# From a Zoo to a Zoology: Towards a General Theory of Graph Polynomials

J.A. Makowsky*

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

**Abstract.** We outline a general theory of graph polynomials which covers all the examples we found in the vast literature, in particular, the chromatic polynomial, various generalizations of the Tutte polynomial, matching polynomials, interlace polynomials, and the cover polynomial of digraphs. We introduce two classes of (hyper)graph polynomials definable in second order logic, and outline a research program for their classification in terms of definability and complexity considerations, and various notions of reducibilities.

## 1 Introduction

In this paper I survey a research program of how a general theory of graph polynomials is being developed. The paper contains no full proofs. Its purpose is to bring the general picture to the attention of other researchers who work on related problems, and to serve as a reference point. I report here on work with various collaborators, some of it previously published and some of it in progress. The collaborators are mainly M. Bläser, B. Courcelle, B. Zilber and our respective graduate students. This is an expanded version of the author's [56].

During the last ten years I have studied questions of computability of graph polynomials, summarized in [53, 55, 51, 59, 57, 58, 54, 11]. I found uncharted territory with plenty of amazing theorems, surprising results, and the more I got into it, the more I was perplexed. I feel that we do not have a comprehensive understanding of graph polynomials, although about particular polynomials, such as the characteristic polynomial, the chromatic polynomial, the matching polynomials and the Tutte polynomial there is more known than what could be told in several books. It is noteworthy that many authors speak in their papers of *the* graph polynomial, suggesting that theirs is the one and only one worth studying. It is also noteworthy, that very few authors who study a particular graph polynomial $P$, have more than this particular polynomial and possibly some immediate relatives of $P$, in mind.

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

The collection of graph polynomials I have gathered from the literature looks like a zoo[1]. There are prominent animals like the elephant, the giraffe, the gorilla, and there are exotic animals defying classification, like the lamprey (petromyzon marinus, not really a fish) or platypus (ornithorhynchus anatinus, not really a water bird, not really a mammal). Some animals look different, but are related, like the elephant and the rock hyrax (procavia capensis); some look alike, but are not related, like the hedgehog (erinaceus europus) and the echidna (tachyglossus aculeatus). *Zoology* is the science of comparing and classifying animals. *Graph-polynomology* would be the art of comparing and classifying graph polynomials.

## 2 Graph polynomials

Let $\mathcal{G}$ be the class of graphs $G = (V, E)$ without loops and multiple edges. Let $\mathcal{R}$ be a ring and $\bar{X}$ be a (not necessarily finite) set of indeterminates. A *graph polynomial* is a function

$$p : \mathcal{G} \to \mathcal{R}[\bar{X}]$$

such that for isomorphic graphs $G_1 \simeq G_2$ we have $p(G_1) = p(G_2)$. If we consider labeled graphs, the notion of isomorphism has to be correspondingly modified. If $p(G)$ takes only values 0 or 1 in $\mathcal{R}$ we speak of *graph properties*. A graph polynomial is, properly speaking, a family of polynomials. In practice, all the graph polynomials in the literature are *uniformly defined families of polynomials*. We shall see that two kinds of uniform definitions will play a key rôle: *recursive or dynamic definitions* and *static definitions*.

There are plenty of graph polynomials which have been discussed in the literature, although no systematic treatment on graph polynomials in general is available[2]. To put our results into perspective we discuss briefly four classical graph polynomials, the *chromatic polynomial* $\chi(G, \lambda)$, the *characteristic polynomial* $P(G, \lambda)$, the *acyclic generating matching polynomials* $m(G, \lambda)$ *and* $g(G, \lambda)$ and the *Tutte polynomial* $T(G, X, Y)$. For historic reasons we also discuss briefly the very first polynomial introduced into graph theory, the *edge difference polynomial*. We also add to our discussion two more recent examples, the two *interlace polynomials*, and the *cover polynomial* defined on digraphs.

**The edge-difference polynomial.** The historically first polynomial in graph theory was introduced by J.J. Sylvester in 1878, [74] and further studied by J. Petersen in [68]. It is a multivariate polynomial depending on the ordering of the vertices $V = \{v_1, v_2, \ldots, v_n\}$ and defined as

$$P_G(X_1, X_2, \ldots, X_n) = \sum_{\substack{i<j \\ (v_i, v_j) \in E}} (X_i - X_j)$$

[1] It was T. Zaslavsky who suggested the title "From a zoo to a zoology" for this research program.

[2] I have found over 250 entries in MathSciNet querying "graph polynomial" or "polynomial of a graph" in the review text.

This polynomial is not a graph invariant, but it was used as a tool in studying regularity and colorability questions of graphs. In particular, N. Alon and M. Tarsi [3] observed that it can be used to study list colorings. For a survey, cf. Z. Tuza [79]. In our context, however, the edge-difference polynomial does not play a prominent rôle.

**The chromatic polynomial** . Let $\chi(G,\lambda)$ denote the number of proper vertex colorings of $G$ with at most $\lambda$ colors. G. Birkhoff, [9], observed in 1912 that $\chi(G,\lambda)$ is, for a fixed graph $G$, a polynomial in $\lambda$, which is now called the *chromatic polynomial of* $G$. The chromatic polynomial is the oldest graph polynomial to appear in the literature, that is a graph invariant. Since then a substantial body of knowledge about the chromatic polynomial of graphs and its applications has been accumulated. The recent book by F.M. Dong, K.M. Koh and K.L. Teo [28] gives an excellent and extensive survey. One of the surprising facts is a theorem of R.P. Stanley, [72], which states that $\chi(G,-1)$ is the number of acyclic orientations of $G$.

**The Tutte polynomial.** Interesting generalizations of the chromatic polynomial were introduced by H. Whitney in 1932 and W.T. Tutte in 1947. The most prominent among them is now called the *Tutte polynomial* $T(G,X,Y)$ which is a two variable polynomial from which the chromatic polynomial can be obtained via a simple substitution and multiplication with a pre-factor. The exact relationship is given by

$$\chi(G,X) = (-1)^{r(G)} X^{k(G)} T(G;1-X,0)$$

Here $k(K)$ is the number of connected components of $G$ and $r(G) = \mid V \mid - k(G)$.

For a modern exposition the reader is referred to [14, chapter X], [38] or [84]. Tutte's own account on how he got involved with his polynomial is very enjoyable, [78]. For this exposition we do not need a full definition of the Tutte polynomial. We only note that it is a polynomial in two variables related to the rank generating function of matroids. But we should note that in the years after 1980 the Tutte polynomial found important interpretations in statistical mechanics and quantum field theory, knot theory, and biology, cf. [84] and [71]. F. Jaeger in [44] showed that the Jones polynomial of knot theory on alternating knots is just an instance of the Tutte polynomial of the knot diagram viewed as a graph. L. Kauffman in [47] was the first to introduce a generalization of the Tutte polynomial which gives the Jones polynomial for arbitrary knots. Different approaches to multivariate versions of the Tutte polynomial are discussed in [15, 71].

Other univariate graph polynomials were introduced after 1955, often first motivated by problems from chemistry and physics. The two most prominent are the characteristic and the matching polynomial (which comes in two versions).

**The characteristic polynomial of a graph** $G$**,** denoted by $P(G,\lambda)$ is the characteristic polynomial of the adjacency matrix $M_G$ of the graph $G$, $P(G,\lambda) =$

$\det(\lambda \cdot \mathbf{1} - M_G)$ and is completely determined by the eigenvalues of $M_G$, which are all real, as the matrix is symmetric. An excellent survey is [26].

**The matching polynomials.** The *acyclic polynomial of* $G$ is the polynomial $m(G,\lambda) = \sum_k (-1)^k \cdot m_k(G) \cdot \lambda^{n-2k}$, where the coefficients $m_k(G)$ count $k$-matchings. A chemical point of view of these polynomials is given in [26], [8] and [77], where algorithmic aspects are also touched. A close relative of the acyclic polynomial is the *matching generating polynomial of a graph* $G$ defined as $g(G,\lambda) = \sum_k m_k(G)\lambda^k$. The two matching polynomials are related by the equation

$$m(G,\lambda) = \lambda^n g(G,(-\lambda^{-2})).$$

An excellent survey on these two matching polynomials may be found in [39, Chapter 1] and [52, Chapter 8.5]. We shall refer to both as *matching polynomials*. Somewhat surprisingly we have $m(G,\lambda) = P(G,\lambda)$ if and only if $G$ is a forest.

**The interlace polynomials.** Two of the more interesting recent graph polynomials were introduced by R. Arratia, B. Bollobás and G. Sorkin in [5, 6]. They are called *interlace polynomials* and there is a univariate and a two-variable version. M. Aigner and H. van der Holst [2] studied these polynomials from a matrix point of view and derived various combinatorial interpretations. B. Courcelle [20] discusses the interlace polynomial in the framework outlined in this paper.

**The cover polynomial of directed graphs.** An interesting recent graph polynomial on directed graphs is the *cover polynomial* introduced by F.R.K. Chung and R.L. Graham [19], and independently in the context of rook polynomials, by I. Gessel, [37]. In [19] it is presented as an attempt to create a Tutte-like polynomial for directed graphs, and is closely related to the chromatic polynomial. There is also related work by R.P. Stanley [73] and T. Chow [18], and very recently, by P. Pitteloud [69].

**A zoo of graph polynomials** Without giving all the necessary references, we list a few of the many graph polynomials we found in the literature. There are variations of matching polynomials, like the rook polynomials, cf. [70]. There are polynomials counting the number of (induced) subgraphs of a certain kind. Let $\mathcal{H}$ be a graph property and put $ind_{\mathcal{H}}(G,k)$ be number of induced subgraphs of size $k$ having property $\mathcal{H}$ in a given graph $G$. Then we can look at the polynomial

$$gen_{\mathcal{H}}(G,\lambda) = \sum_k ind_{\mathcal{H}}(G,k)\lambda^k$$

For $\mathcal{H}$ consisting of all the $K_n$'s (cliques), $E_n$'s (isolated points), $C_n$'s (cycles), $P_n$'s (paths) the corresponding polynomials have been studied. Instead of graph properties one can also use subsets of graphs with desirable properties such as vertex covers, coverings with subgraphs of special type etc. Some of these have been studied in a very general context as Farrell polynomials, cf. [34, 57]. There are Farrell polynomials [34], clique and independent set polynomials [42], dependence polynomials [35], Martin polynomials [32], Penrose polynomials [1], Go-polynomials [33], and many more. It is worth searching for all these at `scholar.google.com`.

## 3 Recursive definitions

One of the outstanding features of the more prominent graph polynomials are recursive definitions with respect to some order independent way of deconstructing the input graph. The main paradigm stems from the chromatic polynomial and the Tutte polynomial.

If $G = (V, E)$ is a graph and $e \in E$ is an edge with end points $u$ and $v$, we denote by $G - e$ the graph obtained from $G$ by deleting the edge, and leaving all the vertices in place. We denote by $G/e$ the graph obtained from $G$ by omitting the edge $e$ and identifying the vertices $u$ and $v$. The vertex obtained from identifying $u$ and $v$ inherits all the other edges. We first note that for the chromatic polynomial we have

$$\chi(G) = \chi(G - e) - \chi(G/e)$$

where $e$ is an edge and $G - e$ and $G/e$ denotes the deletion respectively contraction of the edge $e$. Furthermore, for the disjoint union we have $\chi(G_1 \sqcup G_2) = \chi(G_1) \cdot \chi(G_2)$, for the graph consisting of $n$ isolated vertices $E_n$, $\chi(E_n) = \lambda^n$. One easily verifies that $\chi(G - e - f) = \chi(G - f - e)$, $\chi(G/e/f) = \chi(G/f/e)$, $\chi(G - e/f) = \chi(G/f - e)$ and $\chi(G/e - f) = \chi(G - f/e)$, which is a kind of *Church-Rosser property* or *confluence property*, cf. [27]. The Church-Rosser property guarantees that the recursive definition is *indeed* well-defined. Hence we get a recursive definition of $\chi(G)$ by choosing any order of the edges, and the result is independent of the ordering of the edges. Similarly, for the Tutte polynomial we have

$$T(G, X, Y) = \begin{cases} X \cdot T(G/e, X, Y) & \text{if } e \text{ is a bridge} \\ Y \cdot T(G - e, X, Y) & \text{if } e \text{ is a loop} \\ T(G/e, X, Y) + T(G - e, X, Y) & \text{else} \end{cases}$$

together with multiplicativity for disjoint unions and $T(E_n, X, Y) = 1$. Again one can verify the Church-Rosser property, and, therefore, gets a recursive definition for the Tutte polynomial. In [15] this recursive definition was used as the starting point for the definition of the *colored Tutte polynomial.*

In [5, 6] similar but more complicated recursive definitions are given for the various interlace polynomials. Here the recursion also involves a *pivot operation* $G^{ab}$ on a graph $G$ and two vertices $a, b$. In [19] such recursive definitions are given for the cover polynomial of directed graphs. B. Courcelle gave analyzed recursive definitions in depth for the interlace polynomials in [20]. Even for the matching polynomial one can give such rules: for the acyclic polynomial we have

$m(E_n) = \lambda^n$, multiplicativity for disjoint unions and, for an edge $e = (u, v)$

$$m(G, \lambda) = m(G - e, \lambda) - m(G - u - v, \lambda)$$

Here $G - u - v$ is the graph obtained by deleting the vertices $u$ and $v$ and all the edges connected to at least one of them. The matching polynomial was studied from this point of view in [7]. It is a curious fact that the previous literature does not explore this aspect of the matching polynomial further, and does not even mention the Church-Rosser property, although it is easily verified.

One advantage of a recursive definition of a graph polynomial is that it gives an easy but slow way of computing the graph polynomial. As the recursion unwinds a number of subtasks exponential in the size of the graph has to be computed. But the nature of the recursion usually gives deeper insights. Furthermore, the various Tutte and interlace polynomials can be proven to be, in a certain sense, the *most general graph polynomials satisfying their specific recursion scheme*. Similar characterizations were very recently shown for generalizations of the cover and the matching polynomials in [23, 7].

It is remarkable fact that all researchers so far have taken such recursive definitions as their departure point or main focus in studying new graph polynomials. Although some particular recursion schemes based on the behaviour of the graph polynomial under deletion of vertices or edges, contractions of edges, pivoting, etc. are well studied, no general theory has emerged so far, and it remains an interesting challenge to develop a satisfactory framework for recursion schemes for graph invariants. In the forthcoming paper with B. Courcelle and B. Godlin, [22], an attempt in creating such a framework is presented. It is based on ideas reminiscent of graph grammars. We shall sketch its main ideas in section 5, after we have introduced the logical aspects of graph polynomials. However, we shall see in the sequel that this point of view is too restrictive, even if it usually gives a very elegant treatment of the polynomials in question.

## 4 Complexity

*Easy and hard cases.* *Easy to compute* here means, as usual, polynomial time computable. However, *hard to compute* is best captured with the complexity class $\sharp\mathbf{P}$, which allows to count the number of successful guesses in a non-deterministic polynomial time algorithm. This class was introduced by L. Valiant in [82], and has a rich literature. Typical problems complete for $\sharp\mathbf{P}$ via polynomial time reductions are: counting the number of satisfying assignments of $SAT$, the satisfiability of propositional logic; counting the number of perfect matchings of a graph; and counting the number of $k$-colorings of a graph for all $k \geq 3$. Problems in $\sharp\mathbf{P}$ are likely to be much harder than problems in $\mathbf{NP}$. S. Toda [76] showed that every problem in the polynomial hierarchy is polynomial time Turing reducible to complete problems in $\sharp\mathbf{P}$. The reader may want to consult [67] or [46].

As in all the examples, the graph is given by its adjacency matrix, the coefficients of the graph polynomials are integers. Therefore, what we want to compute, given a graph $G$, are all the coefficients of the graph polynomial in question. It is natural to ask how difficult it is to compute the various graph polynomials, i.e. all its integer coefficients, on restricted graph classes. Clearly, this is a more difficult problem, than just evaluating the graph polynomials on a specific point. However, if we can evaluate the graph polynomial for a fixed graph at sufficiently many points on, say a grid, we can use Lagrange interpolation to compute its coefficients.

The characteristic polynomial is computable in polynomial time using classical algorithms for the determinant of a matrix. Computing the chromatic polynomial is $\sharp\mathbf{P}$-hard due to its connection to counting colorings, cf. [82]. This also makes computing the Tutte polynomial $\sharp\mathbf{P}$-hard. The same is true for the acyclic polynomial due to its connection to counting matchings, cf. [82].

J. Oxley and D. Welsh [66] also noted that the Tutte polynomial for series–parallel graphs, which are graphs of tree-width at most 2, can be computed in polynomial time. This was extended to arbitrary fixed tree-width $k$ independently by A. Andrejak [4] and S. Noble [63], and therefore also holds for the chromatic polynomial. Actually, they showed that computing the Tutte polynomial is fixed parameter tractable **FPT** on graph classes of tree-width at most $k$. In other words, it is computable in time $f(k)n^d$, where $d$ is independent of $k$ and $n$ is the size of the input. For an extensive discussion of the complexity class **FPT**, cf. [29].

This result was extended to the matching and the acyclic polynomial, the interlace polynomials, the multivariate Tutte polynomials, and the cover polynomial, in [54], based on [24]. We shall return to these results in Section 5. We note that we define here the tree-width of a directed graph as the tree-width of its underlying undirected graph. All these results are well defined and meaningful in the Turing model of computation.

*Hard points for evaluation.* In a fundamental paper, Jaeger, Vertigan and Welsh [45], cf. also [84], studied the complexity of computing the Tutte polynomial $T(G, X, Y) \in \mathbb{C}[X, Y]$, as a polynomial over the complex numbers. They looked at the complexity of evaluating $T(G, X, Y)$, for fixed $X = X_0, Y = Y_0$, and letting the graphs vary. They were able to give a complete classification of the set of points $(X_0, Y_0) \in \mathbb{C}^2$ according to the complexity of evaluating $T(G, X_0, Y_0)$. Due to the nature of the problem, complexity had to be measured in a hybrid way, involving both Turing complexity and algebraic complexity. For non-negative integer values $(X_0, Y_0) \in \mathbb{N}^2$, evaluating $T(G, X_0, Y_0)$ is a discrete problem in the class $\sharp\mathbf{P}$. The Turing model of computation also applies for the case when $(X_0, Y_0)$ are rational points. But $\sharp\mathbf{P}$ contains only functions in $\mathbb{N}^{\mathbb{N}}$. Therefore, complexity is expressed via *algebraic* reducibility to a problem in $\sharp\mathbf{P}$. The paper [45] does not elaborate precisely the model of computation needed for the most general version of their result.. The authors seem to use the Turing model, and use algebraic extensions of the field of rational numbers as their base field. For the reducibility they use a notion of algebraic reducibility, where arithmetic operations in $\mathbb{C}$ and access to constants in $\mathbb{C}$ are measured in unit cost[3]. The framework of Blum, Shub and Smale [12] is well suited for this. They work in an arbitrary ring (or field) $\mathcal{R}$ and use register machines where the registers can store arbitrary elements of $\mathcal{R}$ at unit cost. The machine can perform the arithmetic operations also at unit cost, has a mechanism of indirect addressing[4], and can test equality of elements of $\mathcal{R}$. The authors of [45] seemingly were not aware of [13], where this model of computation was first introduced.

We denote by $\mathbf{FP}_{\mathcal{R}}$ and $\mathbf{FEXP}_{\mathcal{R}}$ the complexity classes of functions computable in polynomial respectively exponential time in the BSS-model over an arbitrary ring (or field) $\mathcal{R}$. Usually, $\mathcal{R}$ is the field $\mathbb{R}$ of real numbers or the field $\mathbb{C}$ of complex numbers. We denote by $\mathbf{FP}_{\mathbb{C}}^{F}$ the class of functions computable in polynomial time using oracles from $F$. We say a function $f$ is $\mathbf{AlgP}_{\mathbb{C}}$-reducible to a function $g$ if $f$ is in $\mathbf{FP}_{\mathbb{C}}^{\{g\}}$. Graphs can be coded in a standard way, so the numeric graph invariant $T(-, X_0, Y_0)$ is in $\mathbf{FEXP}_{\mathbb{C}}$. There is no suitable analogue of $\sharp\mathbf{P}$ in the BSS-model which fits our situation. A class $\sharp\mathbf{P}_{\mathbb{C}}$ was defined in [62], which serves a different purpose, as it counts the number of successful guesses in $\mathbb{C}$, not discrete guesses. On the other hand, the functions $f \in \mathbb{N}^{\mathbb{N}}$ which are in $\sharp\mathbf{P}$ of the Turing model are a subclass of the functions in $\mathbb{C}^{\mathbb{C}}$, but they are not computable in the BSS model, because recognizing the integers in $\mathbb{C}$ is not computable in the BSS model[5].

Having this in mind, the main result of [45] can be formulated as follows:

**Theorem 1 (Jaeger, Vertigan and Welsh).** *The complexity of computing the numeric graph invariants $T(-, X_0, Y_0)$ of the Tutte polynomial is described as follows: There is a set $B \subseteq \mathbb{C}^2$ such that*

*(i) For all $(X_0, Y_0) \in B$ the numeric graph invariant $T(-, X_0, Y_0) \in \mathbf{FP}_{\mathbb{C}}$.*
*(ii) For all $(X_0, Y_0), (X_1, Y_1) \notin B$ we have that $T(-, X_0, Y_0)$ is $\mathbf{AlgP}_{\mathbb{C}}$-reducible to $T(-, X_1, Y_1)$.*
*(iii) For some $(X_0, Y_0) \in \mathbb{N}^2 - B$ we have that $T(-, X_0, Y_0) \in \sharp\mathbf{P}$ and it is $\sharp\mathbf{P}$-hard, hence $\sharp\mathbf{P}$-complete.*
*(iv) Furthermore, $B$ is a finite union of algebraic sets in $\mathbb{C}^2$ of dimension $\leq 1$.*

*In fact, in [45] an explicit description of $B$ is given.*

In the case of the univariate chromatic polynomial, the theorem was first proved in [50] as

[3] Actually, in [45], the authors restrict their discussion to fields which are finite dimensional algebraic extensions of the rationals and use some polynomial time encoding of the arithmetic operations in the Turing model of computation.
[4] Indirect addressing poses a problem, because the addresses are natural numbers, but the contents of the registers are elements of an arbitrary but fixed ring $\mathcal{R}$.
[5] P. Bürgisser in [16, 17] also discusses the complexity of certain weighted graph polynomials. The framework he uses is completely algebraic and was first introduced by L. Valiant in [80, 82].

**Theorem 2 (Linial).** *For every $X_0, X_1 \in \mathbb{C} - \{0, 1, 2\}$, evaluating the chromatic polynomial $\chi(-, X_0)$ is $\mathbf{AlgP}_{\mathbb{C}}$-reducible to evaluating $\chi(-, X_1)$. As for $X_0 = k, k \in I\!N, k \geq 3$ it is $\sharp\mathbf{P}$-complete, we get that for every $X_0 \in \mathbb{C} - \{0, 1, 2\}$, evaluating the chromatic polynomial $\chi(-, X_0)$ is $\sharp\mathbf{P}$-complete.*

For various versions of the Tutte polynomial, a similar theorem is true, cf. Makowsky and Bläser [11]. For univariate polynomials, the situation is not obvious. Using results from Dyer and Greenhill [30], Averbouch and Makowsky [7] have shown

**Theorem 3 (Averbouch, Makowsky).** *For every $X_0, X_1 \in \mathbb{C} - \{0\}$, evaluating the matching polynomial $m(-, X_0)$ is $\mathbf{AlgP}_{\mathbb{C}}$-reducible to evaluating $m(-, X_1)$ and is $\sharp\mathbf{P}$-hard.*

For the cover polynomial, Bläser and Dell [10] have very recently shown:

**Theorem 4 (Bläser, Dell).** *For every two points $(X_0, Y_0), (X_1, Y_1) \in \mathbb{R}^2 - B$, evaluating the cover polynomial $cov(-, X_0, Y_0)$ is $\mathbf{AlgP}_{\mathbb{R}}$-reducible to evaluating $cov(-, X_1, Y_1)$ and is $\sharp\mathbf{P}$-hard. The exception set $B$ is given by $Y = 1$ or $X = Y = 0$.*

The techniques used to prove Theorem 3 and 4 each differ from the scenarios given in [45]. [7] is based on ideas from [30], whereas [10] uses a reduction different from the ones used [45].

It seems to be a non-trivial challenge to prove similar theorems for other graph polynomials, such as the interlace polynomials. A generalization of Theorems 1-4 is discussed in section 5.

## 5 Enter logic

**Background.** Let is define some basics for the reader less familiar with second order logic. A *vocabulary* $\tau$ is a set of constant, function and relation symbols. A *one-sorted $\tau$-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). $\tau$-terms are formed using individual variables, constant symbols and function symbols from $\tau$. 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. An excellent reference for our logical background is [31].

B. Courcelle in a series of papers has explored the usefulness of monadic second order logic **MSOL** for graph algorithms and graph theory in general, cf.

[21]. Monadic second order logic $\mathbf{MSOL}(\tau)$ is the restriction of $\mathbf{SOL}(\tau)$ to unary relation variables and quantification over these. B. Courcelle in [25] observed that graph properties definable in Monadic Second Order Logic (**MSOL**) are fixed parameter tractable, i.e. in **FPT**, on graph classes of tree-width at most $k$, cf. also [29, 36]. This approach was extended to graph polynomials by the author in [54]. The fact that the Tutte polynomial is in **FPT** also follows from [54], which also covers the acyclic and the matching polynomial and a wide range of other graph polynomials where summations are restricted to families of subsets of edges which are definable in **MSOL**.

**SOL-polynomials.** To understand better what all the graph polynomials have in common we have to look closer at the way they are defined. Besides their recursive definition 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 relation variable, or in monadic second order logic **MSOL**, where $M$ is unary set variable ranging over subsets of edges.

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

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

where $A$ is a unary relation variable, $\phi(A)$ is an **MSOL**-definable property of the graph, and $t(v)$ is a term in $\mathcal{R}[\bar{X}]$ which may depend uniformly on $v$. One can give an inductive definition of **SOL**-polynomials by defining **SOL**-monomials as being of the form $\prod_{v:v \in A} t(v)$ and allowing closure under addition and multiplication and under summations of the form $\sum_{A:\phi(A)} t(A)$.

We speak of **MSOL**-polynomials, if we allow binary relation variables (and relation variables of higher arity) we speak of **SOL**-polynomials. A detailed exposition may be found in [54].

In the same paper [54], it is shown that, in combination with the work of P. Seymour and S. Oum [65], graph polynomials, where summations are restricted to families of subsets of vertices which are **MSOL**-definable, are in **FPT** for graph classes of clique-width at most $k$. However, this method does not apply to the chromatic polynomial, the Tutte polynomial and the matching polynomials. As discussion of the computational complexity of these polynomials for the class of graphs of clique-width at most $k$ may be found in [60].

In [61] the class of *extended SOL-polynomials* is introduced. In the extended case the basic combinatorial polynomials are also included. More precisely, for every $\phi(\bar{v}) \in \mathbf{SOL}(\tau)$ we define the cardinality of the set defined by $\phi$:

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

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

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

are **SOL**-definable $\mathcal{M}$-monomials.

All the results of [54], stated for **MSOL**-polynomials, are also valid for extended for **MSOL**-polynomials.

**Recursion revisited.** The logic **SOL** can also be used to define a framework in which the existence of a recursive definition of a graph polynomial can be formulated. This program is carried out in [22]. The formalism is a bit heavy in its full generality. What one needs is a finite set of graph operations $T_1, T_2, \ldots, T_s$ definable in **SOL**, which allow us to deconstruct a graph $G$ into graphs $T_i(G) : i \leq s$ in such a way, that the graph polynomial $p(G)$ is a linear combination (in the polynomial ring) of the graph polynomials $p(T_i(G))$. Furthermore, the application of these operations must satisfy a confluence (Church-Rosser) property. Recursive definitions are a kind of *dynamic definition* of graph polynomials, in contrast to the *static definitions* as exemplified by by the **SOL**-polynomials.

In [22]. we prove

**Theorem 5 (Courcelle, Godlin, Makowsky).** *Assume that a graph polynomial $p(G)$ has a recursive definition with* **SOL***-definable graph transformations $T_i : i \leq r$ which satisfy the confluence property. Then $p(G)$ is an* **SOL***-definable graph polynomial.*

It is not clear, whether, when restricting to **MSOL**-definable graph transformations, one gets an **MSOL**-definable polynomial.

Although it is not difficult to define artificially a graph polynomial which is not **SOL**-definable, we have verified that all the graph polynomials studied in the literature are **SOL**-definable. Actually, they are all **CMSOL**-definable, i.e. definable in **MSOL** enriched with a modular counting quantifier, or, alternatively, definable in fixed point logic **FPL** over graphs with a linear ordering on its vertices. For a reference and discussion of these logics cf. [64, 31, 49].

**Complexity revisited.** Returning to complexity issues, here is a challenging conjecture:

**Conjecture 1 (Difficult Point Conjecture)** *Let $p(G, \bar{X})$ be an extended* **SOL***-definable graph polynomial in $r$ many indeterminates. The complexity of computing the numeric graph invariants $p(-, \bar{X})$ of the polynomial $p(G, \bar{X})$ is described as follows: There is a set $B \subseteq \mathbb{C}^r$ such that*

*(i) For all $\bar{X}_0 \in B$ $T(-, \bar{X}_0) \in \mathbf{FP}_{\mathbb{C}}$.*
*(ii) For all $(\bar{X}_0), (\bar{X}_1) \notin B$ $T(-, \bar{X}_0)$ is $\mathbf{AlgP}_{\mathbb{C}}$-reducible to $T(-, \bar{X}_1)$.*
*(iii) $B$ is a finite union of algebraic sets in $\mathbb{C}^r$ of dimension strictly less than $r$.*

*We can also formulate the conjecture for fragments of* **SOL***, such as* **MSOL***, or* **FPL** *on ordered graphs.*

Theorems 1-4 form the basis of this conjecture. Using suitable notions of reducibilities between graph polynomials, one can produce infinitely many instances confirming the conjecture. A simple case of such a reducibility is discussed in [7].

## 6 Generalized chromatic polynomials

It turns out that extended **SOL**-definable graph polynomials turn up naturally also in another approach to graph polynomials developed in [61].

A two-sorted vocabulary is a set of constant, function and relation symbols where the arguments of the functions and relations are typed, i.e. belong to one of the two sorts. A two-sorted structure is an interpretation of a two-sorted vocabulary over two sets (universes) where each universe is the interpretation of one sort. For a natural number $k \in \mathbb{N}$, we denote by $[k]$ the set $\{0, 1, \ldots, k-1\}$. In the sequel we will use sets of the form $[k]$ as an additional sort. We also use $\tau$-structures rather than graphs. The reader less familiar with model-theoretic reasoning may still think of graphs. Only in the proofs in Section 7 the full generality of $\tau$-structures is needed.

Let $\mathcal{M}$ be a $\tau$-structure with universe $M$, and denote by $\mathcal{M}_k$ the two-sorted structure $\langle \mathcal{M}, [k] \rangle$ for the modified vocabulary $\tau_1$, where all the arguments of the function and relation symbols of $\tau$ are of the first sort. Let $\phi(R)$ be a second order $\tau_1 \cup \{R\}$-formula, where $R$ is an $(m+r)$-ary relation symbol and the $r$ last arguments of $R$ are of the sort of $[k]$. A relation $R_M \subset M^m \times [k]^r$ is a *generalised $k - \phi$-coloring* if

(i) $\langle \mathcal{M}_k, R \rangle \models \phi(R)$, and
(ii) If $\langle \mathcal{M}_k, R \rangle \models \phi(R)$, and $k_1 \geq k$, then If $\langle \mathcal{M}_{k_1}, R \rangle \models \phi(R)$.
(iii) If for every $\mathcal{M}$ there is a number $N_M$ such that for all $k \in \mathbb{N}$ the set of colors
$$\{x \in [k] : \exists \bar{y} \in M^m R(\bar{y}, x)\}$$
has size at most $N_M$.

We denote by $\chi_{\phi(R)}(\mathcal{M}, k)$ the number of generalised $k - \phi$-coloring $R$ on $\mathcal{M}$.

**Theorem 6 (Makowsky, Zilber).** *For every $\mathcal{M}$ the number $\chi_{\phi(R)}(\mathcal{M}, k)$ is a polynomial in $k$ of the form*
$$\sum_{j=0}^{N_M} c_{\phi(R)}(\mathcal{M}, j) \binom{k}{j}$$
*where $c_{\phi(R)}(\mathcal{M}, j)$ is the number of generalised $k - \phi$-colorings $R$ with a fixed set of $j$ colors.*

**Proof:** We first observe that any generalised coloring $R$ uses at most $N_M$ of the $k$ colors. For any $m \leq N$, let $c(j)$ be the number of colorings, with a fixed set of $j$ colors, which are generalised vertex colorings and use all $j$ of the colors. Next we observe that any permutation of the set of colors used is also a coloring. Therefore, given $k$ colors, the number of vertex colorings that use exactly $j$ of the $k$ colors is the product of $c_{\phi(R)}(\mathcal{M}, j)$ and the binomial coefficient $\binom{k}{j}$. So

$$\chi_{\phi(R)}(\mathcal{M}, k) = \sum_{j \leq N_M} c_{\phi(R)}(\mathcal{M}, j) \binom{k}{j}$$

The right side here is a polynomial in $k$, because each of the binomial coefficients is. We also use that for $k \leq j$ we have $\binom{k}{j} = 0$. $\square$

In the light of this theorem we call $\chi_{\phi(R)}(\mathcal{M}, k)$ a *generalised chromatic polynomial.*

We extend the definition to construct also graph polynomials in several variables. Let $\mathcal{M}$ be a $\tau$-structure with universe $M$, and denote by $\mathcal{M}_{k_1,\ldots,k_\alpha}$ the $(1+\alpha)$-sorted structure $\langle \mathcal{M}, [k_1], \ldots, [k_\alpha] \rangle$ for the vocabulary $\tau_\alpha$. We put $\bar{k}^\alpha = (k_1, \ldots, k_\alpha)$.

Let $\phi(R)$ be a first order $\tau_\alpha \cup \{R\}$-formula. Recall that $A \sqcup B$ denotes the disjoint union of the sets $A$ and $B$. A relation

$$R_M \subset M^{m_1} \times ([k_1] \sqcup \ldots \sqcup [k_\alpha])^{m_2}$$

is a *generalised* $\bar{k^\alpha} - \phi$*-coloring* if

(i) $\langle \mathcal{M}_{k_1,\ldots,k_\alpha} \rangle \models \phi(R)$, and
(ii) there is a number $d \in \mathbb{N}$ such that for every $y \in M^{m_1}$ the set

$$\{\bar{x} \in ([k_1] \sqcup \ldots \sqcup [k_\alpha])^{m_2} : S(\bar{y}, \bar{x})\}$$

has at most $d$ elements.

We denote by $\chi_{\phi(R)}(\mathcal{M}, \bar{k}^\alpha)$ the number of generalised $\bar{k}^\alpha - \phi$-coloring $R$ on $\mathcal{M}$.

**Theorem 7 (Makowsky, Zilber).** *For every $\mathcal{M}$ the number $\chi_{\phi(R)}(\mathcal{M}, \bar{k}^\alpha)$ is a polynomial in $\bar{k}^\alpha$ of the form*

$$\sum_{j=0}^{d \cdot |M|^m} c_{\phi(R)}(\mathcal{M}, \bar{j}^\alpha) \prod_{1 \leq \beta \leq \alpha} \binom{k_\beta}{j_\beta}$$

*where $c_{\phi(R)}(\mathcal{M}, \bar{j}^\alpha)$ is the number of generalised $\bar{k}^\alpha - \phi$-colorings $R$ with a fixed sets of $j_\beta$ colors respectively.*

**Proof:** Similar to the one variable case. $\square$

Let $\mathcal{M}$ be a $\tau$-structure and $\mathcal{M}_k$ as before. Assume we have a formula $\phi(f_1, \ldots, f_M)$ with $M$ function variables for generalised colorings which specifies the functions simultaneously. If we fix the interpretation of the first $M-1$

function variables and denote these by $F_1, \ldots, F_{M-1}$ we have a new structure $\mathcal{N} = \langle \mathcal{M}_k, F_1, \ldots, F_{M-1} \rangle$ in which we count just one generalised coloring for each interpretation $F_1, \ldots, F_{M-1}$. The general counting is obtained by summing over all interpretations. Hence, as the sum of polynomials is a polynomial, this again gives us a polynomial. The same argument works, if we allow relations on the structure $\mathcal{M}$, which do not involve the sort $[k]$ in $\mathcal{M}_k$, and provided the range of these relations is bounded in the sense that there is a number $d \in \mathbb{N}$ such that for every $y \in M^{m_1}$ the set $\{\bar{x} \in ([k_1] \sqcup \ldots \sqcup [k_\alpha])^{m_2} : S(\bar{y}, \bar{x})\}$ has at most $d$ elements. We call polynomials obtained in this way also *generalised chromatic polynomials*.

The following will be useful.

**Proposition 1 (Sums and products).** *The sum and product of two generalised chromatic polynomials $\chi_{\phi(f)}(G, \lambda)$ and $\chi_{\psi(f)}(G, \lambda)$ is again a generalised chromatic polynomial.*

**Proof:** For the sum we take $\chi_{\theta_1}(G, \lambda)$ with

$$\theta_1(f) = ((\phi(f) \wedge \neg\psi(f)) \vee (\psi(f) \wedge \neg\phi(f)) \vee (\phi(f) \wedge \psi(f)))\,.$$

For the product we take $\chi_{\theta_2}(G, \lambda)$ where we use two distinct function symbols $f$ and $f'$ and $\theta_2(f, f') = (\phi(f) \wedge \psi(f'))$. $\square$

## 7 All SOL-polynomials are generalised chromatic polynomials

We now show how many graph polynomials can be viewed as generalised chromatic polynomials.

**Combinatorial polynomials.** The following combinatorial polynomials can be thought of as generalised chromatic polynomials:

(i) For the polynomial $\lambda^n$ we take all maps $[n] \to [k]$ for $\lambda = k$. So $\lambda^n = \chi_{true(f)}$ where $true(f)$ is $\forall v(f(v) = f(v))$.
(ii) Similarly, for $\lambda_{(n)} = \lambda \cdot (\lambda - 1) \cdot \ldots \cdot (\lambda - n + 1)$ we take all injective maps, which is easily expressed by a first order formula.
(iii) Finally, for $\binom{\lambda}{n}$ we take the ranges of injective maps. This is a coloring property of a second order formula $\phi(\mathbf{P})$ which says that $P \subseteq [k]$ is the range of an injective map $f : [n] \to [k]$.

**Connected components.** We denote by $k(G)$ the number of connected components of $G$. The polynomial $\lambda^{k(G)}$ can be written as $\chi_{\phi_{connected}}(G, \lambda)$ with $\phi_{connected}(f)$ the formula $((u, v) \in E \to f(u) = f(v))$.

**Hypergraph colorings and mixed hypergraph colorings.** A hypergraph $G$ consists of a set of vertices $V(G)$ and a family $E(V)$ of subsets of $V$, called the hyper edges. To make this into a first order structure we have two sorts of elements, the elements of $V$ and of $E$, together with the membership relation, which satisfies extensionality. A mixed hypergraph $G$ has two kinds of hyper edges, $D(G)$ and $E(G)$. Mixed hypergraph colorings come in several flavours. For a recent exhaustive survey, cf. [83]. We discuss here two cases:

**Weak mixed hypergraph colorings** A *weak mixed hypergraph coloring with $k$ colors* is a mapping $f : V \to [k]$ such that if $u, v \in d \in D(G) \to f(u) = f(v)$ and if $\forall e \in E(G) \exists u, v \in e \in E(G) \to f(u) \neq f(v)$.

**Strong mixed hypergraph colorings** A *strong mixed hypergraph coloring with $k$ colors* is a mapping $f : V \to [k]$ such that if $u, v \in d \in D(G) \to f(u) = f(v)$ and if $\forall e \in E(G) \forall u, v \in e \in E(G) \to f(u) \neq f(v)$.

We denote by $\chi_{weak}(G, k)$ and $\chi_{strong}(G, k)$ respectively the number of weak (strong) mixed hypergraph colorings with at most $k$ colors.

**Proposition 2 (V.L. Voloshin).** *$\chi_{weak}(G, k)$ and $\chi_{strong}(G, k)$ are polynomials in $k$.*

Clearly, this is also a corollary to our Theorem 6.

**Matching polynomial.** Let $G = (V, E)$ be a graph. A subset $M \subseteq E$ is a matching if no two edges in $E$ have a common vertex. The matching polynomial of $G$ is given by

$$g(G, \lambda) = \sum_j \mu(G, j)\lambda^j$$

where $\mu(G, j)$ is the number of of matchings of size $j$.

We look at the structure $G_k$ and at pairs $(M, F)$ with $M \subset E$ and $F : E \to [k]$ such that $M$ is a matching and the domain of $F$ is $M$, which can be expressed by a formula $match(M, F)$. We have

$$\chi_{match(M,F)}(G, k) = \sum_j \mu(G, j)k^j = g(G, k)$$

**Tutte polynomial.** We use the Tutte polynomial in the following form:

$$Z(G, q, v) = \sum_{A \subseteq E} q^{k(A)} v^{|A|}$$

where $k(A)$ is the number of connected components of the spanning subgraph $(V, A)$. This form of the Tutte polynomial is discussed in [71]. For this purpose we look at the 4-sorted structure

$$G_{k,l} = \langle V, [k], [l], \wp(E), E \rangle$$

and at the triples $(A, F_1, F_2)$ with $A \in \wp(E)$, $F_1 : V \times \wp(E) \rightarrow [k]$ and $F_2 : A \rightarrow [l]$ such that for $(u, v) \in A \rightarrow F_1(A, u) = F_1(A, v)$. This is expressed in the formula $tutte(A, F_1, F_2)$. Now we have

$$\chi_{tutte(A,F_1,F_2)}(G, k, l) = \sum_{A \subseteq E} k^{conn(A)} l^{|A|}$$

which is the evaluation of $Z(G, q, v)$ for $q = k, v = l$.

In our definition of generalised chromatic polynomials we have requested that the generalised coloring be specified by a formula of first order logic $\mathbf{FOL}(\tau)$. This is not necessary. The formulas of $\mathbf{SOL}(\tau)$ are defined like the ones of **FOL**, with the addition that we allow countably many variables for $n$-ary relation symbols $U_{n,\alpha}$ for $\alpha \in \mathbb{N}$, for each $n \in \mathbb{N}$, and quantification over these. A generalised chromatic polynomial is definable in $\mathbf{SOL}(\tau)$, respectively in $\mathbf{MSOL}(\tau)$ or $\mathbf{FPL}(\tau)$, if it is of the form $\chi_\phi(\mathcal{M}, \bar{\lambda})$, the counting function of a generalised coloring specified by $\phi$, with $\phi \in \mathbf{SOL}$, respectively in $\mathbf{MSOL}(\tau)$ or $\mathbf{FPL}(\tau)$.

Here is a generalization of Theorem 7.

**Theorem 8.** *Every counting function of a generalised* **SOL***-definable coloring is a polynomial, which we call also generalised chromatic polynomial.*

Also Proposition 1 remains true with the same proof. Several classes of generalised chromatic polynomials are of special interest, those which are **SOL**-definable, **MSOL**-definable, and those which are definable in fixed point logic **FPL** on ordered graphs. For details about fixed point logic, cf. [31].

For each of these we have a characterization in terms of **SOL**-definable polynomials.

**Theorem 9 (Makowsky and Zilber).** *Let $p(G, \bar{X})$ be a graph polynomial. The following statements are equivalent:*

*(i) $p(G, \bar{X})$ is an extended $\mathbf{SOL}(\tau)$-polynomial over some $\tau$-structure $\mathcal{A}$.*
*(ii) $p(G, \bar{X})$ is a is a counting function of a generalised coloring definable in $\mathbf{SOL}(\tau)$ in a suitable expansion of $\mathcal{A}$.*

*The same is true if we replace* **SOL** *by* **MSOL** *or* **FPL**.

## 8 Complexity theory for graph polynomials

The literature on Turing complexity or algebraic complexity does not provide a natural framework to develop a complexity theory of graph polynomials. In particular there is no agreed upon notion of *efficient reducibility between graph polynomials.* The existing frameworks do allow the formulation of hardness results by reductions to $\sharp\mathbf{P}$-hard instances which are easily recognizable as polynomial time computable in an intuitive sense. But in the existing frameworks no *hardest graph polynomial* could be identified.

Theorem 9 suggests the **SOL**-polynomials, and the **FPL**-polynomials as reasonable complexity classes to study graph polynomials. These classes have enough closure properties, and have different equivalent definitions. Its members can be computed in exponential time in the unit cost computational model over the underlying ring $\mathcal{R}$, in the sense of the Blum-Shub-Smale model of computation, [12]. They also reflect and generalize classical complexity theory. The **SOL**-definable graph properties correspond to graph properties in the polynomial hierarchy, and the **FPL**-definable graph properties of graphs with an ordering on the vertices correspond to the class **P**. Furthermore, every function $f : \mathbb{N} \rightarrow \mathbb{N}$ which is in $\sharp\mathbf{P}$ is the evaluation of some **FPL**-definable polynomial. As became obvious in descriptive complexity theory, the **MSOL**-definable graph properties do not form a well behaved complexity class. The same is true for graph polynomials: The **MSOL**-definable polynomials seem not to be closed under multiplication.

In our previous work we have verified that all graph polynomials of the literature are **SOL**-definable, and most of them **MSOL**-definable. But I dare to pronounce the following rather vague conjecture:

**Conjecture 2** *All naturally occurring graph polynomials are* **FPL**-*definable graph polynomials.*

This is definitely true for all the examples listed in this paper.

Having identified candidates for good complexity classes of graph polynomials is not enough. Although these classes can accommodate our large *zoo of graph polynomials*, there is still no *general zoology*. We offer here an outline of what such a zoology could look like.

The purpose of the general framework is to initiate a comparative study of the many graph, digraph and hypergraph polynomials which have appeared in the literature. For an extensive list of references cf. [54] and [57]. In particular, we address the following:

**Comparability.** Given two graph polynomials $f(G, \bar{x})$ and $g(G, \bar{x})$, we say that $g$ is weaker than $f$, and write $g \preceq f$, if for any two graphs $G_1, G_2$ with $f(G_1, \bar{x}) = f(G_2, \bar{x})$ we also have $g(G_1, \bar{x}) = g(G_2, \bar{x})$. If $g \preceq f$ and $f \preceq g$, we say the polynomials are graph-equivalent. Comparability of graph polynomials is undecidable. This follows from the undecidability of the consequence problem of First Order Logic if restricted to finite graphs, which was proven by M. Taitslin [75] and sharpened by I. Lavrov [48], cf. [41, Theorem 5.5.1]. We note that the two matching polynomials $m(G, \lambda)$ and $g(G, \lambda)$ are graph equivalent, but incomparable with respect to the characteristic polynomial $P(G, \lambda)$, and also with respect to the chromatic polynomial, and the Tutte polynomial. The study of this partial order among **SOL**-polynomials, **MSOL**-polynomials, **FPL**-polynomials, or other restricted classes of graph polynomials is a natural topic of investigation. In particular, one can ask: is there a strongest **SOL**-polynomial, what are its additional structural properties, is it a lattice, etc.

**Reducibilities.** Reducibilities have now two components:

(i) Computations in the ring, performed on the polynomial, in the uniform computational model BSS, or in the non-uniform model of L. Valiant [81, 16, 17]. Algebraic circuits (straight-line programs) or $\mathbf{P}_{\mathcal{R}}$-programs are natural choices, where $\mathcal{R}$ is the underlying ring.
(ii) Transductions of the graphs (relational structures), expressible in the logic $\mathcal{L}$ for suitably chosen $\mathcal{L}$, or computable by Turing machine transducers in the corresponding complexity class $\mathbf{C}$.

For $\mathbf{P}$-polynomials over $\mathcal{R}$, $\mathbf{P}_{\mathcal{R}}$ and $\mathbf{P}$-transductions, respectively transductions definable in Fixed Point Logic, are natural choices. For details see [43, 31, 49]. In this case we speak of $\mathbf{P}$-reducibility between two graph polynomials $f, g$ and write $g \preceq_P f$. The comparability and reducibility relations between graph polynomials do not coincide. The chromatic polynomial $\chi(G, \lambda)$ is $\mathbf{P}$-reducible to the Tutte polynomials, but it is not comparable to the Tutte polynomial. This can be easily seen from the formula

$$\chi(G, X) = (-1)^{r(G)} X^{k(G)} T(G; 1 - X, 0)$$

Recall that $k(K)$ is the number of connected components of $G$, and $r(G)$ is the rank defined by $r(G) = \mid V \mid - k(G)$. The formula shows that the chromatic polynomial is computable in polynomial time from the Tutte polynomial, but the Tutte polynomial remains invariant under the addition of isolated vertices to the graph $G$, whereas the chromatic polynomial does not.

It is open whether the matching polynomials $m(G, \lambda)$ and $g(G, \lambda)$ are $\mathbf{P}$-reducible to the Tutte polynomial.

**Reduction-complete polynomials.** A graph polynomial is *reduction complete* in a complexity class $\mathbf{C}$ equipped with a notion of reducibility, if every other $\mathbf{C}$-polynomial is reducible to it. To speak about reduction-complete polynomials it may be reasonable to fix the number of variables of the polynomials under consideration. Are there any reduction-complete $\mathbf{P}$-polynomials? Is the Tutte polynomial reduction-complete? We note that the Tutte polynomial has been shown to be the most general graph polynomial with respect to certain reduction rules (contraction and deletion of edges), cf. [14, Chapter X]. But this excludes the matching polynomial from the discussion.

### Acknowledgments

I am indebted to my colleagues, co-authors and students of my seminar on graph polynomials, with whom I could discuss, during the last ten years, various aspects of graph polynomials. Among them are Y. Altschuler, I. Averbouch, M. Bläser, P. Bürgisser, B. Courcelle, B. Dubrov, E. Fischer, B. Godlin, M. Lotz, J. Mariño, A. Matsliach, K. Meer, E. Ravve, and U. Rotics.

In particular, I would like to thank B. Courcelle for inviting me in February and September 2006 to spend time in Bordeaux working with him on graph polynomials, and to M. Bläser for inviting me in February 2007 to Saarbrücken to discuss the complexity of graph polynomials.

Certain passages of the paper are, with the explicit consent of my co-authors, literally taken from our joint work. I thank T. Zaslavsky for suggesting the title of the paper.

The idea of **MSOL**-polynomials occurs first in [24]. It occurred to me after reading P. Bürgisser's [17]. Note also that our general framework is somewhat inspired by [40], but both the scope and the emphasis are quite different. Initial work in this direction may be found in [59].

## References


1. M. Aigner. The Penrose polynomial of graphs and matroids. In *Surveys in Combinatorics, 2001 (Sussex)*, volume 288 of *London Mathematical Society Lecture Note Series*, pages 11–46. Cambridge University Press, 2001.
2. M. Aigner and H. van der Holst. Interlace polynomials. *Linear Algebra and Applications*, 377:11–30, 2004.
3. N. Alon and M. Tarsi. Colorings and orientations of graphs. *Combinatorica*, 12:125–134, 1992.
4. A. Andrzejak. An algorithm for the Tutte polynomials of graphs of bounded treewidth. *Discrete Mathematics*, 190:39–54, 1998.
5. R. Arratia, B. Bollobás, and G.B. Sorkin. The interlace polynomial: a new graph polynomial. *Journal of Combinatorial Theory, Series B*, 92:199–233, 2004.
6. R. Arratia, B. Bollobás, and G.B. Sorkin. A two-variable interlace polynomial. *Combinatorica*, 24.4:567–584, 2004.
7. I. Averbouch and J.A. Makowsky. The complexity of multivariate matching polynomials. Preprint, January 2007.
8. D. Babić, A. Graovac, B. Mohar, and T. Pisanski. The matching polynomial of a polygraph. *Discrete Applied Mathematics*, 15:11–24, 1986.
9. G.D. Birkhoff. A determinant formula for the number of ways of coloring a map. *Annals of Mathematics*, 14:42–46, 1912.
10. M. Bläser and H. Dell. Complexity of the cover polynomial. Preprint, January 2007.
11. M. Bläser and J.A. Makowsky. On the complexity of generalized Tutte polynomials. In preparation, 2006.
12. L. Blum, F. Cucker, M. Shub, and S. Smale. *Complexity and Real Computation*. Springer Verlag, 1998.
13. L. Blum, M. Shub, and S. Smale. On a theory of computation and complexity over the real numbers. *Bulletin of the American Mathematical Society*, 21:1–46, 1989.
14. B. Bollobás. *Modern Graph Theory*. Springer, 1999.
15. B. Bollobás and O. Riordan. A Tutte polynomial for coloured graphs. *Combinatorics, Probability and Computing*, 8:45–94, 1999.
16. P. Bürgisser. On the structure of Valiant's complexity classes. *Discr. Math. Theoret. Comp. Sci.*, 3:73–94, 1999.
17. P. Bürgisser. *Completeness and Reduction in Algebraic Complexity Theory*, volume 7 of *Algorithms and Computations in Mathematics*. Springer, 2000.
18. T.Y. Chow. The path-cycle symmetric function of a digraph. *Advances in Mathematics*, 118:71–98, 1996.
19. F.R.K. Chung and R.L. Graham. On the cover polynomial of a digraph. *Journal of Combinatorial Theory, Ser. B*, 65(2):273–290, 1995.
20. B. Courcelle. A multivariate interlace polynomial. Preprint, December 2006.

21. B. Courcelle. The expression of graph properties and graph transformations in monadic second-order logic. In G. Rozenberg, editor, *Handbook of graph grammars and computing by graph transformations, Vol. 1 : Foundations*, chapter 5, pages 313–400. World Scientific, 1997.
22. B. Courcelle, B. Godlin, and J.A. Makowsky. Towards a theory of graph polynomials, I: Second order definable polynomials. In preparation, 2007.
23. B. Courcelle and J.A. Makowsky. Recursive definitions of graph polynomials. In preparation, 2006.
24. 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*, 108(1-2):23–52, 2001.
25. Bruno Courcelle. The monadic second-order logic of graphs III: Treewidth, forbidden minors and complexity issues. *Informatique Théorique*, 26:257–286, 1992.
26. D.M. Cvetković, M. Doob, and H. Sachs. *Spectra of Graphs.* Johann Ambrosius Barth, 3rd edition, 1995.
27. N. Dershowitz and J.-P. Jouannaud. Rewrite systems. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B, chapter 6. Elsevier Science Publishers, 1990.
28. F.M. Dong, K.M. Koh, and K.L. Teo. *Chromatic Polynomials and Chromaticity of Graphs.* World Scientific, 2005.
29. R.G. Downey and M.F Fellows. *Parametrized Complexity.* Springer, 1999.
30. M. Dyer and C. Greenhill. The complexity of counting graph homomorphisms. *Random Structures and Algorithms*, 17.3-4:260 – 289, 2000.
31. H.D. Ebbinghaus and J. Flum. *Finite Model Theory.* Perspectives in Mathematical Logic. Springer, 1995.
32. J. Ellis-Monaghan. New results for the Martin polynomial. *Journal of Combinatorial Theory, Series B*, 74:326–352, 1998.
33. G.E. Farr. The Go polynomials of a graph. *Theoretical Computer Science*, 306:1–18, 2003.
34. E.J. Farrell. On a general class of graph polynomials. *Journal of Combinatorial Theory, Series B*, 26:111–122, 1979.
35. D.C. Fisher and A.E. Solow. Dependence polynomials. *Discrete Mathematics*, 82:251–258, 1990.
36. J. Flum and M. Grohe. *Parameterized complexity theory.* Springer, 2006.
37. I. Gessel. Generalized rook polynomials and orthogonal polynomials. In *IMA Volumes in Mathematics and Its Applications*, volume 18, pages 159–176. Springer, 1989.
38. C. Godsil and G. Royle. *Algebraic Graph Theory.* Graduate Texts in Mathematics. Springer, 2001.
39. C.D. Godsil. *Algebraic Combinatorics.* Chapman and Hall, 1993.
40. E. Grädel and Y. Gurevich. Metafinite model theory. *Information and Computation*, 140:26–81, 1998.
41. W. Hodges. *Model Theory*, volume 42 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, 1993.
42. C. Hoede and X. Li. Clique polynomials and independent set polynomials of graphs. *Discrete Mathematics*, 125:219–228, 1994.
43. N. Immerman. *Descriptive Complexity.* Graduate Texts in Computer Science. Springer, 1999.
44. F. Jaeger. Tutte polynomials and link polynomials. *Proceedings of the American Mathematical Society*, 103:647–654, 1988.

45. F. Jaeger, D.L. Vertigan, and D.J.A. Welsh. On the computational complexity of the Jones and Tutte polynomials. *Math. Proc. Camb. Phil. Soc.*, 108:35–53, 1990.
46. D.S. Johnson. A catalog of complexity classes. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume 1, chapter 2. Elsevier Science Publishers, 1990.
47. L.H. Kauffman. A Tutte polynomial for signed graphs. *Discrete Applied Mathematics*, 25:105–127, 1989.
48. I.A. Lavrov. Effective inseparability of the set of identically true formulas and the set of formulas with finite counterexamples for certain elementary theories (in russian). *Algebra i Logika*, 2:5–18, 1962.
49. L. Libkin. *Elements of Finite Model Theory.* Springer, 2004.
50. M. Linial. Hard enumeration problems in geometry and combinatorics. *SIAM Journal of Algebraic and Discrete Methods*, 7:331–335, 1986.
51. M. Lotz and J.A. Makowsky. On the algebraic complexity of some families of coloured Tutte polynomials. *Advances in Applied Mathematics*, 32(1-2):327–349, 2004.
52. L. Lovasz and M.D. Plummer. *Matching Theory*, volume 29 of *Annals of Discrete Mathematics*. North Holland, 1986.
53. J.A. Makowsky. Colored Tutte polynomials and Kauffman brackets on graphs of bounded tree width. In *Proceedings of the 12th Symposium on Discrete Algorithms*, pages 487–495. SIAM, 2001.
54. J.A. Makowsky. Algorithmic uses of the Feferman-Vaught theorem. *Annals of Pure and Applied Logic*, 126:1–3, 2004.
55. J.A. Makowsky. Colored Tutte polynomials and Kauffman brackets on graphs of bounded tree width. *Disc. Appl. Math.*, 145(2):276–290, 2005.
56. J.A. Makowsky. From a zoo to a zoology: Descriptive complexity for graph polynomials. In A. Beckmann, U. Berger, B. Löwe, and J.V. Tucker, editors, *Logical Approaches to Computational Barriers, Second Conference on Computability in Europe, CiE 2006, Swansea, UK, July 2006*, volume 3988 of *Lecture Notes in Computer Science*, pages 330–341. Springer, 2006.
57. J.A. Makowsky and J.P. Mariño. Farrell polynomials on graphs of bounded treewidth. *Advances in Applied Mathematics*, 30:160–176, 2003.
58. J.A. Makowsky and J.P. Mariño. The parametrized complexity of knot polynomials. *Journal of Computer and System Sciences*, 64(4):742–756, 2003.
59. J.A. Makowsky and K. Meer. On the complexity of combinatorial and metafinite generating functions of graph properties in the computational model of Blum, Shub and Smale. In *CSL'00*, volume 1862 of *Lecture Notes in Computer Science*, pages 399–410. Springer, 2000.
60. J.A. Makowsky, U. Rotics, I. Averbouch, and B. Godlin. Computing graph polynomials on graphs of bounded clique-width. In F. V. Fomin, editor, *Graph-Theoretic Concepts in Computer Science, 32nd International Workshop, WG 2006, Bergen, Norway, June 22-23, 2006, Revised Papers*, volume 4271 of *Lecture Notes in Computer Science*, pages 191–204. Springer, 2006.
61. J.A. Makowsky and B. Zilber. Polynomial invariants of graphs and totally categorical theories. Submitted, 2006.
62. K. Meer. Counting problems over the reals. *Theoretical Computer Science*, 242(1-2):41–58, 2000.
63. S.D. Noble. Evaluating the Tutte polynomial for graphs of bounded tree-width. *Combinatorics, Probability and Computing*, 7:307–321, 1998.
64. M. Otto. *Bounded variable logics and counting — A study in finite models*, volume 9. Springer-Verlag, 1997. IX+183 pages.

65. S. Oum. Approximating rank-width and clique-width quickly. In *Graph Theoretic Concepts in Computer Science, WG 2005*, volume 3787 of *Lecture Notes in Computer Science*, pages 49–58, 2005.
66. J.G. Oxley and D.J.A. Welsh. Tutte polynomials computable in polynomial time. *Discrete Mathematics*, 109:185–192, 1992.
67. C. Papadimitriou. *Computational Complexity*. Addison Wesley, 1994.
68. J. Petersen. Die Theorie der regulären Graphen. *Acta Math.*, 15:193–220, 1891.
69. P. Pitteloud. Chromatic polynomials and the symmetric group. *Graphs and Combinatorics*, 20:131–144, 2004.
70. J. Riordan. *An Introduction to Combinatorial Analysis*. Wiley, 1958.
71. A. Sokal. The multivariate Tutte polynomial (alias Potts model) for graphs and matroids. In *Survey in Combinatorics, 2005*, volume 327 of *London Mathematical Society Lecture Notes*, pages 173–226, 2005.
72. R. P. Stanley. Acyclic orientations of graphs. *Discrete Mathematics*, 5:171–178, 1973.
73. R.P. Stanley. A symmetric function generalization of the chromatic polynomial of a graph. *Advances in Mathematics*, 111:166–194, 1995.
74. J.J. Sylvester. On an application of the new atomic theory to the graphical presentation of the invariants and covariants of binary quantics, with three appendices. *American Journal of Mathematics*, 1:161–228, 1878.
75. M.A. Taitslin. Effective inseparability of the sets of identically true and finitely refutable formulae of elementary lattice theory (in Russian). *Algebra i Logika*, 1:24–38, 1961.
76. S. Toda. PP is as hard as the polynomial-time hierarchy. *SIAM J. Comput.*, 21(2):865–877, 1991.
77. N. Trinajstić. *Chemical Graph Theory*. CRC Press, 2nd edition, 1992.
78. W. T. Tutte. Graph-Polynomials. *Advances in Applied Mathematics*, 32:5–9, 2004.
79. Z. Tuza. Graph colorings with local constraints – a survey. *Discussiones Mathematicae - Graph Theory*, 17.2:161–228, 1997.
80. L. Valiant. Reducibility by algebraic projections. In *Logic and Arithmetic: An International Symposium held in honour of Ernst Specker*, volume 30 of *L'enseignement Mathématique*, pages 365–380. Université de Genève, 1982.
81. L.G. Valiant. Completeness classes in algebra. In *Proceedings of 11th STOC*, pages 249–261, 1979.
82. L.G. Valiant. The complexity of enumeration and reliability problems. *SIAM Journal on Computing*, 8(3):410–421, 1979.
83. V. I. Voloshin. *Coloring Mixed Hypergraphs: Theory, Algorithms and Applications*, volume 17 of *Fields Institute Monographs*. American Mathematical Society, 2002.
84. D.J.A. Welsh. *Complexity: Knots, Colourings and Counting*, volume 186 of *London Mathematical Society Lecture Notes Series*. Cambridge University Press, 1993.