

Colored Tutte Polynomials and Kauffman Brackets for Graphs of Bounded Tree Width

(Revised, January 2003)

J.A. Makowsky^{a,1}

^a *Department of Computer Science, Technion–Israel Institute of Technology,
Haifa, Israel
e-mail: janos@cs.technion.ac.il*

Abstract

Tutte polynomials are important graph invariants with rich applications in combinatorics, topology, knot theory, coding theory and even physics. The Tutte polynomial $T(G, X, Y)$ is a polynomial in $\mathbb{Z}[X, Y]$ which depends on a graph G . Computing the coefficients of $T(G, X, Y)$, and even evaluating $T(G, X, Y)$ at specific points (x, y) is $\sharp\mathbf{P}$ hard by a result of Jaeger, Vertigan and Welsh, (1990). On the other hand, Andrzejak and Noble (1998) have shown independently, that, if G is a graph of bounded tree width, computing $T(G, X, Y)$ can be done in polynomial time. We extend this result to the signed Tutte polynomials introduced in 1989 by Kauffman and the colored Tutte polynomials introduced in 1999 by Bollobas and Riordan. This allows us to prove similar results for the Jones polynomials and Kauffman brackets for knots and links which have a signed graph presentation of bounded tree width.

Our proof is based on, but extends considerably previous work by B. Courcelle, U. Rotics and the author. It also gives a new proof of the result for Tutte polynomials and generalizes to a wide class of polynomials defined as generating functions definable in Monadic Second Order Logic with order, but invariant under it.

Key words: Fixed parameter complexity, combinatorial enumeration, Tutte polynomial, knot polynomials

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

1 Introduction

The Tutte polynomial $T(G, X, Y)$ of a graph G , possibly with loops and multiple edges, is a polynomial in $\mathbb{Z}[X, Y]$ which depends on G and is invariant under graph isomorphisms. Jaeger in [Jae88] noticed first how to compute link polynomials of alternating links from the Tutte polynomial of their presentation as plane graphs.

An *(edge) colored graph* is a graph $\langle V, E \rangle$ together with a map $c : E \rightarrow \Lambda$ for some finite set Λ . General links can be represented by signed graphs, which are edge colored graphs where only two colors are used. Kauffman defined Tutte polynomials for signed graphs in [Kau89], and showed how to obtain link invariants from these Tutte polynomials. Bollobas and Riordan, [BR99], have introduced generalizations

$$T_{\text{colored}}(G, x_\lambda, y_\lambda, X_\lambda, Y_\lambda : \lambda \in \Lambda)$$

of the Tutte polynomial and the signed Tutte polynomial for colored graphs with arbitrary number of colors, where each color $\lambda \in \Lambda$ contributes 4 variables. The degree of $T_{\text{colored}}(G, x_\lambda, y_\lambda, X_\lambda, Y_\lambda : \lambda \in \Lambda)$ in each variable does not exceed the number of edges of G . They showed that their version is the most general version of an invariant of colored graphs satisfying certain recurrence relations with respect to deletion and contraction of edges typical for the classical Tutte polynomial.

Specializations of both the Tutte polynomials and its colored versions are important graph invariants which are intimately related to knot invariants, cf. [Thi87,Thi88,BR99]. As such they are at the core of modern knot theory which has rich applications in combinatorics, topology, coding theory, cf. [BO92,Bol99], and even physics, chemistry, molecular biology and more, cf. [Kau91,Kau95,Kaw96].

Computing the coefficients of $T(G, X, Y)$, and hence of $T_{\text{colored}}(G, x_\lambda, y_\lambda, X_\lambda, Y_\lambda : \lambda \in \Lambda)$, or even only evaluating $T(G, X, Y)$ at specific points (x, y) , is $\#\mathbf{P}$ hard by a result of Jaeger, Vertigan and Welsh, [JVW90]. We work here in the Turing model of computation, as in [GJ79]. For a survey on complexity of the Tutte polynomial, cf. [Wel93]. An analysis of the complexity of the colored Tutte polynomial in an algebraic model of computation has been given in [LM03].

In this paper we are interested in the parametrized complexity, or rather fixed parameter tractability (FPT) in the sense of [DF99]. We study the colored Tutte polynomials, where the parameter is the tree width of the colored graph, and show that they are FTP.

It was known before that, if G is a series-parallel graph, or equivalently, a graph of tree width at most 2, then computing $T(G, X, Y)$ can be done in polynomial time, by a result of Oxley and Welsh, [OW92]. More recently, Andrzejak [And98] and Noble [Nob98] have extended this to graphs of tree width at most k .

Theorem 1.1 (Andrzejak, Noble, 1998) *For graphs $G = \langle V, E \rangle$ of tree width at most k the Tutte polynomial $T(G, X, Y)$ can be evaluated and its coefficients can be computed in time $O(|V|^3)$, with lower bound for computing all the coefficients $\omega(|V|^3)$.*

Both authors make slightly more precise statements involving a closer analysis of the cost of the input data. Nobel states in his Ph.D. thesis, [Nob97], that his proof should also work for signed graphs, but the details are not given. The proofs of Noble and Andrzejak differ considerably, but both produce explicit splitting formulas for k -sums of graphs, cf. our detailed discussion in Section 2.

Our main result is

Theorem 1.2 *For colored graphs $G = \langle V, E, c \rangle$ of tree width at most k the Tutte polynomial $T_{\text{colored}}(G, x_\lambda, y_\lambda, X_\lambda, Y_\lambda : \lambda \in \Lambda)$, can be evaluated and its coefficients can be computed in polynomial time.*

This result has been first presented, with a sketch of the proof, in [Mak01b]. Our proof proceeds differently even in the case of the classical Tutte polynomial. We actually derive the result from a more general theorem, stated in Section 8 as Theorem 8.1, on graph polynomials with summation ranging over a class of subgraphs definable in Monadic Second Order Logic, cf. [CMR01, MM00, MM03a, MM03b, Mak01a]. However, explaining why and how Theorem 1.2 is a special case of Theorem 8.1, needs considerable work. In this paper we explain this in detail, so as to make the general technique also accessible to specialists of graph and knot polynomials less familiar with logic. We try to keep the paper (almost) selfcontained.

Most recently, cf. [Tra], L. Traldi has given a purely graph theoretical, or rather matroid theoretical, proof of Theorem 1.2. The advantage of our proof methods lies in the fact that it works for other graph polynomials as well, including cases which are not substitution instances of the colored Tutte polynomials, cf. Theorem 8.1. Jones polynomials and Kauffman polynomials of a link L are the most prominent invariants of knot theory, cf. [Lic88] and [Kaw96]. They are denoted by $V_L(t)$ and $\langle L \rangle(A)$ respectively in [Bo199]. Strictly speaking they are Laurent polynomials (polynomials where for each indeterminate X we also have an indeterminate X^{-1} and the identity $XX^{-1} = 1$ holds. In these cases the indeterminates (or variables) are denoted by t , respectively A .

The *crossing* diagram of a link is a plane graph where the crossings are vertices of degree 4 which are labeled as undercrossings or overcrossings. The *shaded* diagram of a link is obtained from the crossing diagram by coloring the surrounded regions black and white, starting with the outer region colored white. The vertices of the graph then are the black regions, and the edges are the shared crossings, signed $+$ or $-$ depending whether the crossing is an overcrossing or undercrossing. Such graphs are also called *signed graphs*. A link is *alternating* if it has a crossing diagram where the overcrossings and undercrossings alternate, or, equivalently, the shading diagram has only one sign. To make this precise one has to think of the plane graph as consisting of several intercrossing oriented closed curves. for details, cf. [Bol99].

By a result of Thistlethwaite, [Thi88], the Jones polynomial and Kauffman bracket of *alternating* links L are easily computable from the Tutte polynomials of their shaded diagrams $D(L)$, cf. [Wel93, Proposition 5.2.14.].

Proposition 1.3 (Thistlethwaite, 1988) *For alternating links L the the Jones polynomial and Kauffman bracket are substitution instances of the Tutte polynomial of their shading diagram $D(L)$. Hence, the computation of the Jones polynomial and the Kauffman bracket is polynomially reducible to the computation of the Tutte polynomial, in the sense of polynomial time Turing reducibility.*

Since knots and links can be represented by labeled plane graphs, the tree width of such a presentation of L is defined as the tree width of its graphical presentation. In this paper we use throughout the tree width of its shading diagram $D(L)$. Note that different representations may have different tree width, and also the tree width of the crossing diagram may differ from the tree width of the shading diagram. In [MM03b] the relation ship between the tree width of a shading diagram and its corresponding crossing diagram is studied, and it is shown that they are linearly related to each other. We shall discuss the details of the definition and the background material of tree width in Section 6.

Mighton, [Mig99], has shown that

Proposition 1.4 (Mighton, 1999) *For links L with shading diagram $D(L)$ of tree width at most 2 the Jones and Kauffman polynomials can be computed from $D(L)$ in $O(n^4)$ steps.*

From Theorem 1.2 and Proposition 1.3 we get

Corollary 1.5 *For alternating links L with shading diagram $D(L)$ (with n edges corresponding to crossings) of tree width at most k the coefficients of the Jones polynomial and the Kaufmann bracket of $D(L)$ can be computed, and hence also evaluated, in polynomial time.*

However, using another result of Thistlethwaite, [Thi87], we prove directly:

Theorem 1.6 *For arbitrary links L with shading diagram $D(L)$ (with n edges corresponding to crossings) of tree width at most k the Jones and the Kauffman polynomial of $D(L)$ can be computed in polynomial time and evaluated in time $O(n)$.*

Bollobas and Riordan have introduced a generalization of the Kauffman polynomial, the Kauffman square bracket, denoted by $[L]$. They show that this is the most general link invariant which can be obtained as a substitution instance from the colored Tutte polynomial. Hence, Theorem 1.6 also holds for the Kauffman square bracket $[L]$.

Acknowledgments. I am grateful to C. Beeri, D. Giorgetta, J. Mighton, U. Rotics and B. Wajnryb for listening to draft versions while the proof was in the making, and to V. Turaev for first suggesting the possibility of applying the techniques of [CMR01] to the Jones polynomial. I am particularly indebted to J. Oxley, who pointed out to me the references [And98,Nob98]; and to B. Courcelle, who suggested an improvement of lemma 6.8. During the work on these results I was, for part of the time, a guest at the Mathematics Department of the Swiss Federal Institute of Technology in Zurich (ETHZ) and at the Fields Institute in Toronto. I wish to thank D. Corneil and A. Nabutovsky, who, through their research grants, provided financial support for my visit in Toronto.

Since the first submission of the paper for publication five anonymous referees all contributed to the final shape of the presentation. I wish to thank them for their careful reading and valuable remarks and suggestions. I would also like to thank L. Traldi, with whom I could discuss splitting theorems for colored Tutte polynomials, and who let me see [Tra]. Finally, J. Mariño worked with me on related topics, cf. [MM03a,MM03b]. I wish to thank him for his inspiring collaboration.

2 Outline and discussion of the proof of theorems 1.2 and 1.6

We first sketch our new proof for the Tutte polynomials, as it contains almost all the essential ingredients which allow us to prove the additional results. The proof takes its initial idea from [CMR01]. However, the presentation here does not depend on it. The main theorem of [CMR01] combines methods from logic and graph theory. But to apply it, one has to do considerable additional work.

2.1 Outline of the proof

We assume our graphs have an ordering on the edges and are represented as quadruples $G = \langle V, E, R, <_E \rangle$, where V is a set of vertices, E is a set of edges, $R \subseteq V \times E$ is the incidence relation between vertices and edges, and $<_E$ is an ordering of the edges. This particular representation of graphs as relational structures turns out more convenient for us. For the tree width of graphs this changes little, as we shall see in Section 6. For the logical formalism of Monadic Second Order Logic this choice turns out to be more convenient and of stronger expressive power, cf. Section 5.

Step 1: The first step in the proof uses the spanning tree expansion of the Tutte polynomials, cf. [Bol99, Chapter X.5]. According to this the Tutte polynomials $T(G)$ of a graph can be written as

$$T(G) = \sum t_{i,j}(G) X^i Y^j \quad (+)$$

where $t_{i,j}(G)$ is the number of spanning forest of G with internal activity i and external activity j . Let F be a spanning forest (as a set of edges) of G and e an edge. If $e \in F$, $F - e$ consists now of two components, and the cut of e is the set of edges which link those two components. We call here an edge $e \in F$ *internally active* (with respect to the ordering of the edges) if it is the smallest edge (with respect to the edge ordering) in its cut. Similarly, if $e \notin F$, e defines a smallest cycle in $F \cup \{e\}$. We call then an edge $e \notin F$ *externally active* (with respect to the ordering of the edges) if it is the smallest edge (with respect to the edge ordering) in its cycle. The *internal (external) activity* of F is the number of its internally (externally) active edges.

The expression (+) can be written as

$$T(G) = \sum_F \left(\prod_{e \in F \text{ active}} X \cdot \prod_{e \in E-F \text{ active}} Y \right) \quad (*)$$

where F ranges over all spanning forests of G , $e \in F$ ranges over all internally active edges, and $e \in E - F$ ranges over all externally active edges. More details concerning Tutte polynomials are recalled in Section 3.

Our observation is that the expression (*) is a sum of monomials, where each monomial depends on a subgraph F , and the summation is over all F with a certain property P . P will be required to be definable in Monadic Second Order Logic, and so will the dependence of the monomial on F . In short, this is a generalization of the *generating functions of graph properties* as treated in [Bür00] and [CMR01] with the additional complication that an ordering of the edges is present. For details, see again Section 3.

Step 2: The next step consists of the observation that the following properties are indeed definable in Monadic Second Order Logic over graphs of the form $G = \langle V, E, R, <_E \rangle$.

- (i) F is a spanning forest of G .
- (ii) The edge $e \in F$ is (internally) active with respect to F and $<_E$.
- (iii) The edge $e \in E - F$ is (externally) active with respect to F and $<_E$.

This shows that the expression (*) is the generating function of a graph property which is Monadic Second Order definable, provided an ordering of the edges is part of the vocabulary. The details concerning Monadic Second Order Logic are recalled in section 5.

Step 3: In the next step we use the fact that the Tutte polynomial, although defined via an ordering of the edges, is *not* dependent on the ordering. This allows us to choose an ordering of the edges of the input graph, which will depend on a given tree decomposition of the graph in such a way, that it facilitates the inductive computation of the Tutte polynomial. Furthermore, we can replace the ordering by a *successor relation*.

Step 4: The final step consists in the adaptation of the techniques of [CMR01] to generating functions of graph properties for graphs of tree width at most k with a suitably chosen successor relation on the edges. For details, see section 6.

Step 5: To get theorem 1.2 we now just observe that the colored Tutte polynomials have a spanning tree expansion like (*) where the (colored) activity is defined using the order of the edges and their respective color, but is independent of the order on the edges.

Step 6: For the Kauffman bracket $\langle L \rangle(A)$ we proceed similarly as in the previous steps. The main ingredient here is the spanning tree expansion of the Kauffman polynomial from [Thi87]. One uses here a shading diagram $D(L)$ of a link L . Given a spanning tree F and an edge e of a shading diagram, Thistlethwaite introduces a function $\mu_F(e)$ whose value is a simple Laurent polynomial in A and depends only on whether $e \in F$ or $e \in E - F$, its activity with respect to F and its label. He then shows that

$$\langle L \rangle(A) = \sum_{F \subset E} \left(\prod_{e \in E} \mu_F(e) \right) \quad (**)$$

where F ranges over all spanning trees of $D(L)$. We recall the necessary material on knot polynomials in Section 4. The remaining steps are essentially the same.

Step 7: For the Kauffman square bracket we observe that its definition is based on a generalization of (**).

Step 8: For the Jones polynomial we just observe, [Bol99, Theorem 19, Chapter X.6], that it can be obtained via a simple equation from the Kauffman bracket:

$$V_L(t) = \left(-t^{\frac{3w(L)}{4}}\right) \langle L \rangle (t^{-\frac{1}{4}}) \quad (***)$$

Here $w(L)$ is the twist number of the link L .

2.2 Comparison with other proofs

Logic vs. pure graph theory. Our method is different from the methods of Andrzejak and Noble [And98,Nob98], of computing the Tutte polynomial for graphs of tree width at most k . It is also different from the method of Mighton, [Mig99], for the Jones and Kauffman polynomial, for series-parallel graphs. We use the logical method, due to Courcelle, which was extended to the computation of generating functions of graph properties in [CMR01]. However, we cannot apply [CMR01] directly. Instead we have to handle the complication of allowing the linear order of the edges to be used in the definition, although in an invariant way. This notion of definability was studied systematically in [Mak97]. Although our algorithm has, asymptotically, a similar running time to the one described in [Nob98] for the classical Tutte polynomial for tree width at most k , its current version is impractical due to the size of the constants involved. But it may be a question of time, cf. the discussion in [DF99], till more practical algorithms emerge. Recently, Grohe and Frick have shown that checking whether a formula of Monadic Second order Logic of fixed size k is true in a graph of size n , is not FPT in a very strong sense, cf. [GF02]. However, this is of little relevance here, as our proof uses such a model checking procedure only for graphs of size $k + 1$, where k is the bound of the tree width.

In the following we discuss more precisely the relationship between the various proofs of theorem 1.1.

Series-parallel graphs. In [OW92], the algorithm for computing the Tutte polynomial of a series-parallel graph is reduced to the problem of computing the Tutte polynomial of the 2-sum $G_1 \sqcup_e G_2$ of two graphs G_1 and G_2 over a common edge e . The resulting graph is like a disjoint union, but the glued together along the edge e , while the common edge is finally omitted. They first observe that series-parallel graphs can be obtained by iteratively applying 2-sums to previous obtained graphs, starting with a simple edge. They then provide the very elegant formula

$$\begin{aligned} \alpha \cdot T(G_1 \sqcup_e G_2) = \\ \alpha_{0,0} \cdot T(G_1 - e) \cdot T(G_2 - e) + \alpha_{0,1} \cdot T(G_1 - e) \cdot T(G_2/e) + \\ \alpha_{1,0} \cdot T(G_1/e) \cdot T(G_2 - e) + \alpha_{1,1} \cdot T(G_1/e) \cdot T(G_2/e) \end{aligned} \quad (\dagger)$$

where $\alpha, \alpha_{i,j} \in \mathbb{Z}[X, Y]$ are independent of the graphs G_1, G_2 and $G - e$ and G/e denote the graphs obtained from G by deleting, respectively contracting the edge e . The formula (\dagger) does not seem to generalize in such an elegant and simple way.

Splitting formulas for Tutte polynomials Both, Noble [Nob98] and Andrzejak [And98], provide algorithms which replace the 2-sum by k -sums. $G = \langle V, E \rangle$ is a k -sum of $G_1 = \langle V_1, E_1 \rangle$ and $G_2 = \langle V_2, E_2 \rangle$ if $E_1 \cap E_2 = \emptyset$ and $V_1 \cap V_2 = X$ has exactly k elements. Previously, Negami [Neg87] has given a general way on how to extend the formula (\dagger) for arbitrary k . The complication shows by having to consider all partitions P of X into sets $P = \{X_1, \dots, X_m\}$ and consider the graphs G_i/P obtained from G_i by identifying all vertices in the X_j 's. *Negami's splitting formula for Tutte polynomials* now says

$$\alpha \cdot T(G) = \sum_{P, P'} \alpha_{P, P'} \cdot T(G_1/P) \cdot T(G_2/P') \quad (\dagger\dagger)$$

where P, P' range over all partitions of X . Andrzejak explicitly refers to [Neg87], whereas Noble implicitly uses an adhoc variation of $(\dagger\dagger)$.

This approach seems to be specially tailored for the Tutte polynomials. It does not generalize in an obvious way both for colored Tutte polynomials and Kauffman brackets, although Noble states in his Thesis [Nob97] that his proof should work for the signed Tutte polynomial. In our approach we really look simultaneously at all polynomials arising from generating functions definable in Monadic Second Order Logic and at more general forms of k -sums. We give in [CMR01] something like the most general form of the formula $(\dagger\dagger)$, which has its origin in the Feferman-Vaught theorem of [FV59]. The price we pay for our generality is a drastic increase in the constants involved.

It remains an interesting challenge to find splitting formulas for colored Tutte polynomials, Kauffman brackets and Jones polynomials.

Since completion of our work, L. Traldi has informed us in January 2003 that he has been able to generalize Negami's formula for the case of the colored Tutte polynomial, [Tra]. On the other hand, in [Mak01a] we have formulated a very general Splitting Theorem *MSOL*-definable graph polynomials.

3 Tutte polynomials

In this section we collect the necessary background on Tutte polynomials. We follow closely [Bol99, chapter X], but we only collect the material necessary to understand our line of reasoning. Graphs may have multiple edges and loops. So we think of them as relational structures consisting of a set of vertices V , a set of edges E and an incidence relation $R \subseteq V \times E$, indicating which vertices are incident with a given edge (there are at most two of them). We always assume that V and E are disjoint. When no confusion can arise we omit reference to R and write $e = (u, v)$ for $(u, e) \in R$ and $(v, e) \in R$, although in the case of multiple edges this is not precise. The union of two graphs $\langle V_1, E_1 \rangle$ and $\langle V_2, E_2 \rangle$ is the graph $\langle V_1 \cup V_2, E_1 \cup E_2 \rangle$. The union is disjoint if $V_1 \cap V_2 = \emptyset$ and $E_1 \cap E_2 = \emptyset$.

Let $G = \langle V, E \rangle$ be a graph and $e = (u, v)$ be an edge in E . We denote by

- $G - e$ the graph $G - e = \langle V, E - \{e\} \rangle$;
- G/e the graph $G/e = \langle V - \{u\}, E' \rangle$, where $(u', v') \in E'$ iff $u' \neq u$ and either $v' \neq v$ and $(u', v') \in E$ or $v' = v$ and $(u', v) \in E$ or $(u', u) \in E$;
- $G(e)$ the graph which consists only of the edge e . $G(e)$ has either one vertex and is a loop, or has two vertices and a single edge e connecting them.

B denotes the graph $\langle \{u, v\}, \{(u, v)\} \rangle$, a single bridge, L denotes the graph $\langle \{v\}, \{(v, v)\} \rangle$, a single loop, and E_n denotes the graph on n vertices without any edges.

Tutte polynomials: Definitions. There are several equivalent definitions of the Tutte polynomial: As a universal function on graphs or matroids satisfying certain recurrence relations, as a rank generating function of (graphical) matroids, and as generating function of certain spanning trees. We shall not include the definition via rank generating functions, because it does not fit our logical approach of the sequel. However, upon seeing an unpublished version of this paper in January 2003, L. Traldi, [Tra], has used this approach to give a purely graph theoretic proof of our Theorem 1.2.

We shall use as our definition Brylawski's characterization of Tutte polynomials as given in [BO92, Theorem 6.2.2] or [Bol99, theorem 2, chapter X.2]:

Theorem 3.1 (Brylawski, 1971) *There is a unique function T from the class of graphs \mathcal{G} into the polynomial ring $\mathbb{Z}[X, Y]$ having the following properties:*

- (i) $T(B, X, Y) = X$, $T(L, X, Y) = Y$ and $T(E_n, X, Y) = 1$.
- (ii) If $G = \langle V, E \rangle$, $e \in E$ and e is neither a bridge nor a loop, then

$$T(G, X, Y) = T(G - e, X, Y) + T(G/e, X, Y)$$

- (iii) If e is either a bridge or a loop of G then

$$T(G, X, Y) = T(G(e), X, Y) \cdot T(G - e, X, Y)$$

Furthermore, for graphs G which are the union of G_1 and G_2 with at most one vertex in common, we have $T(G) = T(G_1) \cdot T(G_2)$.

The Tutte polynomial is this unique function. Theorem 3.1 allows us to compute $T(G)$ if we choose an ordering on the edges. It also establishes that the result of this computation is not dependent on the ordering chosen. However, using Theorem 3.1 recursively leads to an algorithm which uses, in general, exponential time. This is so, because unfolding the recursive definition leads to a binary tree with an exponential number of nodes.

Tutte polynomials: Spanning tree expansion. From now on we assume we have a fixed ordering on the edges of $G = \langle V, E \rangle$, $E = \{e_0, e_1, \dots, e_m\}$. We now follow closely [Bol99, chapter X.5]. A graph $F = \langle V_F, E_F \rangle$ is a *spanning forest* of the graph G if $V = V_F$ and $E_F \subseteq E$, and each component of F is a spanning tree of a component of G .

Let F be a spanning forest of G . For $e \in E_F$ we define

$$Cut_F(e) = \{f \in E - E_F : \langle V, E_F - \{e\} \cup \{f\} \rangle \text{ is a spanning forest of } G\}.$$

We say that $e \in E_F$ is *internally active for F* (with respect to the fixed ordering of E) if e is the smallest edge in $Cut_F(e)$. For $e \in E - E_F$ we define $Cycle_F(e)$ to be the unique cycle in $E_F \cup \{e\}$. We say that $e \in E - E_F$ is *externally active for F* (with respect to the fixed ordering of E) if e is the smallest edge in $Cycle_F(e)$. An (i, j) -forest F for G is a spanning forest which has exactly i internally active and j externally active edges. The following theorem is due to Tutte, [Tut54], cf. also [Bol99, theorem 10, chapter X.5].

Theorem 3.2 (Tutte 1954) *Let G be a graph with an ordering on its edges. Let $t_{i,j}(G)$ denote the number of (i, j) forests in G . Then*

$$T(G) = \sum_{i,j} t_{i,j}(G) X^i Y^j \quad (+)$$

In particular, the coefficients $t_{i,j}(G)$ are independent of the ordering.

Hence, using Theorem 3.2, we can write

$$T(G) = \sum_F \left(\prod_{e \in F \text{ active}} X \cdot \prod_{e \in E-F \text{ active}} Y \right) \quad (*)$$

where F ranges over all spanning forests of G , $e \in F$ ranges over all internally active edges, and $e \in E - F$ ranges over all externally active edges.

Using equation (*) for the computation also leads to an algorithm which uses exponential time. This is so, because, by a celebrated theorem due to Cayley, cf.[HP73, Theorem 1.7.2], the number of trees on n vertices is n^{n-2} , hence a clique has an exponential number of spanning trees. However, on graphs of tree width at most k , this approach will lead to a polynomial algorithm.

Colored Tutte polynomials. Bollobas and Riordan, [BR99], introduce the *colored Tutte polynomial* for a colored graph $G = \langle V, E, R_{<}, c \rangle$ with $c : E \rightarrow \Lambda$ a coloring and $R_{<}$ an ordering on its edges. The polynomial is in

$$\mathbb{Z}_\Lambda = \mathbb{Z}[X_\lambda, Y_\lambda, x_\lambda, y_\lambda : \lambda \in \Lambda].$$

Let T be a spanning tree of G and $e \in E$ with $c(e) = \lambda \in \Lambda$. We first assume that graphs are connected. The notion of internally and externally active edges with respect to T and $R_{<}$ remains unchanged.

Definition 3.3 *The colored weight of e with respect to T with $c(e) = \lambda$ is defined by*

$$w(G, c, R_{<}, T, e) = \begin{cases} X_\lambda & \text{if } e \text{ is internally active} \\ Y_\lambda & \text{if } e \text{ is externally active} \\ x_\lambda & \text{if } e \text{ is not internally active} \\ y_\lambda & \text{if } e \text{ is not externally active} \end{cases}$$

Definition 3.4 *The colored Tutte polynomial $T_{\text{colored}}(G; X_\lambda, Y_\lambda, x_\lambda, y_\lambda : \lambda \in \Lambda)$ is defined for connected graphs by*

$$\sum_T \left(\prod_{e \in E} w(G, c, R_{<}, T, e) \right) \quad (*,c)$$

where T ranges over all spanning trees of G .

For graphs which are not connected one extends this definition by the product rule.

$$T_{\text{colored}}(G; X_\lambda, Y_\lambda, x_\lambda, y_\lambda : \lambda \in \Lambda) = \prod_C T_{\text{colored}}(C; X_\lambda, Y_\lambda, x_\lambda, y_\lambda : \lambda \in \Lambda) \quad (1)$$

where C ranges over all connected components of G .

Finally, to assure order invariance of this definition, one needs, that certain relations between the indeterminates hold, cf. [BR99, Theorem 2].

Theorem 3.5 *Let $I_0 \subseteq \mathbb{Z}_\Lambda$ be the minimal Ideal such that*

$$\begin{aligned} X_\lambda y_\mu - y_\lambda X_\mu - x_\lambda Y_\mu + Y_\lambda x_\mu &\in I_0, \\ X_\nu(x_\lambda Y_\mu - Y_\lambda x_\mu - x_\lambda y_\mu + y_\lambda x_\mu) &\in I_0, \\ Y_\nu(x_\lambda Y_\mu - Y_\lambda x_\mu - x_\lambda y_\mu + y_\lambda x_\mu) &\in I_0 \end{aligned}$$

for any colours λ, μ and ν . Then $T_{\text{colored}}(G; X_\lambda, Y_\lambda, x_\lambda, y_\lambda : \lambda \in \Lambda)$ is independent of the edge ordering iff there is a polynomial $P \in \mathbb{Z}_\Lambda$ such that

$$P \equiv 0 \pmod{I_0}$$

describes the relationship between the indeterminates.

In [BR99] this definition is justified by theorems similar to Brylawski's theorem 3.1. All we need is this definition using spanning trees.

4 Knot polynomials

Here we follow closely [Bol99, chapter X.6]. An *oriented link* L is a subset of \mathbb{R}^3 consisting of n disjoint piecewise linear simple closed oriented curves. A *knot* is a connected oriented link. With a link we associate a plane oriented graph where the vertices are the crossings of the link, the oriented edges are the curves between two crossings and the vertices are labeled $+1$ or -1 , depending whether the link crosses above or below if we follow its orientation. The crossing diagram $D(L)$ of a link L is therefore an oriented graph with labels on the vertices. The *size* of $D(L)$ is the number of its crossings. The *twist number* of $D(L)$, denoted by $w(L)$, is the sum of the labels of its crossings. The crossing diagram divides the plane into regions which can be colored black and white. From this we can construct the shading diagram, which is a planar graph with the black regions as vertices which are connected by signed edges iff the regions share a crossing. The sign of the edge is the same as sign of the shared crossing. The twist number of the crossing diagram is the sum of the signs of the edges of the corresponding shading diagram. For a fixed plane representation of an oriented link the crossing diagram and the shading diagram determine each other uniquely, cf. [Sos99].

The Kauffman bracket and Jones polynomial of a link L , $\langle L \rangle(A)$ and $V_L(t)$ respectively, are uniquely defined from its crossing diagram $D(L)$. They are Laurent polynomials in one variable A , respectively t (hence allowing also negative exponents in the variables). They are both defined from a more general polynomial in $\mathbb{Z}[A, B, d]$, the Kauffman square bracket $[L]$.

For our purpose we do not need their definitions, but instead use two theorems characterizing $\langle L \rangle(A)$ and $V_L(t)$.

Let F be a spanning tree of a connected shading diagram $D(L)$ of an oriented link L , and let e be an edge of $D(L)$. Thistlethwaite, in [Thi87], introduces a function $\mu_F(e)$ whose values are among the Laurent monomials $\{A, A^{-1}, -A^3, -A^{-3}\}$ in A and depends only on whether $e \in F$ or $e \in E - F$, its activity with respect to F , and its label². He then shows the following theorem:

Theorem 4.1 (Thistlethwaite, 1987) *The Kauffman bracket of a shading diagram $D(L)$ of a link L is*

$$\langle L \rangle(A) = \sum_{F \subset E} \left(\prod_{e \in E} \mu_F(e) \right) \quad (**)$$

where F ranges over all spanning trees of $D(L)$.

For the Jones polynomial we just observe, that it can be obtained via a simple equation from the Kauffman polynomial, cf. [Bol99, theorem 19, chapter X.6].

Theorem 4.2 *The Jones polynomial of an oriented link L is given by*

$$V_L(t) = (-t^{\frac{3w(L)}{4}} \langle L \rangle(t^{-\frac{1}{4}})) \quad (***)$$

where $w(L)$ is the twist number of the link L .

Finally, the Kauffman square bracket by the following proposition [BR99, theorem 16].

Proposition 4.3 *The Kauffman square bracket $[L](A, B, d)$ is obtained from the colored Tutte polynomial of the shading diagram $D(L)$ by a set of suitable the substitutions. hence, its computation is polynomial time reducible to the computation of the colored Tutte polynomial.*

The exact nature of these substitutions is rather complicated, and not relevant for our purposes.

² The explicit definition of $\mu_F(e)$ is too space consuming to be given here, and would not add to our understanding.

5 Monadic Second Order Logic

In this section we show that the summation and products in the expression (*) range over properties definable in Monadic Second Order Logic, as used in [CMR01]. For this we view our graph G as a two-sorted³ structure \mathcal{G} with vertices V and edges E is 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 , and a successor relation⁴ on the edges S . The linear ordering can be recovered from the successor relation S via the transitive closure of S .

Clearly every graph G with an ordering on the edges has a presentation as such a structure \mathcal{G} , which is unique up to numbering of the vertices.

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 subset of edges. Atomic formulas are

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

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 $MSOL(\mathcal{G}_{\text{colored}})$ allows additionally for unary predicates $C_\lambda, \lambda \in \Lambda$, which are interpreted as the edges of color λ . Hence we have additional atomic formulas $C_\lambda(x_i)$. The semantics is defined as for First Order Logic, with the exception that the set variables now range over subsets of the domains. The reader not familiar with Monadic Second Order Logic should consult [EF95] for the precise definition of its semantics.

For a detailed discussion of the expressive power of $MSOL(\mathcal{G})$ and $MSOL(\mathcal{G}_{\text{colored}})$ we refer to [Cou90,Cou92,ALS91].

The following is easy (but may be tedious):

Proposition 5.1 *The following graph properties are expressible in $MSOL(\mathcal{G})$:*

- (i) $\phi_{\text{ord}(S)}(x, y, S)$: $x < y$ in the linear ordering which is the transitive closure

³ The reader not familiar with many-sorted structures should consult the introductory chapters of [BF85] or [EFT80,EF95]. Alternatively, one can think of a two-sorted structure as a structure with one universe and a partition of the universe into two sets, V and E , which are the interpretation of two unary predicate symbols P_V and P_E .

⁴ A successor relation on a finite linearly ordered set is a binary relation, such that $(a, b) \in S$ iff b is the smallest element bigger than a , if such a b exists, otherwise $b = a$.

of the successor relation S .

- (ii) $\phi_{\text{path}}(v_1, v_2, X, R)$: X is a path from v_1 to v_2 in G .
- (iii) $\phi_{\text{component}}(U, R)$: U is a connected component of G .
- (iv) $\phi_{\text{tree}}(X, U, R)$: U is a connected component of G and X is a spanning tree of the component U .
- (v) $\phi_{\text{forest}}(F, R)$: F is a spanning forest of G .
- (vi) $\phi_{\text{cut}}(x, e, F, R)$: x is in $\text{Cut}_F(e)$ of G .
- (vii) $\phi_{i\text{-active}}(x, F, R, S)$: x is internally active for F in G .
- (viii) $\phi_{\text{cycle}}(x, e, F, R)$: x is in $\text{Cycle}_F(e)$ of G .
- (ix) $\phi_{e\text{-active}}(x, F, R, S)$: x is externally active for F in G .

Proof: We sketch the proofs of (i), (iii) and (v), and leave the other cases as exercises to the reader.

(i) We first define S_e as the smallest subset of E which contains e , and whenever $e_1 \in S_e$ and $(e_1, e_2) \in S$ then also $e_2 \in S_e$. The formula defining S_e is now $\psi_1(U_e, e)$ given by

$$(U_e(e) \wedge \forall e_1, e_2 ((U_e(e_1) \wedge S(e_1, e_2)) \rightarrow U_e(e_2)) \wedge \\ \forall U ((U_e(e) \wedge \forall e_1, e_2 ((U(e_1) \wedge S(e_1, e_2)) \rightarrow U(e_2)) \rightarrow \forall f (U_e(f) \leftrightarrow U(f))))$$

Now we can define the linear order $e_1 \leq e_2$ by $S_{e_1} \subseteq S_{e_2}$. This is expressed as $\phi_{\text{ord}(S)}(e_1, e - 2, S)$ using ψ_1 by

$$\forall f, U_{e_1}, U_{e_2} ((U_{e_1}(f) \wedge \psi_1(U_{e_1}, e_1) \wedge \psi_1(U_{e_2}, e_2)) \rightarrow U_{e_2}(f))$$

(iii) A set of vertices U is a connected component of a graph G if it is a minimal set which contains with every vertex v in U also its neighbors. We can write this as $\phi_{\text{component}}(U, R)$ given by

$$\forall v_1, v_2 (\exists e (R(v_1, e) \wedge R(v_2, e) \wedge U(v_1)) \rightarrow U(v_2)) \wedge \\ \forall V (\forall v_1, v_2 (\exists e (R(v_1, e) \wedge R(v_2, e) \wedge V(v_1)) \rightarrow V(v_2)) \rightarrow \forall v (U(v) \leftrightarrow V(v)))$$

(v) To say that F is a spanning forest, we just say that the set of vertices v which are incident to an edge e in F consists of all vertices. The formula $\phi_{\text{forest}}(F, R)$ then is

$$\forall v \exists e (F(e) \wedge R(v, e))$$

□

Proposition 5.1 establishes the following:

Corollary 5.2 *The Tutte polynomial $T(G)$ is an $\text{MSOL}(\mathcal{G})$ -definable generating function of the graph G with a successor relation on the edges.*

Proof: The expression $(+)$ can be written as

$$T(G) = \sum_F \left(\prod_{e \in F \text{ active}} X \cdot \prod_{e \in E-F \text{ active}} Y \right) \quad (*)$$

where F ranges over all spanning forests of G , $e \in F$ ranges over all internally active edges, and $e \in E - F$ ranges over all externally active edges.

By Proposition 5.1, these three graph properties are expressible in $MSOLa(\mathcal{G})$. \square

Similarly, using the spanning tree definition $(*,c)$ for the colored Tutte polynomial, we get the following:

Corollary 5.3 *The colored Tutte polynomial $T_{\text{colored}}(G, c)$ is an $MSOL(\mathcal{G}_{\text{colored}})$ -definable generating function of the graph G with a successor relation on the edges.*

6 Tree Width

We now recall one of the various equivalent definitions of *tree width of a graph* $G = \langle V, E \rangle$ and also of a *two-sorted relational structure* $\mathcal{G} = \langle V, E, R, S \rangle$. Recall that we always assume that V and E are disjoint sets. In the latter case, the idea is to consider $V \sqcup E$, the union of V and E , as one universe, and then look at the two relations separately or together, as if R and S were one relation. Among the first to use such a definition for relational structures are [FV93]. The tree width of a graph viewed as a one-sorted structure with universe V or as a two-sorted structure with universes V and E differs, but not by much, cf. Lemma 6.7 below. Other definitions are based on partial k -trees, or various graph grammars, cf. [CM02]. General background on tree width may be found in [Die96].

Definition 6.1 *A k -tree decomposition of \mathcal{G} with respect to R ($R \sqcup S$) 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 fathers.*
- (ii) *The universe $V \sqcup E$ is covered by sets A_t , with $t \in T$ and $|A_t| \leq k + 1$.*
- (iii) *For each $x \in V \sqcup E$ the set $T(x) = \{t \in T : x \in A_t\}$ is a (connected) subtree of \mathcal{T} .*
- (iv) *For each $(x, y) \in R$ ($(x, y) \in R \sqcup S$) there is a $t \in T$ with both $x, y \in A_t$.*

Note that in the presence of additional unary relations P the last condition is vacuous.

Definition 6.2 \mathcal{G} is of tree width at most k with respect to $R (R \sqcup S)$, if there exists a k -tree decomposition of \mathcal{G} with respect to $R (R \sqcup S)$.

For fixed k , checking whether \mathcal{G} has tree width at most k (and if yes, finding a witnessing tree decomposition) can be done in polynomial time, cf. [Bod97]. It can even be done in linear time, but then the constants are prohibitively large. If we guess k slightly above its optimal value, the algorithms become more efficient.

Remark 6.3 If we add unary predicates (labels) to \mathcal{G} , the notion of tree width does not change. Therefore the tree width of a crossing or shading diagram is just the tree width of its underlying graph. Also the tree width of an edge colored graph is the same as its tree width without the coloring.

Remark 6.4 Our definition does not distinguish between the tree width of a directed graph or its undirected version (where there is an edge wherever there was a directed edge). There are attempts in the literature to have a more sophisticated definition of tree width for directed graphs, cf. [JRST03]. But we do not need this here. One should note however, that there are classes of graphs of unbounded tree width but of bounded directed tree width.

Remark 6.5 If S is a linear order on \mathcal{G} , then the tree width of \mathcal{G} with respect to S is $\Omega(|V \sqcup E|)$. To see this note that the graph obtained from a linear ordering of n elements by putting an edge between every two vertices which are comparable gives us a clique, which has tree width $n - 1$. However, the tree-width of any a successor relation is 1.

Remark 6.6 If R is a tree and S is a successor relation, both separately have tree width 1, but it is easy to arrange S in a way that $R \sqcup S$ has tree width $\Omega(|V \sqcup E|)$. This is even true if we have two successor relations S_1 and S_2 on n elements. Let S_1 be the natural successor defined by $(k, k + 1) \in S_1$ for $k < n$, and choose m relatively prime to n and define S_2 by $(k, k + m \pmod{n}) \in S_2$ for $k < n$. Let $E = S_1 \cup S_2$ define an edge relation on these n elements. For n large enough this contains a clique of size m , hence the tree width is at least $m - 1$.

The first lemma states the relationship between the tree width of $G = \langle V, E \rangle$, where the universe is V and E is a binary relation, and $\hat{G} = \langle V \sqcup E, R \rangle$, cf. [CMR01].

Lemma 6.7 If $G = \langle V, E \rangle$ has tree width at most k , so so $\hat{G} = \langle V \sqcup E, R \rangle$ has tree width at most $k + 1$.

Proof: \hat{G} is obtained from G by making each old edge into a new vertex and adding two new edges connecting the new vertex with the old vertices incident with the new edge. Now for each old edge Now assume the tree decomposition

$A_t, t \in T$ of G is given. For each edge of G we do successively the following: If A_t contains an edge $e = (u, v)$, we replace A_t by two copies of $A_t \cup \{e\}$, $A_{t,0}$ and $A_{t,1}$ respectively, which in the new tree decomposition are father and son. \square

The next lemma is crucial for our setting. Let $G = \langle V, E \rangle$ be a graph, and let $<_V$ be a linear order on the vertices V of G . As we work in Monadic Second Order Logic we can equivalently look at the successor relation S_V induced by $<_V$. S_V and $<_V$ are interdefinable. Let G^+ be G augmented with an edge between v and w whenever w is the $<_V$ -successor of v .

Lemma 6.8 *For every k , there is k' such that for every $G = \langle V, E \rangle$ with tree width at most k , there is an successor relation S_V on V such that G^+ has tree width at most k' . Actually⁵, $k' = k + 3$ suffices.*

The same holds for $G = \langle V, E, R \rangle$ and the successor relation on E or even on $V \cup E$.

Proof: We note that this successor relation is *not* definable in Monadic Second Order Logic of G . One must define it from a tree decomposition where the sons of the nodes are linearly ordered.

A tree decomposition is *proper* if each $A(t)$ has at least one vertex not in $A(f(t))$. A set $A(t)$ is called the *box of the node t* .

Step 1: We start with tree decomposition of G . Without loss of generality we can assume that if $f(t_1) = f(t_2) = t$ and $A_{t_1} \cap A_{f(t)} = A_{t_2} \cap A_{f(t)}$ then $t_1 = t_2$. Hence each t has at most 2^k sons and the tree decomposition of G of width k is proper.

Step 2: By a well known fact, cf. also [dF97, Lemma 2.2.5] or [BdF01], we can assume that the tree decomposition has a binary tree of the same width.

Step 3: Now we define S_V . Within each A_t we order the vertices in a way that those elements which do not appear in $A_{f(t)}$ precede those who do appear in $A_{f(t)}$. This defines S_V within the boxes. The two boxes corresponding to the sons of the node t contain now four ordered sets which we order by putting one after the other, hence adding three more links to define S_V .

Step 4: We have to modify again the tree decomposition such that it becomes a $k + 3$ tree decomposition for $R \sqcup S$. We add to $A_{f(t)}$ the first and the last element of each A_t which is not in $A_{f(t)}$. Now we can link the undefined parts of the successor relation as required by adding at most three vertices to each box.

This completes the proof. \square

⁵ I would like to thank an unknown referee for suggesting a way of improving the value of k' .

The main result of [CMR01] can now be stated for our situation here:

Theorem 6.9 (Courcelle, Makowsky, Rotics 1998) *Let G be a graph and \mathcal{G} the corresponding structure with a successor relation on the edges. Let $\phi(F)$, $\theta_1(x, F)$ and $\theta_2(x, F)$ be formulas in $MSOL(\mathcal{G})$. Furthermore, let w_1, w_2 be functions $V \sqcup E \rightarrow \mathcal{W}$ with a finite range \mathcal{W} . If \mathcal{G} is of tree width at most k' with respect to $R \sqcup S$, then*

$$T(G, X, Y) = \sum_{\phi(F)} \left(\prod_{\theta_1(x, F)} w_1(x) \cdot \prod_{\theta_2(x, F)} w_2(x) \right) \quad (++)$$

can be computed and evaluated in polynomial time.

Remark 6.10 *The degree of the polynomial in the complexity of the computation is small (≤ 4 depending on the cost of the operations in the polynomial ring). But the constants involved are prohibitively big in the general theorem, and depend on the number of logically inequivalent $MSOL(\mathcal{G})$ -formulas of quantifier rank at most the maximal quantifier rank of ϕ, θ_1 and θ_2 . Note that the number of inequivalent $MSOL(\mathcal{G})$ -formulas of fixed quantifier rank is finite but large, cf. [Lad77] or [Mak01a] for an exact estimate.*

7 Proof of the main theorems

For theorem 1.2 we want to show that, if G is a graph of tree width at most k , then its Tutte polynomial $T(G, X, Y)$ can be computed in polynomial time over \mathbb{Q} when the arithmetic operations have unit cost.

From theorem 3.2 and the expression (++) we have

$$T(G, X, Y) = \sum_F \left(\prod_{e \in F \text{ active}} X \cdot \prod_{e \in E-F \text{ active}} Y \right) \quad (*)$$

where F ranges over all spanning forests of G , $e \in F$ ranges over all internally active edges, and $e \in E - F$ ranges over all externally active edges. For the colored case we have

$$\sum_T \left(\prod_{e \in E} w(G, c, R_{<}, T, e) \right) \quad (*,c)$$

In both cases we use Proposition 5.1, and its Corollaries 5.2 and 5.3, which make sure summation and products in (*) or (*,c) range over $MSOL(\mathcal{G})$ -definable properties. Then we use lemma 6.7 and 6.8 to make sure \mathcal{G} is of tree width at most $k' = 2k + 5$. Finally we are in position to apply Theorem 6.9 with $\mathcal{W} = \{X, Y\}$ to get the desired result.

We note that this proof easily generalizes to the more complicated *colored Tutte polynomials* of [BR99].

For the proof of theorem 1.6, that the Kauffman polynomial of a link L of tree width at most k is computable in polynomial time, we proceed similarly. We use $(^{**})$ instead of $(^*)$ and apply theorem 6.9 with $w_1(e) = \mu_F(e)$. For the definition of $\mu_F(e)$ we need eight cases, which are all definable in $MSOL(\mathcal{G})$ by Corollary 5.3.

Alternatively, we can use proposition 4.3 for both Kauffman brackets $\langle L \rangle(A)$ and $[L](A, B, d)$.

8 Conclusions and open problems

We have shown that the colored Tutte polynomials, the Kauffman brackets $\langle L \rangle$ and $[L]$ and Jones polynomials are computable in polynomial time on graphs of bounded tree width. In other words, they are in FPT. We have used two ingredients: the spanning tree expansion of these polynomials and Monadic Second Order Logic.

8.1 A general theorem for graph polynomials

Actually, we have shown the following generalization of [CMR01]:

Theorem 8.1 *Let $f(G, c, R_{<})$ be a generating function on ordered and colored graphs which is invariant under the choice of the order $R_{<}$ and which takes values in a polynomial ring $\mathbf{Z}[\bar{X}]$. If $f(G, c, R_{<})$ is Monadic Second Order definable then $f(G, c, R_{<})$ is computable in polynomial time, if it is restricted to graphs of tree width at most k .*

There are several further extensions of this theorem: Monadic Second Order Logic can be replaced by Guarded Second Order Logic as introduced in [GHO00] generalizing Guarded First Order Logic of [AvBN98]. Bounded tree width can be replaced by bounded clique width provided quantification of sets is restricted to vertices, cf. [CMR01]. For a detailed discussion, cf. [CO00]. Graphs can be replaced by arbitrary structures with an appropriate notion of clique width, cf. [CM02].

However, our method does not give the result for the various Tutte polynomials on graphs of bounded clique width. The reason that our proof does not work in this case, lies in the fact, that for the theorem to hold for bounded clique width, quantification of sets has to be restricted to vertices. But in this case

the definability of the Tutte polynomials relies on quantification over sets of edges, which we need to express the concept of a spanning tree. Nevertheless, it is conceivable that Theorem 1.2 does hold also for graphs of bounded clique width.

8.2 *Tree width and clique width of knots and links*

Knot and link invariants are topological properties. $MSOL(\mathcal{G})$ -definable properties speak about properties of link diagrams. It is usually possible to find two link diagrams of the same link which differ with respect to some $MSOL(\mathcal{G})$ -definable property. For example alternating links have also non-alternating diagrams. But alternating shading diagrams are exactly those where all the signs are equal. It is an interesting problem to find $MSOL(\mathcal{G})$ -definable properties which are proper link invariants.

To make tree width a link invariant, it should be defined as the minimal tree width of the shading or crossing diagram representing that link. But there is no established choice whether one should use the shading diagram or the crossing diagram. For a detailed discussion, cf. [MM03b]. It is worth noting that braids over k strands have a crossing diagram of tree width at most $k+1$. k -algebraic links are a generalization of k -braids via k -tangles. They were introduced by J. Przytycki in [Prz01,PT01]. In [MM03b] it is shown that the tree width of link diagrams is intimately connected to their algebraicity,

Although we have not defined clique width in this paper, we should remark that clique width at most k of a crossing or shading diagram of a link implies tree width at most $6k+1$ by a result in [GW00], because the graphs are planar. So the notion of clique width is not interesting for link diagrams.

8.3 *Directed tree width*

In the case of directed tree width, as defined in [JRST03], no analogue of Theorem 6.9 for $MSOL(\mathcal{G})$ -definable classes of graphs is known. Nevertheless, it is shown in [JRST03], that for many properties of directed graphs, which in general are **NP**-hard to verify, their verification on classes of graphs of bounded directed tree width is possible in polynomial time. We can view directed graphs as special cases of colored graphs with, say, three colors, one for each direction and one for the bi-directed edges. So it is natural to ask, whether the colored Tutte polynomial is computable in polynomial time on graphs of bounded directed tree width.

Tutte polynomials are primarily defined over matroids, cf. [Wel76,Oxl92,BO92]. However, matroids are not first order structures, as dealt with in this paper. Neither is entirely clear, what model of computation should be used best, cf. [HK81]. It would be interesting to extend our results to matroids. This would involve an appropriate notion of decomposability of matroids. First steps in this direction were undertaken in [Sey80,Tru92,OW92,And97]. Also, instead of tree width it is more convenient to use branch width of matroids. Hlineny has recently approached matroids from this point of view, and has related *MSOL*-definability of matroids presentable over finite fields, their branch width and their computability in [Hli02]

It remains a challenging task to find both a suitable version of Monadic Second Order Logic and a satisfactory theory of decomposable matroids generalizing bounded tree width or branch width of graphical matroids with less stringent hypotheses on their presentability.

8.5 *Approximation algorithms*

Finally, we should note that in this paper we were mostly interested in *exact computations* of the polynomials in question. An alternative approach consists in *approximate counting*, which is a very active field of research. For our context, excellent surveys are given in [Wel93] and [Wel97]. The approximability of the Tutte polynomial is discussed in [Wel95].

References

- [ALS91] S. Arnborg, J. Lagergren, and D. Seese. Easy problems for tree decomposable graphs. *Journal of Algorithms*, 12:308–340, 1991.
- [And97] A. Andrzejak. Splitting formulas for Tutte polynomials. *Journal of Combinatorial Theory, Series B*, 70.2:346–366, 1997.
- [And98] A. Andrzejak. An algorithm for the Tutte polynomials of graphs of bounded treewidth. *Discrete Mathematics*, 190:39–54, 1998.
- [AvBN98] H. Andréka, J. v. Benthem, and I. Németi. Modal languages and bounded fragments of predicate logic. *Journal of Philosophical Logic*, 27:217–274, 1998.
- [BdF01] H. Bodlander and B. de Fluiter. Reduction algorithms for graphs of small tree width. *Information and Computation*, 167:86–119, 2001.

- [BF85] J. Barwise and S. Feferman, editors. *Model-Theoretic Logics*. Perspectives in Mathematical Logic. Springer Verlag, 1985.
- [BO92] T. Brylawski and J. Oxley. The Tutte polynomial and its applications. In N. White, editor, *Matroid Applications*, pages 123–225. Cambridge University Press, 1992.
- [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.
- [Bol99] B. Bollobás. *Modern Graph Theory*. Springer, 1999.
- [BR99] B. Bollobás and O. Riordan. A Tutte polynomial for coloured graphs. *Combinatorics, Probability and Computing*, 8:45–94, 1999.
- [Bür00] P. Bürgisser. *Completeness and Reduction in Algebraic Complexity*, volume 7 of *Algorithms and Computation in Mathematics*. Springer, 2000.
- [CM02] B. Courcelle and J.A. Makowsky. Fusion on relational structures and the verification of monadic second order properties. *Mathematical Structures in Computer Science*, 12.2:203–235, 2002.
- [CMR01] 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.
- [CO00] B. Courcelle and S. Olariu. Upper bounds to the clique-width of graphs. *Discrete Applied Mathematics*, 101:77–114, 2000.
- [Cou90] B. Courcelle. The monadic second-order theory of graphs I: Recognizable sets of finite graphs. *Information and Computation*, 85:12–75, 1990.
- [Cou92] B. Courcelle. Monadic second-order logic of graphs VII: Graphs as relational structures. *Theoretical Computer Science*, 101:3–33, 1992.
- [dF97] B. de Fluiter. *Algorithms for Graphs of Small Treewidth*. PhD thesis, Department of Computer science, University of Utrecht, 1997.
- [DF99] R.G. Downey and M.F. Fellows. *Parametrized Complexity*. Springer, 1999.
- [Die96] R. Diestel. *Graph Theory*. Graduate Texts in Mathematics. Springer, 1996.
- [EF95] H.D. Ebbinghaus and J. Flum. *Finite Model Theory*. Perspectives in Mathematical Logic. Springer, 1995.
- [EFT80] H.D. Ebbinghaus, J. Flum, and W. Thomas. *Mathematical Logic*. Undergraduate Texts in Mathematics. Springer-Verlag, 1980.
- [FV59] S. Feferman and R. Vaught. The first order properties of algebraic systems. *Fundamenta Mathematicae*, 47:57–103, 1959.

- [FV93] T. Feder and M. Vardi. The computational structure of monotone monadic SNP and constraint satisfaction. In *STOC'93*, pages 612–622. ACM, 1993.
- [GF02] M. Grohe and M. Frick. The complexity of first-order and monadic second-order logic revisited. In *Proceedings of the 17th IEEE Symposium on Logic in Computer Science (LICS'02)*, pages 215–224. IEEE, 2002.
- [GHO00] T. Grädel, C. Hirsch, and M. Otto. Back and forth between guarded and modal logics. In *LiCS'2000*, pages 217–228. IEEE, 2000.
- [GJ79] M.G. Garey and D.S. Johnson. *Computers and Intractability*. Mathematical Series. W.H. Freeman and Company, 1979.
- [GW00] F. Gurski and E. Wanke. The tree-width of clique-width bounded graphs without $K_{n,n}$. *Lecture Notes in Computer Science*, 1928:196–205, 2000.
- [HK81] D. Hausmann and B. Korte. Algorithmic versus axiomatic definitions of matroids. *Mathematical Programming Study*, 14:98–111, 1981.
- [Hli02] P. Hliněný. Branch-width, parse trees, and monadic second order logic for matroids. preprint, 2002.
- [HP73] F. Harary and E. Palmer. *Graphical Enumeration*. Academic Press, 1973.
- [Jae88] F. Jaeger. Tutte polynomials and link polynomials. *Proceedings of the American Mathematical Society*, 103:647–654, 1988.
- [JRST03] T. Johnson, N. Robertson, P. Seymour, and R. Thomas. Directed tree-width. *Journal of Combinatorial Theory, Serie B*, xx:xx–yy, 2003. to appear.
- [JVW90] 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.
- [Kau89] L.H. Kauffman. A Tutte polynomial for signed graphs. *Discrete Applied Mathematics*, 25:105–127, 1989.
- [Kau91] L.H. Kauffman. *Knots and Physics*. World Scientific, 1991.
- [Kau95] L.H. Kauffman, editor. *Knots and Applications*. World Scientific, 1995.
- [Kaw96] A. Kawauchi. *A Survey of Knot Theory*. Birkhäuser Verlag, 1996.
- [Lad77] R.E. Ladner. Application of model theoretic games to discrete linear orders and finite automata. *Information and Control*, 33:281–303, 1977.
- [Lic88] W.B.R. Lickorish. Polynomials for links. *Bull. London Math. Soc.*, 20:558–588, 1988.
- [LM03] M. Lotz and J.A. Makowsky. On the algebraic complexity of some families of coloured Tutte polynomials. *Advances in Applied Mathematics*, xx:xx–yy, 2003. in print.

- [Mak97] J.A. Makowsky. Invariant definability. In *Computational Logic and Proof Theory*, volume 1289 of *Lecture Notes in Computer Science*, pages 186–202. Springer Verlag, 1997.
- [Mak01a] J.A. Makowsky. Algorithmic uses of the Feferman-Vaught theorem. Lecture delivered at the Tarski Centenary Conference, Warsaw, May 2001, paper submitted to APAL in January 2003, special issue of the conference, 2001.
- [Mak01b] 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.
- [Mig99] J. Mighton. *Knot Theory on Bipartite Graphs*. PhD thesis, Department of Mathematics, University of Toronto, Toronto, Canada, 1999.
- [MM00] 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.
- [MM03a] J.A. Makowsky and J.P. Mariño. Farrell polynomials on graphs of bounded treewidth. *Advances in Applied Mathematics*, xx:xx–yy, 2003. in print.
- [MM03b] J.A. Makowsky and J.P. Mariño. The parametrized complexity of knot polynomials. *Journal of Computer and System Sciences*, xx:xx–yy, 2003. in print.
- [Neg87] S. Negami. Polynomial invariants of graphs. *Transaction of the American Mathematical Society*, 299:601–622, 1987.
- [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.
- [OW92] J.G. Oxley and D.J.A. Welsh. Tutte polynomials computable in polynomial time. *Discrete Mathematics*, 109:185–192, 1992.
- [Oxl92] J.G. Oxley. *Matroid Theory*. Oxford University Press, 1992.
- [Prz01] J. Przytycki. Algebraic tangles. In XXXX, editor, *Encyclopedia of Mathematics*, pages xx–yy. Kluwer, 2001.
- [PT01] J. Przytycki and T. Tsukamoto. The fourth skein module and the Montesinos-Nakanishi conjecture for 3-algebraic links. *J. Knot Theory Ramifications*, 10(7):959–982, 2001.
- [Sey80] P. D. Seymour. Decomposition of regular matroids. *Journal of Combinatorial Theory, Series B*, 28:305–359, 1980.

- [Sos99] A. Sossinsky. *Noeuds: Genèse d'une théorie mathématique*. Seuil, Paris, 1999.
- [Thi87] M.B. Thistlethwaite. A spanning tree expansion of the Jones polynomial. *Topology*, 26:297–309, 1987.
- [Thi88] M.B. Thistlethwaite. Kauffman's polynomial and alternating links. *Topology*, 27:297–309, 1988.
- [Tra] L. Traldi. A note on the colored tutte polynomial of a graph of bounded tree width. Preprint, January 2003.
- [Tru92] K. Truemper. *Matroid Decomposition*. Academic Press, 1992.
- [Tut54] W.T. Tutte. A contribution to the theory of chromatic polynomials. *Canadian Journal of Mathematics*, 6:80–91, 1954.
- [Wel76] D.J.A. Welsh. *Matroid Theory*, volume 8 of *London Mathematical Society Monographs*. Academic Press, 1976.
- [Wel93] D.J.A. Welsh. *Complexity: Knots, Colourings and Counting*, volume 186 of *London Mathematical Society Lecture Notes Series*. Cambridge University Press, 1993.
- [Wel95] D.J.A. Welsh. Randomized approximation schemes for Tutte-Grothedieck invariants. In D. Aldous et. al., editor, *Discrete Probability and Algorithms*, volume 72 of *IMA Volumes in Mathematics and Applications*, pages 133–148. Springer, 1995.
- [Wel97] D.J.A. Welsh. Approximate counting. In R.A. Bailey, editor, *Surveys in Combinatorics, 1997*, volume 241 of *London Mathematical Society Lecture Notes Series*, pages 287–324. Cambridge University Press, 1997.