FARRELL POLYNOMIALS ON GRAPHS OF BOUNDED TREE WIDTH

J.A. MAKOWSKY[1] AND J.P. MARIÑO[2]

Abstract. We consider various classes of graph polynomials and study their computational complexity. Our main focus is on Farrell polynomials and generating functions of graph properties. All these polynomials have a wide range of applications in in combinatorics, but also in physics, chemistry and biology. In general the worst case complexity of most these polynomials is known to be **NP**, or even ♯**P** hard. We show that such polynomials characterized by a definability condition in the formalisms of Monadic Second Order Logic can be computed in polynomial time if restricted to graphs of tree width at most k.

Résumé. Nous considérons plusieurs classes de polynômes graphiques et nous étudions leurs complexite algorithmique. L'accent est mis sur les polynômes de Farrell et les fonctions génèratrices de proprietés des graphes. Tous ces polynômes ont des vastes applications en combinatoire, mais aussi en physique, chimie et biologie. En génèral la complexite algorithmique de ces polynômes est bornée infèrieurement par des problèmes complets pour **NP** ou ♯**P**. Nous montrons que pour des polynômes définissables par des expressions en logique monadique de second ordre, ils sont calculables en temps polynomiel si l'on se restreint aux graphes de largeur arborescencente bornée.

1. Introduction

In the literature of graph theory we find an abundance of graph polynomials. Some are studied because they are generating functions of certain graph properties, others count the number of occurencies of certain features of the graphs, and finally, others are attempts to find complete graph invariants. Most of them are hard to compute, in the sense that they are ♯**P** hard or at least **NP** hard. In this paper we examine all the polynomials we could find in the literature (we may have missed some more exotic examples), and see whether the dynamic programming techniques based on various definability criteria related to Monadic Second Order Logic developed for graphs of bounded tree width can be applied.

One of the more intriguing graph polynomials, the Tutte polynomial, [Wel93, Bol99, BR99], is known to be hard to compute, ♯**P**-hard to be precise. Andrzejak and, independently, Noble, cf. [And98, Nob98], have shown that the Tutte polynomial is computable in polynomial time on graphs of fixed bounded tree width. In [Mak00], this result is extended to the colored Tutte polynomials building on previous work by Arnborg, Courcelle, Lagergren, Makowsky, Rotics and Seese [ALS91, CMR00a, CMR00b]. Our proof had two ingredients, one *logical* and one *combinatorial*. The first consists in establishing that the polynomial in question can be expressed in a certain formalism of sums of products where the summation ranges over subsets of edges satisfying a condition in Monadic Second Order

[1] partially supported by a Grant of the Fund for Promotion of Research of the Technion–Israeli Institute of Technology.

[2] partially supported by a postdoctoral grant of the Technion.

Logic ($MSOL$), and the product ranges over weights on edges with a bounded number of case distinction also definable in $MSOL$. The second consists in exploiting a tree decomposition of the graph on which the polynomial is to be computed inductively. We produce here an abstract theorem (Theorem 6) describing a very general situation where these ingredients yield polynomial time algorithms which extends the scope of applicability beyond the theorem given in [CMR00b, Mak01, Mak00]. However, it is sometimes not at all obvious how to recognize whether a graph polynomial is $MSOL$ definable in the above sense.

The purpose of this paper is to exhibit a wide range of graph polynomials which are hard to compute on arbitrary graphs but for which the existence of polynomial time algorithms can be established for graphs of bounded tree width using directly or indirectly our general method. Some of the cases were previously treated in [CMR00b, Mak01, Mak00]. But here we push the applicability of these results further, and, most importantly, our new Theorem 6 covers new cases.

Our results apply to the following cases:

(i) The colored Tutte polynomials and all the polynomials which are substitution instances thereof, especially Kauffman brackets and other knot polynomials viewed as graph polynomials. [BR99]. This was shown in [Mak01, Mak00]. This also includes Martin polynomials and the interlace polynomial, but only when restricted to link-like graphs, cf. [EM98, ABS].

(ii) Farrell's F-polynomials for $\mathcal{F}$ covers, which we simply call *calF*-polynomials, cf. [Far79b], with $\mathcal{F}$ a *finite* set of graphs. This includes both the *matching defect polynomial*, the *matching generating polynomial*, cf. [LP86], and their generalizations, cf. [God81, God93, Far79a], the various *rook polynomials* of Riordan, [Rio58], and the like. These results are *new and do not follow from any of the results described in* [CMR00b, Mak01, Mak00].

(iii) Many enumeration polynomials, cf. [Wil90, Sta86]. These include the *clique and independent set polynomials* of [HL94] and the literature cited therein. These results are also new in as much as the framework does not correspond exactly to the one presented in [CMR00b].

On the other hand, our method does not seem directly applicable to

(i) The *bivariate permanent polynomial*, cf. [BP80, Par99]. Here the polynomials defined using both the graph G and its complement, but the tree width of the combined input is always maximal.

(ii) The *hit polynomials*, cf. [God93]. Although they are generalizations of the matching polynomials, the polynomial depends on a bipartite graph B on $2n$ vertices and the graph $K_{n,n}$, hence the tree width of the combined object is again maximal.

(iii) the *frame polynomial* of [BP80, Par99], although this is a Farrell polynomial. Here the reason lies in the weight function which gives a different formal variable to each of the basic graphs in $\mathcal{F}$ which is infinite.

(iv) the Martin polynomial on general undirected graphs, cf. [EM98, Ver83].

Nevertheless we conjecture that on classes of graphs of bounded tree width these polynomials are also computable in polynomial time.

The paper is organized as follows: In section 2 we introduce a general form of our polynomials, formulate the definability conditions in Monadic Second Order Logic, define the $MSOL$-polynomials, and discuss some immediate examples. In section 3 we define the Farrell polynomials and discuss definability conditions for Farrell polynomials. In section

4 we recall the definition of tree width of a graph and state our main theorem (Theorem 6). In section 5 we discuss limitations of our theorems and open problems, and formulate various conjectures.

2. $MSOL$-POLYNOMIALS

2.1. **General form of graph polynomials.** Our polynomials are all of the form

(general) $$p(G) = \sum_{subgraphs H} c(H) \left(\prod_{case_1} w_1 \cdot \ldots \cdot \prod_{case_\alpha} w_\alpha \right)$$

with

(i) the summation ranges over a family $\mathcal{H}$ of subgraphs of G;
(ii) the products range over elements or subsets of vertices and edges of H, distributed over α many cases, where α is a fixed integer.
(iii) $w_\beta : \beta \leq \alpha$ are weight functions of subsets of vertices or edges with values in some fixed ring $\mathcal{R}$.
(iv) $c(H)$ is a coefficient depending on the subgraph $H \in \mathcal{H}$.

A typical example is the *characteristic polynomial* $\Phi_{char}(G, \lambda)$ of a graph G defined by

$$\Phi_{char}(G, \lambda) = det(\lambda I - A(G))$$

where $A(G)$ is the adjacency matrix of a graph. We put $M = \lambda I - A(G)$ and interpret it as a weight function on $K_{n,n}$ where the weight of (i, j) is given by $m_{i,j}$. A perfect matching E of $K_{n,n}$ can also be viewed as a graph of a permutation π_E of $\{1, \ldots n\}$. Now

(char) $$\Phi_{char}(G, \lambda) = det(\lambda I - A(G)) = \sum_E c(E) \prod_{(i,j) \in E} m'_{i,j} = \sum_T \prod_{(i,j) \in E} m_{i,j}$$

where E ranges over the perfect matchings of G, T ranges over spanning trees, and $c(E) = (-1)^{sign(\pi_E)}$. The last equality is Kirchhoff's theorem on counting spanning trees, with $(m'_{i,j}$ a matrix computable from $(m_{i,j}$ in linear time, hence, the coefficient $c(E)$ can be avoided. For details, cf. [Eve79, Bol79].

Similarly, the *bivariate permanent polynomial* $\Phi_{perm}(G, \lambda, \mu, \nu)$ of a graph (without loops) G is defined by

(per) $$\Phi_{perm}(G, \lambda, \mu, \nu) = per(\lambda I + \mu \cdot A(G)) + \nu \cdot \bar{A}(G))$$

where $\bar{A}(G)$ is the adjacency matrix of the complement graph. Again, for $M' = (\lambda I + \mu \cdot A(G)) + \nu \cdot \bar{A}(G))$ we can write this in the required form

(per') $$\Phi_{perm}(G, \lambda, \mu, \nu) = per(M') = \sum_E \prod_{(i,j) \in E} m'_{i,j}$$

This family of polynomials was introduced in [BP80] as a candidate for a complete graph invariant, cf. also [Par99]. Note that the definition of the weights in M' requires a case distinction.
Complexity: Clearly, $\Phi_{char}(G, \lambda)$ can be computed in time $O(|G|^3)$ whereas the computation of $\Phi_{perm}(G, \lambda, \mu, \nu)$ is $\sharp\mathbf{P}$ hard.

2.2. **Monadic Second Order Logic.** For our further developments we shall require that both the family $\mathcal{H}$ of subgraphs and the case distinction be definable in the formalism of Monadic Second Order Logic, as used in [CMR00b].

For this, we view our graph G as a two sorted structure $\mathcal{G}$ with vertices V and edges E in its universe. Furthermore we have a binary relation $R \subset V \times E$ with $(v, e) \in R$ iff v is a vertex of e.

The Monadic Second Order Logic $MSOL(\mathcal{G})$ on such structures has variables v_i for vertices, x_i for edges, U_j for subsets of vertices and X_j for subsets of edges. Atomic formulas are

$$v_i \in U_j, x_i \in X_j, R(v_i, x_j), v_i = v_j, x_i = x_j$$

The formulas are now defined inductively using boolean connectives and quantification over the variables v_i, x_i, U_i and X_i.

The Monadic Second Order Logic $MSOL(\mathcal{G}_{colored})$ allows additionally unary predicates C_λ, $\lambda \in \Lambda$, which are interpreted as the edges of color λ. Hence we have additional atomic formulas $C_\lambda(x_i)$. This extension of $MSOL(\mathcal{G})$ is necessary to deal with Tutte polynomials of signed graphs and the colored Tutte polynomial.

For a detailed discussion of the expressive power of $MSOL(\mathcal{G})$ and $MSOL(\mathcal{G}_{colored})$ we refer to [ALS91]. Here are some easily verifiable examples:

Proposition 1. *The following are expressible in $MSOL(\mathcal{G})$:*

(i) $\phi_{pmatch}(X)$: *X is a partial matching.*
(ii) $\phi_{path}(v_1, v_2, X, R)$: *$X$ is a path from v_1 to v_2 in G.*
(iii) $\phi_{component}(U, R)$: *U is a connected component of G.*
(iv) $\phi_{tree}(X, U, R)$: *X is a spanning tree of the component U of G.*
(v) $\phi_{forest}(F, R)$: *F is a spanning forest of G.*
(vi) $\phi_{clique}(C)$: *C is a clique of G.*
(vii) $\phi_{maxclique}(C)$: *C is a maximal clique of G.*
(viii) $\phi_{indep}(I)$: *I is an independent set of G.*

2.3. $MSOL$**-polynomials.** $MSOL$-polynomials are now of the form

$$p(G) = c \sum_{subgraphs H} \left(\prod_{case_1} w_1 \cdot \ldots \cdot \prod_{case_\alpha} w_\alpha \right) \tag{MSOL}$$

where the summation ranges over an $MSOL$-definable family of subgraphs H and the products range over all edges and vertices of H with a $MSOL$-definable finite case distinction where each case receives the same weight. Furthermore the coefficient $c(H) = c$ is assumed to be constant[1].

Proposition 2. *The characteristic polynomial and the permanent polynomial are $MSOL$-polynomials.*

2.4. **Matching and rook polynomials.** We follow [God93, LP86]. The *generating matching polynomials* $g(G, \lambda)$, the *defect matching polynomials* $m(G, \lambda)$ and the *rook polynomials*

[1]A more general definition still works for our main theorems, but the additional complications do not allow us to include this in the extended abstract

$\rho(B,\lambda)$ of a graph G, respectively bipartite graph $B \subset K_{n,n}$ are defined as

$$\text{(gen-m)} \qquad g(G,\lambda) = \sum_r p(G,r)\lambda^r = \sum_{E' \subseteq E(G)} \prod_{e \in E'} \lambda$$

$$\text{(def-m)} \qquad m(G,\lambda) = \sum_r (-1)^r p(G,r)\lambda^{n-2r} = \lambda^{-n} \sum_{E' \subseteq E(G)} \prod_{e \in E'} (-1) \cdot \lambda^{-2}$$

$$\text{(rook)} \qquad \rho(B,\lambda) = \sum_r (-1)^r p(B,r)\lambda^{n-r} = \lambda^{-n} \sum_{E' \subseteq E(B)} \prod_{e \in E'} (-1) \cdot \lambda^{-1}$$

where $p(G,r)$ denotes the number of (partial) r-matchings of G, and E' ranges over the partial matchings of G, respectively B.

Proposition 1 gives immediately

Proposition 3. *Both matching polynomials and the rook polynomial are MSOL-polynomials.*

Complexity: Both matching polynomials and the rook polynomials are $\sharp\mathbf{P}$ hard to compute. This follows from the fact that $p(G, \frac{n}{2})$ is already $\sharp\mathbf{P}$ hard to compute for bipartite graphs, cf. [Val79, God93]. Noble in his thesis, [Nob97] has shown that on graphs of tree width k the matching polynomials can be computed in polynomial time.

2.5. **Clique and independent set polynomials.** We follow [HL94]. Let $a_k(G)$, $\bar{a}_k(G)$ and $b_k(G)$ be the number of k-cliques, maximal k-cliques and k-independent sets of G, with $a_0(G) = b_0(G) = 1$. The *clique polynomials* $C(G;x)$ the *maximal clique polynomials* $MC(G;x)$ and *independent set polynomials* $I(G;x)$ are defined as

$$\text{(clique)} \qquad C(G;x) = \sum_k a_k(G)x^k = \sum_{C \subset v(G)} \prod_{v \in C} x$$

$$\text{(m-clique)} \qquad MC(G;x) = \sum_k \bar{a}_k(G)x^k = \sum_{C \subset v(G)} \prod_{v \in C} x$$

$$\text{(indep)} \qquad I(G;x) = \sum_k b_k(G)x^k = \sum_{I \subset v(G)} \prod_{v \in I} x$$

where the sum in the right most term ranges over the cliques, maximal cliques and independent sets, respectively. $C(G;x)$ and $I(G;x)$ were studied in various contexts by Fisher and Solow [FS90] and Gutman and Harary [GH83] and Hoede and Li [HL94].

Proposition 1 gives immediately

Proposition 4. *The clique polynomials, maximal clique polynomials and the independent set polynomials are MSOL-polynomials.*

Complexity: It is easy to see that $C(G;x)$ and $I(G;x)$ are $\mathbf{NP}$ hard to compute and that $MC(G;x)$ is $\sharp\mathbf{P}$ hard. In [HL94] is is noted that for k-trees G one has

$$C(G;x) = x(1+x)^k(1+(n-k)x).$$

Here n is the size of vertex set $V(G)$. Hence, for k-trees G the computation of the polynomial is trivially in $\mathbf{P}$. We shall extend this result to partial k-trees, i.e. graphs of tree width at most k.

3. Farrell polynomials

We follow and extend the framework given in [Far79b]. Let $\mathcal{F}$ be a family of connected graphs[2], closed under isomorphisms. For each $D \in \mathcal{F}$ (up to isomorphism) let X_D be an indeterminate and let $R_{\mathcal{F}} = \mathbb{Z}[X_D : D \in \mathcal{F}]$ be the polynomial ring associated with $\mathcal{F}$.

Given a graph $G = \langle V, E \rangle$, a *vertex disjoint $\mathcal{F}$-cover C of G* is a spanning subgraph of G where each connected component D of C is in $\mathcal{F}$. An *edge disjoint $\mathcal{F}$-cover C of G* is a spanning subgraph $\langle V, E' \rangle$ of G where $E' = \bigsqcup_i E_i$ is partitioned as the disjoint union of edge sets E_i (also called components) such that the underlying graph of E_i is in $\mathcal{F}$. Here different components may overlap at vertices. A $\mathcal{F}$-cover is either vertex disjoint or edge disjoint and we specify which if needed.

With each $\mathcal{F}$-cover C of G we associate a monomial $\mu_G(C) = \prod_{D \subseteq_{comp} C} X_D$. The $\mathcal{F}$-polynomial of G is now defined as

$$\sum_{C \text{ is an } \mathcal{F}-\text{cover of } G} \mu_G(C) \tag{Farrell}$$

A *Farrel polynomial* is a substitution instance of an $\mathcal{F}$-polynomial for suitable $\mathcal{F}$ and weight function w mapping indeterminates into some value of $R_{\mathcal{F}}$. Often the value set of the weight function w is assumed to be finite.

Let $\mathcal{F}$ consist of the degenerate cliques K_1 and K_2. A vertex disjoint $\mathcal{F}$-cover C is an m-matching with $m = m(C)$ the number of K_2's in C. To see that the matching polynomial $m(G, \lambda)$ is a Farrell polynomial we put $X_{K_1} = 1$ and $X_{K_2} = -\lambda^{-2}$.

Other examples, with vertex disjoint covers, include:

(i) Rook polynomials of bipartite graphs, cf.[Rio58]; Here $\mathcal{F}$ is like for the matching polynomial and contains two graphs.
(ii) The circuit polynomials, where $\mathcal{F}$ consists of all proper and improper cicuits, hence $\mathcal{F}$ is infinite.
(iii) The characteristic polynomial of a graph can be obtained as a special case of the cicuit polynomial with proper choice of the weight function, cf. [Sac64, Far79b].
(iv) The subgraph polynomial, where $\mathcal{F}$ consists of all connected finite graphs. The Tutte polynomials (also called dichromatic polynomials), are of this form for a properly chosen weight function [3], cf. [Far79b, Bol99].
(v) Using edge disjoint covers, we can also include the *Martin polynomials for undirected graphs*, cf. [Ver83, EM98].

For a Farrell polynomial to be an $MSOL$-polynomial it is not enough to require that $\mathcal{F}$ be $MSOL$-definable, as the weight function on $\mathcal{F}$ can depend on more than one edge or vertex of $D \in \mathcal{F}$ and its range can be infinite, when $\mathcal{F}$ is. The weight function in the case of $MSOL$-polynomials is a function of the vertices or edges alone.

We say that a graph polynomial is *$MSOL$-Farrell*, if

(i) $\mathcal{F}$ is $MSOL$-definable,
(ii) the $\mathcal{F}$-covers are vertex disjoint,
(iii) the range A of the weight function w is finite,
(iv) and for each $a \in A$ the set $\{D \in \mathcal{F} : w(D) = a\}$ is $MSOL$-definable.

[2] One could also consider non-connected graphs, but no reasonable examples occur in the literature

[3] This does not apply to the colored Tutte polynomials.

Remark 1. *If the $\mathcal{F}$-cover is vertex disjoint then the F-polynomial can be written as*

$$\sum_{E_0 \subseteq E} \left(\prod_{D \in CC(\langle V, E_0 \rangle)} \mu_G(D) \right) \tag{Farrell-1}$$

with $CC(\langle V, E_0 \rangle)$ the set of connected components of $\langle V, E_0 \rangle$ and

$$\mu_G(D) = \begin{cases} 0 & \text{if } D \notin \mathcal{F} \\ X_D & \text{else} \end{cases}$$

Proposition 5. *If $\mathcal{F}$ is finite, then every Farrell polynomial defined using $\mathcal{F}$ is $MSOL$-Farrell.*

Proof. We only observe that every finite class of structures is $MSOL$-definable. □

The *frame polynomials* are the vertex disjoint Farrell polynomials where $\mathcal{F}$ consists of circles $C_n : n \geq 3$, and paths P_n, hence $\mathcal{F}$ is $MSOL$-definable, but the weight P_n is X_n and of C_n is Y_n, i.e. the range of the weight function on $\mathcal{F}$ is infinite. They are not $MSOL$-Farrell. The *bounded frame polynomials*, defined like the frame polynomials, but with $\mathcal{F}$ truncated containing only the C_n's and P_n's for $n \leq N$, are $MSOL$-Farrell.

4. $MSOL$-POLYNOMIALS ON GRAPHS OF BOUNDED TREE WIDTH

We now recall the definition of *tree width of a graph $G = \langle V, E \rangle$*.

4.1. Tree width.

General background on tree width may be found in [Die96].

Definition 1. *A* k-tree decomposition of G *is given as follows:*

(i) *We have a rooted tree $\mathcal{T} = \langle T, f \rangle$, where T is a set and f is a function mapping nodes onto their father.*
(ii) *The vertex set $V(G)$ of the graph covered by sets A_t, with $t \in T$ and $|A_t| \leq k + 1$.*
(iii) *For every edge $e = (x, y) \in E(G)$ there is a $t \in T$ such that both $x, y \in A_t$.*
(iv) *For each $x \in V$ the set $T(x) = \{t \in T : x \in A_t\}$ is a (connected) subtree of $\mathcal{T}$.*

Remark 2. *Under conditions (i)-(iii), (iv) is equivalent to: For every connected subgraph H of G, the set $\{t \in T : V(H) \cap A_t \neq \emptyset\}$ is a connected subtree of $\mathcal{T}$*

Definition 2. G is of tree width at most k, *if there exists a k-tree decomposition of G. Such graphs are also called* partial k-trees.

For fixed k, checking whether G has tree width at most k (and if yes, finding a witnessing tree decomposition) can be done in polynomial time, cf. [Bod97].

If we add unary predicates (labels) to G, the notion of tree width does not change. Therefore the tree width of a crossing diagram is just the tree width of its underlying graph. Also the tree width of an edge colored graph is, by definition, the same as its tree width without the coloring.

The tree width of K_n is $n - 1$ and of $K_{n,n}$ it is n. If the graph polynomial $P(G, \bar{X})$ depends also on the complement graph of G, as is the case for the permanent polynomial Φ_{perm} or the hit polynomial, the restriction of $P(G, \bar{X})$ to graphs of tree width at most k may seem to be meaningless.

4.2. **Main theorem.** Modifying the proof of [CMR00b] for our definition of $MSOL$-polynomials and $MSOL$-Farrell polynomials we get:

Theorem 6. *Let K be class of graphs of tree width at most k. Let $\Phi(G, \bar{X})$ be a graph polynomial which is either an $MSOL$-polynomial or $MSOL$-Farrell. Then $\Phi(G, \bar{X})$ can be computed on K in polynomial time and even in* **NC**. *Furthermore, $\Phi(G, \bar{X})$ can be evaluated in linear time (over the ring with unit cost).*

Outline. For the case that $\Phi(G, \bar{X})$ is an $MSOL$-polynomial, we proceed as in [CMR00b], but compute in the polynomial ring $R_{\mathcal{F}}$, using a unit cost computation model for operations in the ring.

For the case that $\Phi(G, \bar{X})$ is an $MSOL$-Farrell, we have to rework the proof of [CMR00b] considerably.

In this case the assumption that the covers are vertex disjoint and that the weights are given to the connected components of the cover are essential. We exploit remark 1, after the definition of $MSOL$-Farrel polynomials, and remark 2, after the definition of k-tree decompositions together with the finiteness of the range of w and the definability condition of its preimages. Again we use the tree decomposition as the basis for an inductive computation, but we have to track the connected components of the cover on the way, which creates additional complications. This results from the fact, that neither the decomposition theorem à la Feferman-Vaught (as in [CMR00b]), nor the automata theory approach (as in [ALS91]) give us enough information on how to handle the various connected components of the $\mathcal{F}$-cover. The details will be given in the full paper. □

5. Limitations of the methods

From theorem 6 (and the results in [Mak00, Mak01]) it follows that the following polynomials can be computed in polynomial time on graphs of tree width at most k:

- **Tutte::** The colored Tutte polynomials.
- **Knots::** The various Kauffman brackets and Jones polynomials.
- **Matching::** The various matching and rook polynomials. (but not the hit polynomials).
- **Clique::** The clique and maximal clique polynomial.
- **Indep::** The independent set polynomial.
- **Bounded frame::** The bounded frame polynomials.

It is likely that for each of these cases specific methods will lead to much better algorithms, as in the case of matching polynomials and Tutte polynomials (cf. [Nob97]. Once one knows of the existence of a polynomial time algorithm, it is worthwhile searching for a feasible one, cf. [DF99].

Our method does not apply to the following:
The Martin polynomials, which are Farrell polynomials for edge disjoint covers. However, for graphs representing link diagrams (e.g. eulerian and regular of degree 4) they can be obtained from the Kauffman bracket, cf. [ABS]. Based on recent work on Martin polynomials on arbitrary graphs in [EM98] we conjecture:

Conjecture 1. *On graphs of tree width at most k the Martin polynomials can be computed in polynomial time.*

However, for the proof we have to use some new ideas.

The permanent polynomial, because the permanent is the permanent of a matrix without any zeros. There is an interesting question here:

Problem 1. *Assume M is a matrix with only two values a, b both different from zero. How do the values have to be distributed such that $per(M)$ can be computed fast?*

The answer to this probelm may also shed light on computing the hit polynomial.
The frame polynomials, because the weight P_n is X_n and of C_n is Y_n, i.e. the range of the weight function on $\mathcal{F}$ is infinite. Nevertheless, we conjecture:

Conjecture 2. *On graphs of tree width at most k the frame polynomials can be computed in polynomial time.*

It is not yet clear how the newly introduced *interlace polynomials* of Arratia, Bollobás and Sorkin [ABS] fit into our framework. For graphs representing link diagrams the are also computable from the Kauffman brackets, as shown in [ABS].

Problem 2. *Are the interlace polynomials computable in polynomial time on graphs of bounded tree width?*

Finally, our methods may have some bearing on the search of complete invariant for graphs of bounded tree width. A graph invariant is complete if it distinguishes non-isomorphic graphs. In [GM99], it is shown that there is a graph canonization, and hence a complete graph invariant, for graphs of bounded tree width which is computable in polynomial time. However, this canonozation is not given as a graph polynomial.

Problem 3. *Is there a* natural *family $Inv(k)$ of polynomials which is a complete invariant for graphs of tree width at most k? The family of frame polynomials and of permanent polynomials may be good candidates.*

References

[ABS] R. Arratia, B. Bollobas, and G.B. Sorkin. The interlace polynomial: A new graph polynomial. IBM Research Report, RC 21813 (98165), 31 July 2000.

[ALS91] S. Arnborg, J. Lagergren, and D. Seese. Easy problems for tree decomposable graphs. *Journal of Algorithms*, 12:308–340, 1991.

[And98] A. Andrzejak. An algorithm for the Tutte polynomials of graphs of bounded treewidth. *Discrete Mathematics*, 190:39–54, 1998.

[Bod97] H. Bodlaender. Treewidth: Algorithmic techniques and results. In I. Privara and P. Ruzicka, editors, *Proceedings of the 22th International Symposium on the Mathematical Foundation of Computer Science, MFCS'97*, volume 1295 of *Lecture Notes in Computer Science*, pages 29–36. Springer, 1997.

[Bol79] B. Bollobás. *Graph Theory*. Springer, 1979.

[Bol99] B. Bollobás. *Modern Graph Theory*. Springer, 1999.

[BP80] K. Balasubramanian and K.R. Parthasarathy. In search for a complete invariant of graphs. In S.B. Rao, editor, *Combinatorics and Graph Theory*, volume 885 of *Lecture Notes in Mathematics*, pages 42–59. Springer, 1980.

[BR99] B. Bollobás and O. Riordan. A Tutte polynomial for coloured graphs. *Combinatorics, Probability and Computing*, 8:45–94, 1999.

[CMR00a] B. Courcelle, J.A. Makowsky, and U. Rotics. linear time solvable optimization problems on graphs of bounded clique-width. *Theory of Computing Systems*, 33.2:125–150, 2000.

[CMR00b] B. Courcelle, J.A. Makowsky, and U. Rotics. On the fixed parameter complexity of graph enumeration problems definable in monadic second order logic. *Discrete Applied Mathematics*, xx:xx–yy, 2000.

[DF99] R.G. Downey and M.F Fellows. *Parametrized Complexity*. Springer, 1999.

[Die96] R. Diestel. *Graph Theory.* Graduate Texts in Mathematics. Springer, 1996.

[EM98] J. Ellis-Monaghan. New results for the Martin polynomial. *Journal of Combinatorial Theory, Series B*, 74:326–352, 1998.

[Eve79] Shimon Even. *Graph Algorithms.* Pitman, London, 1979.

[Far79a] E.J. Farrell. An introduction to matching polynomials. *Journal of Combinatorial Theory, Series B*, 27:75–86, 1979.

[Far79b] E.J. Farrell. On a general class of graph polynomials. *Journal of Combinatorial Theory, Series B*, 26:111–122, 1979.

[FS90] D.C. Fisher and A.E. Solow. Dependence polynomials. *Discrete Mathematics*, 82:251–258, 1990.

[GH83] I. Gutman and F. Harary. Generalizations of the matching polynomial. *Utilitas Mathematicae*, 24:97–106, 1983.

[GM99] M. Grohe and J. Mariño. Definability and descriptive complexity on databases of bounded tree–width. In C. Beeri and P. Bunemann, editors, *Database Theory–ICDT'99*, volume 1540 of *Lecture Notes in Computer Science*, pages 70–82. Springer, 1999.

[God81] C.D. Godsil. On the theory of the matching polynomial. *Journal of Graph Theory*, 5:137–144, 1981.

[God93] C.D. Godsil. *Algebraic Combinatorics.* Chapman and Hall, 1993.

[HL94] C. Hoede and X. Li. Clique polynomials and independent set polynomials of graphs. *Discrete Mathematics*, 125:219–228, 1994.

[LP86] L. Lovasz and M. Plummer. *Matching Theory.* North Holland, 1986.

[Mak00] J.A. Makowsky. Colored Tutte polynomials and Kauffman brackets for graphs of bounded tree-width. *submitted to Combinatorics, Probability and Computation*, xxx:xx–yy, 2000.

[Mak01] J.A. Makowsky. Colored Tutte polynomials and Kauffman brackets on graphs of bounded tree width. In XX, editor, *Proceedings of the 12th Symposium on Discrete Algorithms*, volume yy, pages xx–yy. SIAM, 2001.

[Nob97] S.D. Noble. *Complexity of graph polynomials.* PhD thesis, New College, Oxford University, England, 1997.

[Nob98] S.D. Noble. Evaluating the Tutte polynomial for graphs of bounded tree-width. *Combinatorics, Probability and Computing*, 7:307–321, 1998.

[Par99] K.R. Parthasarathy. Graph characterizing polynomials. *Discrete Mathematics*, 206:171–178, 1999.

[Rio58] J. Riordan. *An Introduction to Combinatorial Analysis.* Wiley, 1958.

[Sac64] H. Sachs. Beziehungen zwischen den in einem Graphen enthaltenen Kreisen und seinem charakteristischen Polynom. *Publ. Math. Debrecen.*, 11:119–134, 1964.

[Sta86] R.P. Stanley. *Enumerative Combinatorics*, volume 1. Wadsworth and Brooks, 1986.

[Val79] L.G. Valiant. The complexity of computing the permanent. *Theoretical Computer Science*, 8:189–201, 1979.

[Ver83] M. Las Vergnas. Le polynôme de Martin d'un graphe eulerien. *Annals of Discrete Mathematics*, 17:397–411, 1983.

[Wel93] D.J.A. Welsh. *Complexity: Knots, Colourings and Counting*, volume 186 of *London Mathematical Society Lecture Notes Series.* Cambridge University Press, 1993.

[Wil90] H.S. Wilf. *generatingfunctionology.* Academic Press, 1990.

Department of Computer Science, Technion–Israel Institute of Technology, Haifa, Israel
E-mail address: {janos,julian}@cs.technion.ac.il