs G_n is *iteratively constructible*[2] if it can be built from an initial graph by means of a repeated fixed succession of elementary operations involving addition of vertices and edges, and deletion of edges. Let G_n be a iteratively constructible sequence of graphs. In a recent paper, [27], M. Noy and A. Ribò have proven linear recurrences with polynomial coefficients for the Tutte polynomials $T(G_i, x, y)$, i.e. recurrences of the form

$$T(G_{n+r}, x, y) = p_1(x, y) \cdot T(G_{n+r-1}, x, y) + \ldots + p_r(x, y) \cdot T(G_n, x, y).$$

Particular cases were studied previously in [4]. We show in Theorem 1 that such linear recurrences hold much more generally for a wide class of graph polynomials (also of labeled or signed graphs), namely the MSOL-definable graph polynomials introduced in [10] and further studied[3] in [25,23,26,24]. These include the classical chromatic polynomial and the Tutte polynomial, the matching polynomials, the interlace polynomials, the cover polynomial, certain Farrell polynomials, and the various colored Tutte polynomials studied by Bollobás and Riordan, [6]. Because of the latter, our result can also be applied to the computation of the Jones polynomials and Kauffman brackets for iteratively constructible knots and links. Actually, a close inspection of the literature reveals that virtually all graph polynomials studied in the literature fall into this class, [24]. Only the interlace polynomials seem to be an exception. For those one has to use an extended logic CMSOL obtained from MSOL by adding modular counting quantifiers, cf. [1,2,8].

Our proof is based on a further refinement of Makowsky's Splitting Theorem for MSOL-definable graph polynomials from [23]. All graphs and logical structures in this paper are *finite*.

2 Guiding Examples

We consider six iteratively constructed graph sequences of undirected simple graphs. These sequences are constructed from an initial graph by the repeated application of a deterministic graph operation.

2.1 Six Graph Sequences and Their Iterative Constructions

For a graph $G = (V, E)$ we denote by $\bar{G}$ the complement graph $\bar{G} = (V, V^2 - E - Diag(V))$, with $Diag(V) = \{(v, v) : v \in V\}$ We look at the following six graph sequences:

[1] Strictly speaking the extension of the Feferman-Vaught theorem for First Order Logic FOL to MSOL emerged explicitly only in later papers of H. Läuchli, Y. Gurevich and S. Shelah [22,29,19].

[2] In [27] they are called *recursively constructible*.

[3] The papers [23,26,24] contain extensive bibliographies on graph polynomials.

(i) The sequence E_n of empty graphs with vertex set $V(E_n) = \{0, \ldots, n-1\}$ and edge set $E(E_n) = \emptyset$. $E_1 = P_1$ is an isolated vertex $\{0\}$. E_{n+1} is obtained from E_n by the disjoint union of $E_n \sqcup E_1$.

(ii) The sequence K_n of cliques on n vertices. $K_1 = E_1$ and $K_{n+1} = \overline{E_{n+1}}$. Iteratively we have $K_{n+1} = K_n \bowtie K_1$ where $\bowtie$ denotes the join operation.

(iii) The sequence P_n of paths on n vertices, i.e. the graphs with vertex set $V(P_n) = \{0, \ldots, n-1\}$ and edge set $E(P_n) = \{(i, i+1) : 0 \leq i \leq n-2\}$. $E_1 = P_1$ is an isolated vertex. P_{n+1} is obtained from P_n by setting $V(P_{n+1}) = V(P_n) \sqcup \{n\}$ and $E(P_{n+1}) = E(P_n) \cup \{(n-1, n)\}$. To specify this sequence with an iterated graph operation we look at $\bar{P}_n$ obtained from P_n by distinguishing a vertex of degree one. Then we have $\bar{P}_{n+1} = \eta(\bar{P}_n \sqcup \bar{P}_1)$ where η put an edge between the two distinguished elements, and leaves only the vertex coming from $\bar{P}_1$ as the distinguished element, and P_{n+1} is obtained from $\bar{P}_{n+1}$ by ignoring the distinguished element.

(iv) The sequence C_n of circles on n vertices, i.e. the graphs with vertex set $V(C_n) = \{0, \ldots, n-1\}$ and edge set $E(C_n) = \{(i, i+1) : 0 \leq i \leq n-2\} \cup \{(n-1, 0)\}$. C_1 is a single vertex with a loop. C_{n+1} is obtained from C_n by setting $V(C_{n+1}) = V(C_n) \sqcup \{n\}$ and $E(C_{n+1} = E(P_{n+1}) \cup \{(0, n)\}$. To specify this sequence with an iterated graph operation we can use a vertex replacement operation where a distinguished vertex of C_n is replace by $\bar{P}_2$.

(v) The sequence L_n of ladders on $2n$ vertices, i.e. the graphs with vertex set $V(L_n) = \{0, \ldots, 2n-1\}$ and edge set

$$\begin{aligned} E(L_n) = \{(2i, 2i+2) : 0 \leq i \leq n-2\} \cup \\ \{(2i+1, 2i+3) : 0 \leq i \leq n-2\} \cup \\ \{(2i, 2i+1) : 0 \leq i \leq n-1\} \end{aligned}$$

The reader can easily describe how L_{n+1} is obtained from L_n by a suitable choice of distinguished elements and appropriate vertex replacements. A formal definition is presented in Sect. 3, cf. Proposition 2.

(vi) Similarly, the sequence W_n of wheels on $n+1$ vertices can be obtained. Here W_n is the graph with vertex set $V(W_n) = \{0, \ldots, n\}$ and edge set

$$\begin{aligned} E(W_n) = \{(i, i+1) : 0 \leq i \leq n-2\} \cup \{(n-1, 0)\} \\ \cup \{(i, n) : 0 \leq i \leq n-1\} \end{aligned}$$

For $G = (V, E)$ let $I(G)$ be the graph with $V(I(G)) = V \sqcup E$ and $E(I(G)) = \{(v, e) \in V \times E :$ there is an u with $(v, u) = e\}$. The sequence $I(G_n)$ is often much more complicated to describe iteratively than the sequence G_n. In particular we shall see in the next section, Corollary 1, that the sequence $I(K_n)$ is not iteratively constructible in the sense we have in mind. A general definition of *iteratively constructed* and *iteratively constructible* classes is given in Sect. 3.4.

2.2 The Matching Polynomial

For a graph G, the *matching polynomial* $\mu(G, x) \in \mathbb{Z}[x]$ is defined by

$$\mu(G, x) = \sum_k m_k(G) \cdot x^k$$

where $m_k(G)$ is the number of k-matchings of G.

To compute $\mu(P_n, x)$ we use auxiliary polynomials

$$\mu^+(P_n, x) = \sum_k m_k^+(P_n) \cdot x^k$$

and

$$\mu^-(P_n, x) = \sum_k m_k^-(P_n) \cdot x^k$$

where $m_k^+(P_n)$ and $m_k^-(P_n)$ is the number of k-matchings of P_n which includes, respectively excludes the last vertex.

Clearly we have

$$m_k(P_n) = m_k^+(P_n) + m_k^-(P_n)$$

hence

$$\mu(P_n, x) = \mu^+(P_n, x) + \mu^-(P_n, x).$$

It is easy to see that

$$\mu^-(P_{n+1}) = \mu^-(P_n) + \mu^+(P_n)$$
$$\mu^+(P_{n+1}) = x \cdot \mu^-(P_n)$$

Let[4] $\bar{\mu}_n = (\mu^-(P_n), \mu^+(P_n))^t$. We get

$$A\bar{\mu}_n = \bar{\mu}_{n+1}$$

with

$$a_{1,1} = 1, a_{1,2} = 1, a_{2,1} = x, a_{2,2} = 0$$

The characteristic polynomial of A is

$$det(\lambda \mathbf{1} - A) = \lambda^2 - \lambda - x$$

so we get the linear recurrence relation (independent of n)

$$\mu(P_{n+2}) = \mu(P_{n+1}) + x \cdot \mu(P_n)$$

2.3 The Vertex-Cover Polynomial

For a graph G, the *vertex-cover polynomial* $vc(G, x) \in \mathbb{Z}[x]$ is defined by

$$vc(G, x) = \sum_k vc_k(G) \cdot x^k$$

where $vc_k(G)$ is the number of k-vertex-covers of G. In [12] the following recurrence relations are derived:

(i) $vc(P_{n+1}, x) = x \cdot vc(P_n, x) + x \cdot vc(P_{n-1}, x)$

[4] $\bar{a}^t$ denotes the transposed vector of the vector $\bar{a}$.

(ii) $vc(C_{n+1}, x) = x \cdot vc(C_n, x) + x^2 \cdot vc(C_{n-2}, x)$
(iii) Let $Loop_n$ be the graph which consists of n isolated loops.
$vc(Loop_{n+1}, x) = x \cdot vc(Loop_n, x) = x^n$
(iv) For the wheel graph W_n we have

$$vc(W_{n+1}, x) = x \cdot vc(W_n, x) + x^n = \\ x \cdot vc(W_n, x) + x \cdot vc(Loop_n, x)$$

Using the characteristic polynomial of the matrix, $A = (a_{i,j})$ with

$$a_{1,1} = a_{1,2} = a_{2,2} = x \text{ and } a_{2,1} = 0$$

we get

$$vc(W_{n+1}, x) = 2x \cdot vc(W_n, x) - x^2 \cdot vc(W_{n-1}, x)$$

2.4 The Tutte Polynomial

We deal now with multi-graphs (multiple edges and loops are allowed). For a graph $G = (V(G), E(G))$ we denote by $k(G)$ the number of connected components of G. We define the *rank* $r(G)$ of G by,

$$r(G) = \mid V(G) \mid - k(G)$$

and the *nullity* $n(G)$ of G by

$$n(G) = \mid E(G) \mid - \mid V(G) \mid + k(G).$$

For $F \subseteq E(G)$ we put $\langle F \rangle = (V(G), F)$, the spanning subgraph of G with edges in F. We write $k\langle F \rangle_G, r\langle F \rangle_G, n\langle F \rangle_G$ for the number of connected components, the rank and the nullity of $\langle F \rangle_G$. We omit the G in $\langle F \rangle_G$, when the context is clear.

The Tutte polynomial is now defined as

$$T_G(x, y) = \sum_{F \subseteq E} (x-1)^{r\langle E \rangle - r\langle F \rangle} (y-1)^{n\langle F \rangle}$$

There is a rich literature on the Tutte polynomial, cf. [5]. In [4], the question was studied, for which iteratively constructed sequences the Tutte polynomial can be computed with linear recurrence relations. Positive answers and explicit formulas were given for, among others, the paths P_n, the circles C_n, the ladders L_n, and the wheels W_n. To describe this phenomenon the authors called these sequences T-recursive, indicating that the Tutte polynomial T could be computed by a linear recurrence relation. In [27], a fairly general method is described by which one can obtain many iteratively constructed sequences of graphs, which are T-recursive. This method is reminiscent of graph grammars. We shall see that is no coincidence.

Instead of using the existing formal setting of graph grammars as described in [28], Noy and Ribó give an ad hoc definition of *repeated fixed succession of elementary operations*, which can be applied to a graph with a *context*, i.e. a labeled graph.

Definition 1. *Let F denote such an operation. Given a graph (with context) G, we put*

$$G_0 = G, G_{n+1} = F(G_n)$$

Then the sequence

$$\mathcal{G} = \{G_n : n \in \mathbb{N}\}$$

is called iteratively constructible *using F, or an F*-iteration sequence.

A precise version of a generalization of this definition is given in the next section, Definition 3.

2.5 The General Strategy

Given a graph polynomial $\mathfrak{P}$, such as the matching polynomial, the vertex-cover polynomial or the Tutte polynomial, and a sequence of iteratively constructible graphs G_n using an operation F, we want to compute $\mathfrak{P}(G_n)$ for all n.

To compute $\mathfrak{P}(G_{n+1})$, we try to find, depending on $\mathfrak{P}$ and, possibly, on G_0 and F, but *independently of* n, an $m \in \mathbb{N}$, auxiliary polynomials $\mathfrak{P}_i(G_{n+1}), i \leq m$, and a matrix $Q = (q_{i,j}) \in \mathbb{Z}[\bar{x}]^{m \times m}$, such that

$$\mathfrak{P}_j(G_{n+1})(\bar{x}) = \sum_i q_{i,j}(\bar{x}) \cdot \mathfrak{P}_i(G_n)(\bar{x})$$

Then we use the *characteristic polynomial of Q* to convert this into a *linear recurrence* relation.

We shall give very general sufficient conditions on the definability of $\mathfrak{P}$ and F, which will allow us to carry through such an argument.

3 Enter Logic

3.1 The Logic MSOL

Let us define some basics for the reader less familiar with Monadic Second Order Logic. A *vocabulary* τ is a set of constant, function and relation symbols. A *one-sorted τ-structure* is an interpretation of a vocabulary over *one* fixed set, the *universe*. Interpretations of constant symbols are elements of the universe, interpretations of function symbols are functions, and interpretations of relation symbols are relations of the prescribed arity. τ-terms are formed using individual variables, constant symbols and function symbols from τ. Interpretations of terms are elements of the universe. In first order logic FOL we have atomic formulas which express equality between terms and assert basic relations between terms. We are allowed to form boolean combination of formulas and to quantify existentially and universally over elements of the universe. In second order logic SOL we are allowed, additionally, to quantify over relations and functions of some fixed arity (number of arguments). In monadic second order logic MSOL, quantification over relations is restricted to unary relations, and quantification over functions is not allowed. The quantifier rank r of a formula in MSOL is defined like for FOL and without distinguishing between first order and second order quantification. An excellent reference for our logical background is [14].

3.2 MSOL-Polynomials

To understand better what many of the graph polynomials have in common we have to look closer at the way they are defined. Besides their recursive definition, like in the case of the Tutte polynomial and its close relatives, cf. [5,6,7], they usually also have an equivalent (up to some transformation) *static definition* as some kind of generating function. The matching polynomial e.g. can be written as

$$\sum_{M \subseteq E} x^{|M|} = \sum_{M \subseteq E} \prod_{e \in M} x$$

where M ranges over all subsets of edges which have no vertex in common i.e. subsets of edges which are matchings. The property of being a matching can be expressed in first order logic FOL with M a new relation variable, or in monadic second order logic MSOL, where M is a unary set variable ranging over subsets of edges.

Without going into the more delicate details, the MSOL-definable polynomials are in a polynomial ring $\mathcal{R}[\bar{x}]$ and are typically of the form

$$g(G, \bar{x}) = \sum_{A: \phi(A)} \prod_{v: v \in A} t(v) \tag{1}$$

where A is a unary relation variable, $\phi(A)$ is an MSOL-formula with A as a parameter,[5] and $t(v)$ is a term in $\mathcal{R}[\bar{x}]$ which may depend uniformly on v.

Alternatively, and more precisely, one can give an inductive definition of MSOL-polynomials as follows: First one introduces MSOL-monomials as being of the form $\prod_{v:v\in A} t(v)$, and then one closes under addition and multiplication, and under summations of the form $\sum_{A:\phi(A)} t(A)$ and multiplications of the form $\prod_{v:A(v)} t(\bar{v})$. Note that this gives more polynomials than just those of the form given in 1, due to nesting of summations and multiplications. To get a normal form of the type 1 one has to allow full second order logic, rather than MSOL.

In [26] the class of *extended* MSOL*-polynomials* is introduced. In the extended case the basic combinatorial polynomials are also included. More precisely, for every $\phi(\bar{v}) \in \mathrm{SOL}(\tau)$ and τ-structure $\mathcal{M}$ we define the cardinality of the set defined by ϕ:

$$card_{\mathcal{M},\bar{v}}(\phi(\bar{v})) = \mid \{\bar{a} \in M^m : \langle \mathcal{M}, \bar{a} \rangle \models \phi(\bar{a})\}. \mid$$

The *extended* MSOL(τ)*-polynomials* are defined inductively by allowing as extended MSOL-monomials additionally:
For every $\phi(\bar{v}) \in \mathrm{MSOL}(\tau)$ and for every $x \in \mathbf{x}$, the polynomials

$$x^{card_{\mathcal{M},\bar{v}}(\phi(\bar{v}))}, \quad x_{(card_{\mathcal{M},\bar{v}}(\phi(\bar{v})))}, \quad \binom{x}{card_{\mathcal{M},\bar{v}}(\phi(\bar{v}))}$$

[5] It may be a subgraph or induced subgraph generated by A, or a spanning subgraph generated by A, if A is a subset of edges. But the main point is that it be definable and the definition is part of ϕ.

are MSOL-definable $\mathcal{M}$-monomials. The first two are the exponentiation and the falling factorial respectively. The last is the real continuation of the number of subsets of a fixed size, see [18].[6]

Example 1. For a graph G and a non-negative integer n, let $P(G,n)$ denote the number of proper vertex colorings of G. It is well known that $P(G,n)$ is a polynomial in n, which is called the *chromatic polynomial* of G. To see this one uses a recursive definition. The static definition, given in [3,13] is not an MSOL-definable polynomial, but it is an extended MSOL-definable polynomial.

The extended MSOL-polynomials play an important role in the study initiated in [26]. The choice of extended MSOL-monomials was dictated by the characterization theorems proved in [26].

It is straight forward to see that all the results of [23], stated for MSOL-polynomials, are also valid for extended for MSOL-polynomials. In particular this applies to Theorem 2 in the sequel.

3.3 MSOL-Smooth Operations

Let $\mathfrak{A}$ and $\mathfrak{B}$ be two τ-structures. We write $\mathfrak{A} \equiv_r^{\mathrm{MSOL}} \mathfrak{B}$, if $\mathfrak{A}$ and $\mathfrak{B}$ cannot be distinguished by MSOL(τ)-formulas of quantifier rank r.

A unary operation F on τ-structures is MSOL-*smooth* if whenever $\mathfrak{A} \equiv_r^{\mathrm{MSOL}} \mathfrak{B}$, then also $F(\mathfrak{A}) \equiv_r^{\mathrm{MSOL}} F(\mathfrak{B})$.

The operation F should be MSOL-smooth for the presentation of the graphs, for which the polynomial is MSOL-definable. The presentation matters. For forming the cliques K_n we need the operation of adding a vertex connected to all previous vertices. This is MSOL-smooth for graphs $G = (V,E)$ with an edge relation E, but not for two sorted graphs $I(G) = (V \cup E, R)$. with vertices and edges as disjoint universes, and an incidence relation R.

3.4 Iteration Operations

We shall define inductively a large class of unary iteration operations which are MSOL-smooth on τ-structures enhanced with a fixed number of labels or colors.

For $k \in \mathbb{N}$, a k-τ-structure is a τ-structure with k additional unary relations $C_1^A, \ldots C_k^A$, called *colors*. We denote by τ_k the vocabulary $\tau \cup \{C_1, \ldots, C_k\}$.

Definition 2. *The following are the* basic operations *on k-τ-structures:*

$Add_i(\mathfrak{A})$**:** *For $i \leq k$, add a new element to A of color C_i.*
$\rho_{i,j}(\mathfrak{A})$**:** *For $i,j \leq k$, recolor all elements of A of color i with color j.*
$\eta_{R,i_1,\ldots,i_m}(\mathfrak{A})$**:** *For an m-ary relation symbol $R \in \tau$ and for each $a_1 \in C_{i_1}^A, \ldots, a_m \in C_{i_m}^A$ add the tuple $(a_1,\ldots,a_m)$ to R^A.*
$\delta_{R,i_1,\ldots,i_m}(\mathfrak{A})$**:** *For an m-ary relation symbol $R \in \tau$ and for each $a_1 \in C_{i_1}^A, \ldots, a_m \in C_{i_m}^A$ delete the tuple $(a_1,\ldots,a_m)$ from R^A.*

[6] The choice of these three combinatorial functions as the basic functions in the definition of extended MSOL-polynomials seems natural. However, we have not addressed the question whether this choice is *complete* in a sense yet to be defined.

Quantifier-free transductions: *For each $R \in \tau_k$ of arity $\alpha(R)$ let $\phi_R(x_1, \ldots, x_{\alpha(R)})$ be a quantifier-free τ_k formula with free variables as indicated. A quantifier free transduction redefines all the predicates R^A in $\mathfrak{A}$ by ϕ_R^A.*

Duplication: *The unary operation which associates with a graph $\mathfrak{A}$ the disjoint union of two copies of $\mathfrak{A}$.*

Remark 1. Note that the binary operation of the disjoint union of two τ-structures is not a basic operation for the purpose of iteration operations. It is, however, one of the basic operations in the induction definition of graphs of tree-width at most k or clique-width at most k, cf. [23].

Proposition 1. *All the basic operations are* MSOL*-smooth.*

We now state the key definitions for our main result.

Definition 3

(i) An operation F on τ_k-structures is MSOL-elementary *if F is a finite composition of any of the basic operations on τ_k-structures.*

(ii) Let F be MSOL*-elementary. Given a graph G, we put*

$$G_0 = G, G_{n+1} = F(G_n)$$

Then the sequence

$$\mathcal{G} = \{G_n : n \in \mathbb{N}\}$$

is called iteratively constructed *using F, or an F*-iteration sequence.

(iii) A sequence of graphs G_n is iteratively constructible *if it is an F-iteration sequence for some* MSOL*-elementary operation F.*

Proposition 2. *All the sequences E_n, K_n, P_n, C_n, L_n, W_n are F-iteration sequences for some* MSOL*-elementary operation.*

Proof. We sketch the proof for the ladders L_n, and leave the remaining cases to the reader. H_1 is L_1 with the vertices colored by the colors C_1 and C_2 respectively. H_n will be the ladder L_n with the vertices $2n-1$ and $2n$ colored with colors C_1 and C_2 respectively, and all the other vertices colored by C_0. To construct H_{n+1} we add two isolated vertices colored with the colors C_3 and C_4 respectively. Then we connect the vertices colored by C_1 and C_3, C_2 and C_4, and C_3 and C_4. Finally we recolor C_1 and C_2 by C_0, and then C_3 by C_1 and C_4 by C_2. □

The following is from [23, Sect. 2]:

Proposition 3. *Let F be* MSOL*-elementary and $\mathfrak{A}$ and $\mathfrak{B}$ two τ_k structures with $\mathfrak{A} \equiv_r^{\mathrm{MSOL}} \mathfrak{B}$. Then $F(\mathfrak{A}) \equiv_r^{\mathrm{MSOL}} F(\mathfrak{B})$, hence, F is a* MSOL*-smooth.*

Proof. The proof is straightforward from our definitions. □

The basic operations $Add_i, \rho_{i,j}$ and $\eta_{E,i,j}$ are the basic operations used to inductively define the class of graphs of clique-width at most k. The other operations are generalizations thereof. For the vocabulary of graphs, it was shown in [9] that any class of graphs, defined inductively using these operations and starting with a finite set of graphs, is of bounded clique-width. Hence we have the following:

Proposition 4. *Let F be an* MSOL*-elementary operation for k-graphs, and $\mathcal{G}$ be an F-iteration sequence. Then $\mathcal{G}$ has bounded clique-width.*

Remark 2. If we exclude the use of duplication in Proposition 4, we get that $\mathcal{G}$ has bounded linear clique-width. The notion of linear clique-width is introduced in [20].

It was shown in [11,17] that the class of square grids and the class $I(K_n)$ are of unbounded clique-width. Therefore we conclude:

Corollary 1. *The sequences $I(K_n)$, $Grid_{n,n}$ are not F-iteration sequences for any F which is* MSOL*-elementary.*

Remark 3. The notion of a *iteratively constructible* sequence of graphs, as defined in [27], cf. 1, is a special case of our F-iteration sequences for an MSOL-elementary operation F.

Remark 4. We have not attempted here to classify all the MSOL-smooth unary operations on τ-structures. Although we think that there are MSOL-smooth unary operations which are provably not MSOL-elementary, we have no example at hand. Related questions were studied in [9].

3.5 Main Result

Our main result can now be stated.

Theorem 1. *Let*

(i) F *be an* MSOL*-smooth operation on τ_k-structures;*
(ii) $\mathfrak{P}$ *be an extended* MSOL(τ)*-definable τ-polynomial;*
(iii) $\mathcal{A} = \{A_n : n \in \mathbb{N}\}$ *be an F-iteration sequence of τ-structures.*

Then $\mathcal{A}$ is $\mathfrak{P}$-iterative, i.e. there exists $\beta \in \mathbb{N}$, and polynomials $p_1, \ldots, p_\beta \in \mathbb{Z}[\bar{x}]$ such that for sufficiently large n

$$\mathfrak{P}(G_{n+\beta+1}) = \sum_{i=1}^{\beta} p_i \cdot \mathfrak{P}(G_{n+i})$$

3.6 Proof of Theorem 1

The proof of Theorem 1 uses first the splitting theorem for graph polynomials from [23]. Its scenario is as follows.

A binary operation on $k-\tau$-structures $\bowtie_X$ is MSOL-*smooth*, if whenever $\mathfrak{A} \equiv_r^{\mathrm{MSOL}} \mathfrak{B}$, and $\mathfrak{A}' \equiv_r^{\mathrm{MSOL}} \mathfrak{B}'$, then also

$$\mathfrak{A} \bowtie_X \mathfrak{A}' \equiv_r^{\mathrm{MSOL}} \mathfrak{B} \bowtie_X \mathfrak{B}'.$$

Here X is used to indicate the dependence on the particular choice of the MSOL-smooth binary operation.

Let $\mathfrak{P}_i^r$, $i \in I_r$, the set of all extended MSOL-definable graph polynomials with defining formulas of quantifier rank at most r. I_r is finite of size α_r, as there are, up to logical equivalence, only finitely many formulas of fixed quantifier rank r.

A sharpened form of the splitting theorem [23, Theorem 6.4] now states the following:

Theorem 2 (Bilinear Splitting Theorem). *Let $\bowtie_X$ be an* MSOL-*smooth binary operation. There exists $A(X) = (a_{i,k,\ell}(X)) \in \{0,1\}^{\alpha_r \times \alpha_r \times \alpha_r}$ such that*

$$\mathfrak{P}_i^r(\mathfrak{A} \bowtie_X \mathfrak{B}) = \sum_{k,\ell \leq \alpha_r} a_{i,k,\ell}(X) \cdot \mathfrak{P}_k^r(\mathfrak{A}) \cdot \mathfrak{P}_\ell^r(\mathfrak{B})$$

Proof (Sketch). The Bilinear Splitting Theorem is a refinement of the Feferman-Vaught Theorem for MSOL. Its proof is exactly as the proof of [23, Theorem 6.4]. The generalization to extended MSOL-definable polynomials is straight forward. □

The next step in the proof consists of a characterization of the MSOL-elementary operations F.

Proposition 5. *Let F be an* MSOL-*elementary operation on τ_k-structures where exactly m many new elements are added. Then there exists a τ_k-structure $\mathfrak{C}_F$ of size m and a binary* MSOL- *smooth operation $\bowtie_F$ such that for all τ_k-structures $\mathfrak{A}$ we have*

$$F(\mathfrak{A}) = \mathfrak{C}_{\mathfrak{F}} \bowtie_F \mathfrak{A}$$

Proof (Sketch). The proof is by induction on the sequence of basic operations used in the definition of F. □

Now we define

$$\mathfrak{q}_{i,\ell} = \sum_k a_{i,k,\ell} \cdot \mathfrak{P}_k^r(\mathfrak{C})$$

and use the Bilinear Splitting Theorem. We obtain:

$$\mathfrak{P}_i^r(F(\mathfrak{A})) = \sum_{\ell \leq \alpha_r} \mathfrak{q}_{i,\ell} \cdot \mathfrak{P}_\ell^r(\mathfrak{B})$$

The matrix $Q = (\mathfrak{q}_{i,\ell})$ is a matrix of polynomials. To obtain Theorem 1 we compute the characteristic polynomial

$$\chi(Q) = \sum_{i=0}^{\alpha_r} q_i \lambda^i$$

of Q and obtain the required linear recurrence relation with

$$\beta = \alpha_q \text{ and } p_i = -q_i \text{ for } i = 0, \dots \alpha_r - 1$$

where q_i are the coefficients of $\chi(Q)$. Note that $q_{\alpha_r} = 1$.

This completes the proof of Theorem 1.

4 Conclusions and Further Research

We have introduced the class of MSOL-elementary operations F of τ-structures and their associated F-iteration sequences. We have shown that a very wide class of graph polynomials, and even of polynomial invariants of general τ-structures, can be computed on F-iteration sequences by linear recurrence relations. This explains a widely observed, but not systematically studied phenomenon.

As a consequence of our method we get immediately the following.

Corollary 2. *Let F be an* MSOL*-smooth operation, G_n be an F-iteration sequence and $\mathfrak{P}$ an extended* MSOL*-definable graph polynomial. Then $\mathfrak{P}(G_n)$ can be computed in polynomial time in n.*

Proof (Sketch). We first observe that the size of G_n is linear in n. besides that, proof is the same as the one given in [10,23]. $\square$

Although our method is theoretically computable, it is not effective for several reasons:

- The number α_r is too large for practical use.
- The boolean array $a_{i,k,\ell}$ cannot be efficiently computed. In fact, its computation is non-elementary, cf. [16].

It is likely that in practice, for explicitly given F, a much smaller recurrence relation can be found explicitly. This remains a challenging topic for further research.

Our results can be extended in several ways:

(i) We can add one more MSOL-smooth basic operation, provided the vocabulary τ contains only unary and binary relation symbols.

Fuse: The graph $fuse_i(G)$ is obtained from G by identifying all vertices of color C_i and leaving all the resulting edges with the exception of the resulting loops.

A detailed discussion of this and further operations may be found in [23, Sect. 3] and in [9]. We do not go into further details, due to space limitations.

(ii) The logic MSOL can be extended to the logic CMSOL where we have additionally modular counting quantifiers $C_{m,n} x \phi(x)$ for each $m, n \in \mathbb{N}$, which say that there are, modulo m, exactly n elements satisfying $\phi(x)$. The Splitting Theorem in [23] is proven for CMSOL. Clearly CMSOL is a sublogic of SOL. The interlace polynomials, [1,2] are CMSOL-definable, but it is not known whether they are MSOL-definable.

(iii) In [15] the notion of clique-width was generalized further and the notion of patch-width was introduced. For F an MSOL-elementary operation the F-iteration sequences of arbitrary τ-structures are of bounded patch-width. It remains to be investigated whether there are interesting polynomial invariants of τ-structures, where our method leads to useful results.

Our method does not apply to square grids $G_{n,n}$. But they are obviously regularly constructed in some way, and it is to be expected that for most graph polynomials $\mathfrak{P}$, some recurrence relation does exist to compute the values of $\mathfrak{P}(G_{n,n})$. Can one find a general theorem which captures this intuition?

Acknowledgments

We would like to thank the referee for has very careful reading of the first version of this paper. We would also like to thank the editors of this volume, Arnon Avron, Nachum Dershowitz, and Alexander Rabinovich, for their encouragement and patience.

References

1. Arratia, R., Bollobás, B., Sorkin, G.B.: The interlace polynomial: a new graph polynomial. Journal of Combinatorial Theory, Series B 92, 199–233 (2004)
2. Arratia, R., Bollobás, B., Sorkin, G.B.: A two-variable interlace polynomial. Combinatorica 24(4), 567–584 (2004)
3. Biggs, N.: Algebraic Graph Theory, 2nd edn. Cambridge University Press, Cambridge (1993)
4. Biggs, N.L., Damerell, R.M., Sand, D.A.: Recursive families of graphs. J. Combin. Theory Ser. B 12, 123–131 (1972)
5. Bollobás, B.: Modern Graph Theory. Springer, Heidelberg (1999)
6. Bollobás, B., Riordan, O.: A Tutte polynomial for coloured graphs. Combinatorics, Probability and Computing 8, 45–94 (1999)
7. Chung, F.R.K., Graham, R.L.: On the cover polynomial of a digraph. Journal of Combinatorial Theory, Ser. B 65(2), 273–290 (1995)
8. Courcelle, B.: A multivariate interlace polynomial (December 2006) (preprint)
9. Courcelle, B., Makowsky, J.A.: Fusion on relational structures and the verification of monadic second order properties. Mathematical Structures in Computer Science 12(2), 203–235 (2002)
10. Courcelle, B., Makowsky, J.A., Rotics, U.: On the fixed parameter complexity of graph enumeration problems definable in monadic second order logic. Discrete Applied Mathematics 108(1–2), 23–52 (2001)
11. Courcelle, B., Olariu, S.: Upper bounds to the clique–width of graphs. Discrete Applied Mathematics 101, 77–114 (2000)
12. Dong, F.M., et al.: The vertex-cover polynomial of a graph. Discrete Mathematics 250, 71–78 (2002)
13. Dong, F.M., Koh, K.M., Teo, K.L.: Chromatic Polynomials and Chromaticity of Graphs. World Scientific, Singapore (2005)
14. Ebbinghaus, H.D., Flum, J.: Finite Model Theory. In: Perspectives in Mathematical Logic, Springer, Heidelberg (1995)

15. Fischer, E., Makowsky, J.A.: On spectra of sentences of monadic second order logic with counting. Journal of Symbolic Logic 69(3), 617–640 (2004)
16. Frick, M., Grohe, M.: The complexity of first-order and monadic second-order logic revisited. Annals of Pure and Applied Logic 130(1), 3–31 (2004)
17. Golumbic, M.C., Rotics, U.: On the clique-width of some perfect graph classes. Internation Journal of Foundations of Computer Science 11, 423–443 (2000)
18. Graham, R., Knuth, D., Patashnik, O.: Concrete Mathematics, 2nd edn. Addison-Wesley, Reading (1994)
19. Gurevich, Y.: Modest theory of short chains, I. Journal of Symbolic Logic 44, 481–490 (1979)
20. Gurski, F., Wanke, E.: On the relationship between NLC-width and linear NLC-width. Theoretical Computer Science 347(1–2), 76–89 (2005)
21. Kobrinski, N.E., Trakhtenbrot, B.A.: Introduction to the Theory of Finite Automata. In: Studies in Logic and the Foundations of Mathematics, North-Holland, Amsterdam (1965)
22. Läuchli, H.: A decision procedure for the weak second order theory of linear order. In: Logic Colloquium 1966, pp. 189–197. North-Holland, Amsterdam (1968)
23. Makowsky, J.A.: Algorithmic uses of the Feferman-Vaught theorem. Annals of Pure and Applied Logic 126, 1–3 (2004)
24. Makowsky, J.A.: From a zoo to a zoology: Towards a general theory of graph polynomials. Theory of Computing Systems, published online first (July 2007)
25. Makowsky, J.A., Mariño, J.P.: Farrell polynomials on graphs of bounded treewidth. Advances in Applied Mathematics 30, 160–176 (2003)
26. Makowsky, J.A., Zilber, B.: Polynomial invariants of graphs and totally categorical theories. MODNET Preprint No. 21 server (2006), `http://www.logique.jussieu.fr/modnet/Publications/Preprint%20`
27. Noy, M., Ribó, A.: Recursively constructible families of graphs. Advances in Applied Mathematics 32, 350–363 (2004)
28. Rozenberg, G. (ed.): Handbook of graph grammars and computing by graph transformations, Foundations, vol. 1. World Scientific, Singapore (1997)
29. Shelah, S.: The monadic theory of order. Annals of Mathematics 102, 379–419 (1975)
30. Trakhtenbrot, B.: Finite automata and the logic of monadic predicates. Doklady Akademy Nauk SSSR 140, 326–329 (1961)
31. Trakhtenbrot, B.: Some constructions in the monadic predicate calculus. Doklady Akademy Nauk SSSR 138, 320–321 (1961)