Journal of Applied and Computational Topology (2018) 2:33–53
https://doi.org/10.1007/s41468-018-0016-2

# Algorithms and complexity for Turaev–Viro invariants

Benjamin A. Burton[1] · Clément Maria[1] · Jonathan Spreer[2]

Received: 26 September 2017 / Accepted: 3 August 2018 / Published online: 22 August 2018

## Abstract

The Turaev–Viro invariants are a powerful family of topological invariants for distinguishing between different 3-manifolds. They are invaluable for mathematical software, but current algorithms to compute them require exponential time. The invariants are parameterized by an integer $r \geq 3$. We resolve the question of complexity for $r = 3$ and $r = 4$, giving simple proofs that the Turaev–Viro invariants for $r = 3$ can be computed in polynomial time, but computing the invariant for $r = 4$ is #P-hard. Moreover, we describe an algorithm for arbitrary $r$, which is fixed-parameter tractable with respect to the treewidth of the dual graph of the input triangulation. We show through concrete implementation and experimentation that this algorithm is practical—and indeed preferable—to the prior state of the art for real computation. The algorithm generalises to every triangulated 3-manifold invariant defined from tensor network contraction.

**Keywords** Computational topology · 3-manifolds · Invariants · #P-hardness · Parameterized complexity

**Mathematics Subject Classification** 57M27 · 57Q15 · 68Q17

This work is supported by the Australian Research Council (ARC), Projects DP1094516 and DP140104246.

✉ Jonathan Spreer
jonathan.spreer@fu-berlin.de

Benjamin A. Burton
bab@maths.uq.edu.au

Clément Maria
clement.maria@inria.fr

[1] School of Mathematics and Physics, The University of Queensland, Brisbane, QLD 4072, Australia

[2] Institut für Mathematik, Freie Universität Berlin, Arnimallee 2, 14195 Berlin, Germany

## 1 Introduction

In geometric topology, testing homeomorphism (topological equivalence) is a fundamental algorithmic problem. However, beyond dimension two it is remarkably difficult. In dimension three—the focus of this paper—an algorithm follows from Perelman's proof of the geometrization conjecture (Kleiner and Lott 2008), but it is extremely intricate, its complexity is unknown and it has never been implemented.

As a result, practitioners in computational topology rely on simpler *topological invariants*—computable properties of a topological space that can be used to tell different spaces apart. One of the best known invariants is homology, but for 3-manifolds (the 3-dimensional generalisation of surfaces) this is weak: there are many topologically different 3-manifolds that homology cannot distinguish. Therefore major software packages in 3-manifold topology rely on invariants that are stronger but more difficult to compute.

In the discrete setting, there exist large numbers of invariants for 3-manifolds defined with the help of triangulations and *state sums*, i.e., large sums of weights associated to colourings of the edges of a triangulation (Barrett and Westbury 1996; Turaev 2010). Among the most commonly used such *state sum invariants* are the *Turaev–Viro invariants* (Turaev and Viro 1992).

They can be computed using the major software packages *Regina* (Burton et al. 1999) and *Manifold Recogniser* (Matveev 2003; Matveev et al. 2012), and they play a key role in developing census databases, which are analogous to the well-known dictionaries of knots (Burton 2007; Matveev 2003). However, current implementations (Burton et al. 1999; Matveev et al. 2012) are based on backtracking searches, and require exponential time.[1]

The purpose of this paper is threefold: (1) to introduce the Turaev–Viro invariants (and their close relatives) to the wider computational topology community; (2) to understand the complexity of computing these invariants; and (3) to develop new algorithms that are suitable for practical software.

The Turaev–Viro invariants are parameterized by an integer $r$, $r \geq 3$, and additionally depend on a $2r$-th root of unity $\zeta = e^{i\pi s/r}$, $0 < s < 2r$ coprime to $r$; we denote these invariants by $\mathrm{TV}_r$. A typical algorithm for computing $\mathrm{TV}_r$ takes as input a triangulated 3-manifold, composed of $n$ tetrahedra attached along their triangular faces; we use $n$ to indicate the input size. For all known algorithms, the difficulty of computing $\mathrm{TV}_r$ grows significantly as $r$ increases (but in contrast, the difficulty is essentially independent of $\zeta$).

Our main results are as follows.

- Kauffman and Lins (1991) state that for $r = 3, 4$ one can compute $\mathrm{TV}_r$ via "simple and efficient methods of linear algebra", but they give no details on either the algorithms or the complexity. We point out that in fact the situations for $r = 3$ and $r = 4$ are markedly different: computing $\mathrm{TV}_r$ for orientable manifolds and $r = 3$ can be done in polynomial time, but for $r = 4$ it is #P-hard.

[1] For these reasons, this work focuses on the family of Turaev–Viro invariants, although our methods can be applied to any other state sum invariants with very little effort. This is briefly explained in Sect. 5.

- We give an explicit algorithm for computing $\mathrm{TV}_r$ for general $r$ that is fixed-parameter tractable (FPT) in the treewidth of the dual graph of the input triangulation. Specifically, for any fixed $r$ and any class of input triangulations whose dual graphs have bounded treewidth, the algorithm has running time linear in $n$. Furthermore, we show through comprehensive experimentation that this algorithm is *practical*. That is, we implement it in the open-source software package *Regina* (Burton et al. 1999), run it through exhaustive census databases, and find that this new FPT algorithm is comparable to—and often significantly faster than—the prior backtracking algorithm.
- We sketch how this algorithm naturally generalises to all invariants which can be expressed as a state sum over tensor network contractions. In particular, we describe FPT-algorithms for all invariants arising from the very general Turaev–Viro–Barrett–Westbury construction for triangulated 3-manifolds (Barrett and Westbury 1996).
- We give a new geometric interpretation of the formula for $\mathrm{TV}_r$, based on systems of "normal arcs" in triangles. This generalises earlier observations of Kauffman and Lins for $r = 3$ based on embedded surfaces (Kauffman and Lins 1991), and offers an interesting potential for future algorithms based on Hilbert bases.

Proving #P-hardness for $r = 4$ and polynomial-time solvability for $r = 3$ is simple: the algorithm for $r = 3$ derives from a known homological formulation (Matveev 2003), and the result for $r = 4$ is essentially Kirby and Melvin's NP-hardness proof for the more complex Witten–Reshetikhin–Turaev invariants (Kirby and Melvin 2004). The latter result complements other hardness results for the Turaev–Viro invariants in the literature: It is known that approximating Turaev–Viro invariants for triangulated manifold is both #P-hard (Alagic and Lo 2017, Theorem 5.1) and universal for *quantum* computations (Alagic et al. 2010) for infinitely many values of $r$.

The FPT algorithm for general $r$ is significant in that it is not just theoretical, but also practical—and indeed *preferable*—for real software. It was previously known that computing general tensor network contractions is FPT in the treewidth, see Markov (2008) where this result is used to simulate quantum computations efficiently. Moreover, a specialised FPT result for $\mathrm{TV}_r$ is explicitly mentioned in Burton and Downey (2014). However, these prior result are purely theoretical, and lead to infeasibly large constants in the running time if translated to a concrete algorithm. More generally, FPT algorithms do not always translate well into practical software tools, and this paper is significant in giving the first demonstrably practical FPT algorithm in 3-manifold topology. The FPT algorithm is implemented in the open-source software package *Regina* (Burton et al. 1999).

An extended abstract of this work has appeared in the proceedings of ICALP 2015 (Downey et al. 2015).

## 2 Preliminaries

Let $M$ be a closed 3-manifold. A *generalised triangulation* of $M$ is a collection of $n$ abstract tetrahedra $\Delta_1, \ldots, \Delta_n$ equipped with affine maps that identify (or "glue together") their $4n$ triangular faces in pairs, so that the underlying topological space is homeomorphic to $M$.

In particular, as a consequence of the face identifications, it is possible that several vertices of the same tetrahedron may be identified together (and likewise for edges and triangles). Indeed, it is common in practical applications to have a *one-vertex triangulation*, in which all vertices of all tetrahedra are identified to a common point. In general, the $4n$ tetrahedron vertices are partitioned into equivalence classes according to how they are identified together; we refer to each such equivalence class as a single *vertex of the triangulation*, and likewise for edges and triangles.

Generalised triangulations are widely used across major 3-manifold software packages. They are (as the name suggests) more general than simplicial complexes, which allows them to express a rich variety of different 3-manifolds using very few tetrahedra. For instance, with just $n \leq 11$ tetrahedra one can create 13,400 distinct prime orientable 3-manifolds (Burton 2011; Matveev 2003).

### 2.1 The Turaev–Viro invariants

Let $\mathfrak{T}$ be a generalised triangulation of a closed 3-manifold $M$, and let $r \geq 3$ be an integer. We define the Turaev–Viro invariant $\mathrm{TV}_r(\mathfrak{T})$ as follows.

Let $V$, $E$, $F$ and $T$ denote the set of vertices, edges, triangles and tetrahedra respectively of the triangulation $\mathfrak{T}$. Let $I = \{0, 1/2, 1, 3/2, \ldots, (r-2)/2\}$; note that $|I| = r - 1$. We define a *colouring* of $\mathfrak{T}$ to be a map $\theta\colon E \to I$; that is, $\theta$ "colours" each edge of $\mathfrak{T}$ with an element of $I$. A colouring $\theta$ is *admissible* if, for each triangle of $\mathfrak{T}$, the three edges $e_1$, $e_2$, and $e_3$ bounding the triangle satisfy:

- the *parity condition* $\theta(e_1) + \theta(e_2) + \theta(e_3) \in \mathbb{Z}$;
- the *triangle inequalities* $\theta(e_1) \leq \theta(e_2) + \theta(e_3)$, $\theta(e_2) \leq \theta(e_1) + \theta(e_3)$, and $\theta(e_3) \leq \theta(e_1) + \theta(e_2)$; and
- the *upper bound constraint* $\theta(e_1) + \theta(e_2) + \theta(e_3) \leq r - 2$.

More generally, we refer to any triple $(i, j, k) \in I \times I \times I$ satisfying these three conditions as an *admissible triple* of colours.

For each admissible colouring $\theta$ and for each vertex $v \in V$, edge $e \in E$, triangle $f \in F$ or tetrahedron $t \in T$, we define *weights* $|v|_\theta$, $|e|_\theta$, $|f|_\theta$, $|t|_\theta \in \mathbb{C}$. The precise definition of these weights is rather involved and explained in detail below.

Our notation differs slightly from Turaev and Viro (1992); most notably, Turaev and Viro do not consider triangle weights $|f|_\theta$, but instead incorporate an additional factor of $|f|_\theta^{1/2}$ into each tetrahedron weight $|t|_\theta$ and $|t'|_\theta$ for the two tetrahedra $t$ and $t'$ containing $f$. This choice of notation simplifies the notation and avoids unnecessary (but harmless) ambiguities when taking square roots.

Let $\zeta = e^{i\pi s/r} \in \mathbb{C}$, $0 < s < 2r$, $\gcd(r, s) = 1$. Note that this implies that $\zeta$ is a $(2r)$th root of unity, and that $\zeta^2$ is a *primitive* $r$th root of unity; that is, $(\zeta^2)^k \neq 1$ for $k = 1, \ldots, r-1$. For each positive integer $i$, we define $[i] = (\zeta^i - \zeta^{-i})/(\zeta - \zeta^{-1})$ and, as a special case, $[0] = 1$. We next define the "bracket factorial" $[i]! = [i]\,[i-1]\ldots[0]$. Note that $[r] = 0$, and thus $[i]! = 0$ for all $i \geq r$.

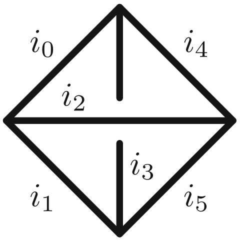


**Fig. 1** Edge colours of a tetrahedron

We give every vertex constant weight

$$|v|_\theta = \frac{\left|\zeta - \zeta^{-1}\right|^2}{2r},$$

and to each edge $e$ of colour $i \in I$ (i.e., for which $\theta(e) = i$) we give the weight

$$|e|_\theta = (-1)^{2i} \cdot [2i + 1].$$

A triangle $f$ whose three edges have colours $i, j, k \in I$ is assigned the weight

$$|f|_\theta = (-1)^{i+j+k} \cdot \frac{[i+j-k]! \cdot [i+k-j]! \cdot [j+k-i]!}{[i+j+k+1]!}.$$

Note that the parity condition and triangle inequalities ensure that the argument inside each bracket factorial is a non-negative integer.

Finally, let $t$ be a tetrahedron with edges colours $i_0, i_1, i_2, i_3, i_4, i_5$ as indicated in Fig. 1. In particular, the four triangles surrounding $t$ have colours $(i_0, i_1, i_3)$, $(i_0, i_2, i_4)$, $(i_1, i_2, i_5)$ and $(i_3, i_4, i_5)$, and the three pairs of opposite edges have colours $(i_0, i_5)$, $(i_1, i_4)$ and $(i_2, i_3)$. We define

$$\begin{aligned}\tau_\phi(t,z) &= [z - i_0 - i_1 - i_3]! \cdot [z - i_0 - i_2 - i_4]! \cdot [z - i_1 - i_2 - i_5]! \\ &\quad \cdot [z - i_3 - i_4 - i_5]!, \\ \kappa_\phi(t,z) &= [i_0 + i_1 + i_4 + i_5 - z]! \cdot [i_0 + i_2 + i_3 + i_5 - z]! \cdot [i_1 + i_2 + i_3 + i_4 - z]!\end{aligned}$$

for all integers $z$ such that the bracket factorials above all have non-negative arguments; equivalently, for all integers $z$ in the range $z^- \le z \le z^+$ with

$$\begin{aligned}z^- &= \max\{i_0 + i_1 + i_3,\ i_0 + i_2 + i_4,\ i_1 + i_2 + i_5,\ i_3 + i_4 + i_5\}; \\ z^+ &= \min\{i_0 + i_1 + i_4 + i_5,\ i_0 + i_2 + i_3 + i_5,\ i_1 + i_2 + i_3 + i_4\}.\end{aligned}$$

Note that, as before, the parity condition ensures that the argument inside each bracket factorial above is an integer. We then declare the weight of tetrahedron $t$ to be

$$|t|_\phi = \sum_{z^- \le z \le z^+} \frac{(-1)^z \cdot [z+1]!}{\tau_\phi(t,z) \cdot \kappa_\phi(t,z)},$$

Note that all weights are polynomials in $\zeta$ with rational coefficients.

Using these weights, we define the *weight of the colouring* to be

$$|\mathfrak{T}|_\theta = \prod_{v \in V} |v|_\theta \times \prod_{e \in E} |e|_\theta \times \prod_{f \in F} |f|_\theta \times \prod_{t \in T} |t|_\theta, \tag{1}$$

and the Turaev–Viro invariant to be the sum over all admissible colourings

$$\mathrm{TV}_r(\mathfrak{T}) = \sum_{\theta \text{ admissible}} |\mathfrak{T}|_\theta. \tag{2}$$

Turaev and Viro (1992) show that $\mathrm{TV}_r(\mathfrak{T})$ is indeed an invariant of the manifold; that is, if $\mathfrak{T}$ and $\mathfrak{T}'$ are generalised triangulations of the same closed 3-manifold $M$, then $\mathrm{TV}_r(\mathfrak{T}) = \mathrm{TV}_r(\mathfrak{T}')$ for all $r$. Although $\mathrm{TV}_r(\mathfrak{T})$ is defined on the complex numbers $\mathbb{C}$, it always takes a real value (it is equal to the square of the modulus of a Witten–Reshetikhin–Turaev invariant) (Walker 1991).

In Sect. 5 we describe the philosophy of the Turaev–Viro–Barrett–Westbury construction of topological invariants (Barrett and Westbury 1996), which substantially generalises the Turaev–Viro model.

### 2.2 Treewidth and parameterized complexity

Throughout this paper we always refer to *nodes* and *arcs* of graphs, to clearly distinguish these from the *vertices* and *edges* of triangulations.

Robertson and Seymour (1986) introduced the concept of the *treewidth* of a graph, which now plays a major role in parameterized complexity. Here, we adapt this concept to triangulations in a straightforward way.

**Definition 1** Let $\mathfrak{T}$ be a generalised triangulation of a 3-manifold, and let $T$ be the set of tetrahedra in $\mathfrak{T}$. A *tree decomposition* $(X, \{B_\tau\})$ of $\mathfrak{T}$ consists of a tree $X$ and *bags* $B_\tau \subseteq T$ for each node $\tau$ of $X$, for which:

- each tetrahedron $t \in T$ belongs to some bag $B_\tau$;
- if a face of some tetrahedron $t_1 \in T$ is identified with a face of some other tetrahedron $t_2 \in T$, then there exists a bag $B_\tau$ with $t_1, t_2 \in B_\tau$;
- for each tetrahedron $t \in T$, the bags containing $t$ correspond to a connected subtree of $X$.

The *width* of this tree decomposition is defined as $\max |B_\tau| - 1$. The *treewidth of* $\mathfrak{T}$, denoted $\mathrm{tw}(\mathfrak{T})$, is the smallest width of any tree decomposition of $\mathfrak{T}$.

The relationship between this definition and the classical graph-theoretical notion of treewidth is simple: $\mathrm{tw}(\mathfrak{T})$ is the treewidth of the *dual graph* of $\mathfrak{T}$, the 4-valent multigraph whose nodes correspond to tetrahedra of $\mathfrak{T}$ and whose arcs represent pairs of tetrahedron faces that are identified together.

Figure 2 shows the dual graph of a 9-tetrahedra triangulation of a 3-manifold, along with a possible tree decomposition. The largest bags have size three, and so the width of this tree decomposition is $3 - 1 = 2$.

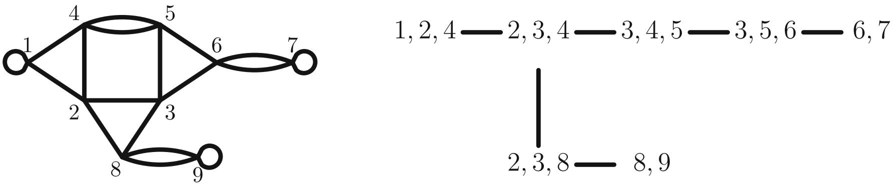


**Fig. 2** The dual graph and a tree decomposition of a 3-manifold triangulation

**Definition 2** A *nice tree decomposition* of a generalised triangulation $\mathfrak{T}$ is a tree decomposition $(X, \{B_\tau\})$ of $\mathfrak{T}$ whose underlying tree $X$ is rooted, and where:

- The bag $B_\rho$ at the root of the tree is empty ($B_\rho$ is called the *root bag*);
- If a bag $B_\tau$ has no children, then $|B_\tau| = 1$ (such a $B_\tau$ is called a *leaf bag*);
- If a bag $B_\tau$ has two children $B_\sigma$ and $B_\mu$, then $B_\tau = B_\sigma = B_\mu$ (such a $B_\tau$ is called a *join bag*);
- Every other bag $B_\tau$ has precisely one child $B_\sigma$, and either:
  - $|B_\tau| = |B_\sigma| + 1$ and $B_\tau \supset B_\sigma$ (such a $B_\tau$ is called an *introduce bag*), or
  - $|B_\tau| = |B_\sigma| - 1$ and $B_\tau \subset B_\sigma$ (such a $B_\tau$ is called a *forget bag*).

Given a tree decomposition of a triangulation $\mathfrak{T}$ of width $k$ and $O(n)$ bags, we can convert this in $O(n)$ time into a *nice* tree decomposition of $\mathfrak{T}$ that also has width $k$ and $O(n)$ bags (Kloks 1994).

## 3 Algorithms for computing Turaev–Viro invariants

All of the algorithms in this paper use exact arithmetic. This is crucial if we wish to avoid floating-point numerical instability, since computing $\mathrm{TV}_r$ may involve exponentially many arithmetic operations.

We briefly describe how this exact arithmetic works. Since all weights in the definition of $\mathrm{TV}_r$ are rational polynomials in $\zeta = e^{i\pi s/r}$, $0 < s < 2r$, all arithmetic operations remain within the rational field extension $\mathbb{Q}(\zeta)$. If $\zeta$ is a primitive $n$th root of unity then this field extension is called the *nth cyclotomic field*. This in turn is isomorphic to the polynomial field $\mathbb{Q}[X]/\Phi_n(X)$, where $\Phi_n(X)$ is the $n$th cyclotomic polynomial with degree $\varphi(n)$ (Euler's totient function). Therefore we can implement exact arithmetic using degree $\varphi(n)$ polynomials over $\mathbb{Q}$.

If $r$ is odd and $s$ is even, then $\zeta$ is a primitive $r$th root of unity, and $\mathbb{Q}(\zeta) \cong \mathbb{Q}[X]/\Phi_r(X)$. Otherwise $\zeta$ is a primitive $(2r)$th root of unity, and $\mathbb{Q}(\zeta) \cong \mathbb{Q}[X]/\Phi_{2r}(X)$. In this paper we give our complexity results in terms of arithmetic operations in $\mathbb{Q}(\zeta)$.

Let $\zeta$ be an $n$-root of unity and $\mathbb{Q}(\zeta)$ be the $n$th cyclotomic field. We represent elements of $\mathbb{Q}(\zeta)$ by polynomials of degree at most $\varphi(n)$, with rational coefficients, using the isomorphism $\mathbb{Q}(\zeta) \cong \mathbb{Q}[X]/\Phi_n(X)$. Asymptotically, the Euler totient function satisfies $\varphi(n) = \Theta(n)$. Additions of two polynomials of degree at most $n$ are performed in $O(n)$ operations in $\mathbb{Q}$, and multiplications and divisions are performed in $O(M(n))$ operations in $\mathbb{Q}$, with $M(n) = O(n \log n \log\log n)$ (Cantor et al. 1991).

Hence, for fixed $r$, Turaev–Viro invariants can be computed in $O(N(r) \cdot r \log r \log\log r)$ operations in $\mathbb{Q}$ using exact arithmetic over cyclotomic fields, where $N(r)$ denotes the number of arithmetic operations needed to compute $\mathrm{TV}_r$.

### 3.1 The backtracking algorithm for computing $\mathrm{TV}_r$

There is a straightforward but slow algorithm to compute $\mathrm{TV}_r$ for arbitrary $r$. The core idea is to use a backtracking algorithm to enumerate all admissible colourings of edges, and compute and sum their weights. Both major software packages that compute Turaev–Viro invariants—the *Manifold Recogniser* (Matveev et al. 2012) and *Regina* (Burton et al. 1999)—currently employ optimised variants of this.

Let $\mathfrak{T}$ be a 3-manifold triangulation, with $\ell$ edges $e_1, \ldots, e_\ell$. A simple Euler characteristic argument gives $\ell = n + v$ where $n$ is the number of tetrahedra and $v$ is the number of vertices in $\mathfrak{T}$. Therefore $\ell \in \Theta(n)$.

To enumerate colourings, since each edge admits $r - 1$ possible colours, the backtracking algorithm traverses a search tree of $O((r-1)^\ell)$ nodes: a node at depth $i$ corresponds to a partial colouring of the edges $e_1, \ldots, e_i$, and each node has degree $r-1$ (one edge per colour). Each leaf on the tree represents a (possibly not admissible) colouring of all the edges. At each node we maintain the weight of the current partial colouring, and updates this weight as we traverse the tree. If we reach a leaf whose colouring is admissible, we add this weight to our total.

**Lemma 1** *If we sort the edges $e_1, \ldots, e_\ell$ by decreasing degree, the backtracking algorithm terminates in $O((r-1)^\ell)$ arithmetic operations in $\mathbb{Q}(\zeta)$.*

***Proof*** The proof is simple. The main complication is to ensure that updating the weight of the current partial colouring takes amortised constant time. For this we use Chebyshev's inequality, plus the observation that the average edge degree is $\leq 6$.

In more detail, suppose that the edges $e_1, \ldots, e_\ell$ are ordered by decreasing degree. Let $\deg(e_i)$ be the degree of edge $e_i$. Changing the colour of $e_i$ affects the colours of the $\deg(e_i)$ triangles and $\deg(e_i)$ tetrahedra containing $e_i$. Hence the update of the current partial colouring weight is performed in $O(\deg(e_i))$ arithmetic operations in $\mathbb{Q}(\zeta)$. The total number of arithmetic operations performed by the algorithm is consequently $O(\sum_i (r-1)^i \deg(e_i))$. Following an Euler characteristic argument, a triangulation of a closed 3-manifold with $\ell$ edges and $v$ vertices has $n = \ell - v$ tetrahedra and, consequently, the average degree of an edge is $6(\ell - v)/\ell$ and thus constant. Considering that the sequence $((r-1)^i)_i$ is increasing and $\deg e_i$ is decreasing, we conclude using Chebyshev's sum inequality that

$$O\left(\sum_i (r-1)^i \deg(e_i)\right) = O\left(\sum_i (r-1)^i\right) = O((r-1)^\ell).$$

To obtain a bound in the number of tetrahedra $n$, we note that a closed and connected 3-manifold triangulation with $n > 2$ tetrahedra must have $v \leq n + 1$ vertices. Combined with $n = \ell - v$ above, we have a worst-case running time of $O((r-1)^{2n+1})$ arithmetic operations in $\mathbb{Q}(\zeta)$.

### 3.2 A polynomial-time algorithm for $r = 3$

Throughout this section, $\mathfrak{T}$ denotes an $n$-tetrahedra triangulation of an orientable 3-manifold $M$.

***Remark 1*** We introduce homology with coefficients in the field $\mathbb{Z}_2$. A generalised triangulation $\mathfrak{T}$, after gluing, contains a set of vertices (dimension 0), edges (dimension 1), triangles (dimension 2) and tetrahedra (dimension 3) with incidence relations. The group of $d$-chains of $\mathfrak{T}$, $d \in \{0, \dots, 3\}$, denoted by $\mathbf{C}_d(\mathfrak{T}, \mathbb{Z}_2)$ is the group of formal sums of $d$-simplices with $\mathbb{Z}_2$ coefficients. The $d$th *boundary operator* is a linear operator $\partial_d : \mathbf{C}_d(\mathfrak{T}, \mathbb{Z}_2) \to \mathbf{C}_{d-1}(\mathfrak{T}, \mathbb{Z}_2)$ that assigns to a $d$-face the alternate sum of its boundary faces. The kernel of $\partial_d$, denoted by $\mathbf{Z}_d(\mathfrak{T}, \mathbb{Z}_2)$, is the group of $d$-*cycles* and the image of $\partial_d$, denoted by $\mathbf{B}_{d-1}(\mathfrak{T}, \mathbb{Z}_2)$, is the group of $(d-1)$-*boundaries*. The fundamental property of homology is that $\partial_d \circ \partial_{d+1} = 0$, and we define the $d$th homology group $\mathbf{H}_d(\mathfrak{T}, \mathbb{Z}_2)$ of $\mathfrak{T}$ to be the quotient $\mathbf{H}_d(\mathfrak{T}, \mathbb{Z}_2) = \mathbf{Z}_d(\mathfrak{T}, \mathbb{Z}_2)/\mathbf{B}_d(\mathfrak{T}, \mathbb{Z}_2)$. It is a finite dimensional $\mathbb{Z}_2$-vector space and we denote its dimension by $\beta_d(\mathfrak{T}, \mathbb{Z}_2)$.

For a more thorough introduction into homology theory, see Munkres (1984).

There are two Turaev–Viro invariants for $r = 3$, at $\zeta = e^{i\pi/3}$ and at $\zeta = e^{i\pi 2/3}$. In both cases the value of $\mathrm{TV}_3(\mathfrak{T})$ is closely related to $\mathbf{H}_2(M, \mathbb{Z}_2)$, the 2- dimensional homology group of $M$ with $\mathbb{Z}_2$ coefficients. $\mathbf{H}_2(M, \mathbb{Z}_2)$ is a $\mathbb{Z}_2$-vector space whose dimension is the second Betti number $\beta_2(M, \mathbb{Z}_2)$. Its elements are (for our purposes) equivalence classes of 2-cycles, called *homology classes*, which can be represented by 2-dimensional triangulated surfaces $S$ embedded in $\mathfrak{T}$.

The *Euler characteristic* of a triangulated surface $S$, denoted by $\chi(S)$, is $\chi(S) = v - e + f$, where $v$, $e$ and $f$ denote the number of vertices, edges and triangles of $S$ respectively. We define the Euler characteristic $\chi(c)$ of a 2-cycle $c$ to be the Euler characteristic of the embedded surface it represents. Given $\mathfrak{T}$, the dimension $\beta_2(M, \mathbb{Z}_2)$ of $\mathbf{H}_2(M, \mathbb{Z}_2)$ may be computed in $O(\mathrm{poly}(n))$ operations.

The following result is well known (Matveev 2003):

**Proposition 1** *Let $M$ be a closed orientable 3-manifold. If $M$ contains a 2-cycle with odd Euler characteristic, then $\mathrm{TV}_3(M) =$ the order of $\mathbf{H}_2(M, \mathbb{Z}_2)$ at $\zeta = e^{i\pi 2/3}$ and $\mathrm{TV}_3(M) = 0$ at $\zeta = e^{i\pi/3}$. Otherwise we have $\mathrm{TV}_3(M) =$ the order of $\mathbf{H}_2(M, \mathbb{Z}_2)$ for both $\zeta = e^{i\pi/3}$ and $\zeta = e^{i\pi 2/3}$.*

Consequently $\mathrm{TV}_3(M) = 2^{\beta_2(M,\mathbb{Z}_2)}$ at $\zeta = e^{i\pi 2/3}$, and one can compute $\mathrm{TV}_3(M)$ in this case in polynomial time. The parity of the Euler characteristic of 2-cycles does not change within a homology class; moreover, given two 2-cycles $c$ and $c'$, $\chi(c + c') \equiv \chi(c) + \chi(c') \bmod 2$. Consequently, for $\zeta = e^{i\pi/3}$, one can check whether $\mathrm{TV}_3(M) = 0$ or $\mathrm{TV}_3(M) =$ the order of $\mathbf{H}_2(M, \mathbb{Z}_2)$ by computing the Euler characteristic of a cycle in each of the $\beta_2(M, \mathbb{Z}_2)$ homology classes that generate $\mathbf{H}_2(M, \mathbb{Z}_2)$. Because $\beta_2(M, \mathbb{Z}_2) = O(n)$, this leads to a polynomial time algorithm also.

### 3.3 #P-hardness of $\mathrm{TV}_4$

The complexity class #P is a function class that counts accepting paths of a non-deterministic Turing machine (Valiant 1979). Informally, given an NP decision problem $C$ asking for the existence of a solution, its #P analogue $\#C$ is a counting problem asking for the number of such solutions. A problem is *#P-hard* if every problem in #P polynomially reduces to it. For example, the problem $\#3SAT$, which asks for the number of satisfying assignments of a $3CNF$ formula, is #P-hard.

Naturally, counting problems are "harder" than their decision counterpart, and so #P-hard problems are at least as hard as NP-complete problems—specifically, #P-complete problems are as hard as any problem in the polynomial hierarchy (Toda 1991). Hence proving #P-hardness is a strong complexity statement.

Kirby and Melvin (2004) prove that computing the Witten–Reshetikhin–Turaev invariant $\tau_r$ is #P-hard for $r = 4$. This invariant $\tau_r$ is a more complex 3-manifold invariant which is closely linked to the Turaev–Viro invariant $\mathrm{TV}_r$ at $\zeta = e^{i\pi/r}$ by the formula $\mathrm{TV}_r(M) = |\tau_r(M)|^2$. Although computing $\mathrm{TV}_r$ is "easier" than computing $\tau_r$, we can use the Kirby-Melvin hardness proof to fit our purposes.

To prove their result, Kirby and Melvin reduce the problem of counting the zeros of a cubic form to the computation of $\tau_4$. Given a cubic form

$$c(x_1, \ldots, x_n) = \sum_i c_i\ x_i + \sum_{i,j} c_{ij}\ x_i x_j + \sum_{i,j,k} c_{ijk}\ x_i x_j x_k$$

in $n$ variables over $\mathbb{Z}_2$ and with $\#c$ zeros, they define a triangulation of a 3-manifold $M_c$ with $O(\mathrm{poly}(n))$ tetrahedra satisfying $\tau_4(M_c) = 2\#c - 2^n$ and hence $\mathrm{TV}_r = (2\#c - 2^n)^2$.

Consequently, counting the zeros of $c(x_1, \ldots, x_n)$ reduces to computing $\tau_4(M_c)$, and so computing $\mathrm{TV}_4$ determines $\#c$ up to a ± sign ambiguity (depending on whether or not $c$ admits more than half of the input as zeros).

Establishing the existence of a zero for a cubic form is an NP-complete problem, which implies that counting the number of zeros is #P-complete. Consequently, computing $\tau_4$ is #P-hard. Kirby and Melvin prove this claim by reducing $\#3SAT$ to the problem of counting the zeros of a cubic form; moreover, we observe that their construction ensures that this cubic form admits *more than half of its inputs as zeros*.

We recall the reduction of $\#3SAT$ to the problem of counting the number of zeros of a cubic form over $\mathbb{Z}_2$ found in Kirby and Melvin (2004). Given a $3CNF$ formula over variables $x_1, \ldots, x_n$:

$$C = C_1 \wedge \cdots \wedge C_m \quad \text{with } C_j = \overline{x}_{j_1} \vee \overline{x}_{j_2} \vee \overline{x}_{j_3}$$

and $\overline{x}_i$ is either $x_i$ or its negation $\neg x_i$ the problem $\#3SAT$ consists in counting the number of assignments of "true" and "false" to the variables $x_1, \ldots, x_n$ satisfying the formula.

To each form $C_j = \overline{x}_{j_1} \vee \overline{x}_{j_2} \vee \overline{x}_{j_3}$ we assign a cubic equation $q_j$ over $\mathbb{Z}_2$ by setting "true" = 0 and "false" = 1, and replacing $x_i$ by the variable $x_i$ and $\neg x_i$ by $(1 - x_i)$. For example, a form $(\neg x_i \wedge \neg x_j \wedge x_k)$ leads to the equation $(1 - x_i)(1 - x_j)x_k = 0$.

An assignment satisfies $C_j$ if and only if it cancels $q_j$, hence the number of solution to the system of equations $\{q_1(x_1, \ldots, x_n) = 0, \ldots, q_m(x_1, \ldots, x_n) = 0\}$ is equal to $\#c$, the number of satisfying assignments for $C$.

We turn each cubic equation into two quadratic equations by introducing a new variable $x_{ij}$ for each monomial $x_i x_j x_k$ of degree 3 and a new quadratic equation $x_{ij} - x_i x_j = 0$, and by replacing the product $x_i x_j x_k$ by $x_{ij} x_k$. We obtain a set of $m'$ equations $\{\overline{q_1}(x_1, \ldots, x_n) = 0, \ldots, \overline{q_{m'}}(x_1, \ldots, x_{n'}) = 0\}$ in $n'$ variables over $\mathbb{Z}_2$, with $m < m' \leq 2m$ and $n < n' \leq 2n$. The number of solutions of this system remains $\#c$.

Finally, define the following cubic form $c$ by introducing $m'$ extra variables $z_1, \ldots, z_{m'}$:

$$Q = \sum_{i=1}^{m'} z_i \overline{q_i}(x_1, \ldots, x_{n'}).$$

The number of zeros of $Q$ is equal to $2^{m'}\#c + 2^{m'-1}(2^{n'} - \#c) \geq \frac{1}{2}2^{n'+m'}$. Because $Q$ is defined on $n' + m'$ variables it admits more than half of its input as zeros. Finally, $\#3SAT$ reduces to counting the number of zeros of a cubic form which admits at least half of its input as zeros.

Thus the same reduction process as for $\tau_4$ applies for $\mathrm{TV}_4$, and so:

**Corollary 1** *Computing* $\mathrm{TV}_4$ *at* $\zeta = e^{i\pi/4}$ *is #P-hard.*

***Remark 2*** Very recently, a subset of the authors refined this complexity study of $\mathrm{TV}_4$ in Maria and Spreer (2017). More precisely, given an $n$-tetrahedra triangulation $\mathfrak{T}$ of a 3-manifold $M$, they present an algorithm to compute $\mathrm{TV}_4(M)$ in running time $O(2^{\beta_1(M,\mathbb{Z}_2)}n^3)$. Note that, in particular, the exponential part of the running time is independent of the triangulation $\mathfrak{T}$.

However, the methods to obtain this result fail to be applicable to $\mathrm{TV}_r$, $r > 4$.

## 4 A fixed-parameter tractable algorithm for Turaev–Viro invariants

Here, we present an explicit algorithm for computing the Turaev–Viro invariants $\mathrm{TV}_r$ of a 3-manifold triangulation, $r$ fixed, which is FPT in the treewidth of the dual graph of the input triangulation. As is common for treewidth-based methods, the algorithm involves dynamic programming over a tree decomposition $(X, \{B_\tau\})$. We first describe the data that we compute and store at each bag $B_\tau$, and then give the algorithm itself.

Our first step is to reorganise the formula for $\mathrm{TV}_r(\mathfrak{T})$ to be a product over tetrahedra only. This makes it easier to work with "partial colourings" corresponding to subsets of edges $\mathfrak{T}$.

**Definition 3** Let $\mathfrak{T}$ be a generalised triangulation of a 3-manifold, and let $V$, $E$, $F$ and $T$ denote the vertices, edges, triangles and tetrahedra of $\mathfrak{T}$ respectively. For each vertex $x \in V$, each edge $x \in E$ and each triangle $x \in F$, we arbitrarily choose some tetrahedron $\Delta(x)$ that contains $x$.

Now consider the definition of $\mathrm{TV}_r(\mathfrak{T})$. For each admissible colouring $\theta\colon\ E \to I$ and each tetrahedron $t \in T$, we define the *adjusted tetrahedron weight* $|t|'_\theta$:

$$|t|'_\theta = |t|_\theta \times \prod_{\substack{v\in V\\ \Delta(v)=t}} |v|_\theta \times \prod_{\substack{e\in E\\ \Delta(e)=t}} |e|_\theta \times \prod_{\substack{f\in F\\ \Delta(f)=t}} |f|_\theta.$$

It follows from Eq. (1) that the full weight of the colouring $\theta$ is just

$$|\mathfrak{T}|_\theta = \prod_{t\in T} |t|'_\theta.$$

**Notation 1** *Let $X$ be a rooted tree. For any non-root node $\tau$ of $X$, we denote the parent node of $\tau$ by $\hat{\tau}$. For any two nodes $\sigma, \tau$ of $X$, we write $\sigma \prec \tau$ if $\sigma$ is a descendant node of $\tau$.*

**Definition 4** Let $\mathfrak{T}$ be a generalised triangulation of a 3-manifold, and let $V$, $E$, $F$ and $T$ denote the vertices, edges, triangles and tetrahedra of $\mathfrak{T}$ respectively. Let $(X, \{B_\tau\})$ be a nice tree decomposition of $\mathfrak{T}$. For each node $\tau$ of the rooted tree $X$, we define the following sets:

- $T_\tau \subseteq T$ is the set of all tetrahedra that appear in bags beneath $\tau$ but not in the bag $B_\tau$ itself. More formally: $T_\tau = (\bigcup_{\sigma\prec\tau} B_\sigma)\backslash B_\tau$.
- $F_\tau \subseteq F$ is the set of all triangles that appear in some tetrahedron $t \in T_\tau$.
- $E_\tau \subseteq E$ is the set of all edges that appear in some tetrahedron $t \in T_\tau$.
- $E^*_\tau \subseteq E_\tau$ is the set of all edges that appear in some tetrahedron $t \in T_\tau$ and also some other tetrahedron $t' \notin T_\tau$; we refer to these as the *current edges* at node $\tau$.

We can make the following immediate observations:

**Lemma 2** *If $\tau$ is a leaf of the tree $X$, then we have $T_\tau = F_\tau = E_\tau = E^*_\tau = \emptyset$. If $\tau$ is the root of the tree $X$, then we have $T_\tau = T$, $F_\tau = F$, $E_\tau = E$, and $E^*_\tau = \emptyset$.*

The key idea is, at each node $\tau$ of the tree, to store explicit colours on the "current" edges $e \in E^*_\tau$ and to aggregate over all colours on the "finished" edges $e \in E_\tau\backslash E^*_\tau$. For this we need some further definitions and notation.

**Definition 5** Again let $\mathfrak{T}$ be a generalised triangulation of a 3-manifold, and let $(X, \{B_\tau\})$ be a nice tree decomposition of $\mathfrak{T}$. Fix some integer $r \geq 3$, and consider the set of colours $I = \{0, 1/2, 1, 3/2, \ldots, (r-2)/2\}$ as used in defining the Turaev–Viro invariants $\mathrm{TV}_r$.

Let $\tau$ be any node of $X$. We examine "partial colourings" that only assign colours to the edges in $E_\tau$ or $E^*_\tau$:

- Consider any colouring $\theta\colon\ E_\tau \to I$. We call $\theta$ *admissible* if, for each triangle in $F_\tau$, the three edges $e, f, g$ bounding the triangle yield an admissible triple $(\theta(e), \theta(f), \theta(g))$.
- Define $\Psi_\tau$ to be the set of all colourings $\psi\colon\ E^*_\tau \to I$ that can be extended to *any* admissible colouring $\theta\colon\ E_\tau \to I$.

- Consider any colouring $\psi \in \Psi_\tau$ (so $\psi \colon E^*_\tau \to I$). We define the "partial invariant"

$$\mathrm{TV}_r(\mathfrak{T}, \tau, \psi) = \sum_{\substack{\theta \text{ admissible} \\ \theta = \psi \text{ on } E^*_\tau}} \prod_{t \in T_\tau} |t|'_\theta.$$

Essentially, the partial invariant $\mathrm{TV}_r(\mathfrak{T}, \tau, \psi)$ considers all admissible ways $\theta$ of extending the colouring $\psi$ from the current edges $E^*_\tau$ to also include the "finished" edges in $E_\tau$, and then sums the partial weights $|t|'_\theta$ for all such extensions $\theta$ using only the tetrahedra in $T_\tau$.

We can now give a description of our algorithm for $\mathrm{TV}_r$.

**Algorithm 1** Let $\mathfrak{T}$ be a generalised triangulation of a 3-manifold. We compute $\mathrm{TV}_r(\mathfrak{T})$ for given $r$ and a given $2r$-th root of unity $\zeta$ as follows.

Build a *nice* tree decomposition $(X, \{B_\tau\})$ of $\mathfrak{T}$. Then work through each node $\tau$ of $X$ from the leaves of $X$ to the root, and compute $\Psi_\tau$ and $\mathrm{TV}_r(\mathfrak{T}, \tau, \psi)$ for each $\psi \in \Psi_\tau$ as follows.

1. If $\tau$ is a leaf bag, then $E^*_\tau = E_\tau = \emptyset$, $\Psi_\tau$ contains just the trivial colouring $\psi$ on $\emptyset$, and $\mathrm{TV}_r(\mathfrak{T}, \tau, \psi) = 1$.
2. If $\tau$ is some other introduce bag with child node $\sigma$, then $T_\tau = T_\sigma$. This means that $\Psi_\tau = \Psi_\sigma$, and for each $\psi \in \Psi_\tau$ we have $\mathrm{TV}_r(\mathfrak{T}, \tau, \psi) = \mathrm{TV}_r(\mathfrak{T}, \sigma, \psi)$.
3. If $\tau$ is a forget bag with child node $\sigma$, then $T_\tau = T_\sigma \cup \{t\}$ for the unique "forgotten" tetrahedron $t \in B_\tau \backslash B_\sigma$. Moreover, $E^*_\tau$ extends $E^*_\sigma$ by including the six edges of $t$ (if they were not already present).
   For each colouring $\psi \in \Psi_\sigma$, enumerate all possible ways of colouring the six edges of $t$ that are consistent with $\psi$ on any edges of $t$ that already appear in $E^*_\sigma$, and are admissible on the four triangular faces of $t$. Each such colouring on $t$ yields an extension $\psi' \colon E^*_\tau \to I$ of $\psi \colon E^*_\sigma \to I$. We include $\psi'$ in $\Psi_\tau$, and record the partial invariant $\mathrm{TV}_r(\mathfrak{T}, \tau, \psi') = \mathrm{TV}_r(\mathfrak{T}, \sigma, \psi)$.
4. If $\tau$ is a join bag with child nodes $\sigma_1, \sigma_2$, then $T_\tau$ is the disjoint union $T_{\sigma_1} \dot\cup T_{\sigma_2}$. Here $E^*_\tau$ is a subset of $E^*_{\sigma_1} \cup E^*_{\sigma_2}$.
   For each pair of colourings $\psi_1 \in \Psi_{\sigma_1}$ and $\psi_2 \in \Psi_{\sigma_2}$, if $\psi_1$ and $\psi_2$ agree on the common edges in $E^*_{\sigma_1} \cap E^*_{\sigma_2}$ then record the pair $(\psi_1, \psi_2)$.
   Each such pair yields a "combined colouring" in $\Psi_\tau$, which we denote by $\psi_1 \cdot \psi_2 \colon E^*_\tau \to I$; note that different pairs $(\psi_1, \psi_2)$ might yield the same colouring $\psi_1 \cdot \psi_2$ since some edges from $E^*_{\sigma_1} \cup E^*_{\sigma_2}$ might not appear in $E^*_\tau$. Then $\Psi_\tau$ consists of all such combined colourings $\psi_1 \cdot \psi_2$ from recorded pairs $(\psi_1, \psi_2)$. Moreover, for each combined colouring $\psi \in \Psi_\tau$ we compute the partial invariant $\mathrm{TV}_r(\mathfrak{T}, \tau, \psi)$ by aggregating over all duplicates:

$$\mathrm{TV}_r(\mathfrak{T}, \tau, \psi) = \sum_{\substack{(\psi_1, \psi_2) \text{ recorded} \\ \psi_1 \cdot \psi_2 = \psi}} \mathrm{TV}_r(\mathfrak{T}, \sigma_1, \psi_1) \cdot \mathrm{TV}_r(\mathfrak{T}, \sigma_2, \psi_2).$$

Once we have processed the entire tree, the root node $\rho$ of $X$ has $E^*_\rho = \emptyset$, $\Psi_\rho$ contains just the trivial colouring $\psi$ on $\emptyset$, and $\mathrm{TV}_r(\mathfrak{T}, \rho, \psi)$ for this trivial colouring is equal to the Turaev–Viro invariant $\mathrm{TV}_r(\mathfrak{T})$.

The time complexity of this algorithm is simple to analyse. Each leaf bag or introduce bag can be processed in $O(1)$ time (of course for the introduce bag we must avoid a deep copy of the data at the child node). Each forget bag produces $|\Psi_\tau| \le (r-1)^{|E^*_\tau|}$ colourings, each of which takes $O(|E^*_\tau|)$ time to analyse.

Naïvely, each join bag requires us to process $|\Psi_{\sigma_1}| \cdot |\Psi_{\sigma_2}| \le (r-1)^{|E^*_{\sigma_1}|+|E^*_{\sigma_2}|}$ pairs of colourings $(\psi_1, \psi_2)$. However, we can optimise this. Since we are only interested in colourings that agree on $E^*_{\sigma_1} \cap E^*_{\sigma_2}$, we can first partition $\Psi_{\sigma_1}$ and $\Psi_{\sigma_2}$ into buckets according to the colours on $E^*_{\sigma_1} \cap E^*_{\sigma_2}$, and then combine pairs from each bucket individually. This reduces our work to processing at most $(r-1)^{|E^*_{\sigma_1} \cup E^*_{\sigma_2}|}$ pairs overall. Each pair takes $O(|E^*_\tau|)$ time to process, and the preprocessing cost for partitioning $\Psi_{\sigma_i}$ is $O(|\Psi_{\sigma_i}| \cdot \log|\Psi_{\sigma_i}| \cdot |E^*_{\sigma_i}|) = O((r-1)^{|E^*_{\sigma_i}|} \cdot |E^*_{\sigma_i}|^2 \log r)$.

Suppose that our tree decomposition has width $k$. At each tree node $\tau$, every edge in $E^*_\tau$ must belong to some tetrahedron in the bag $B_\tau$, and so $|E^*_\tau| \le 6(k+1)$. Likewise, at each join bag described above, every edge in $E^*_{\sigma_1}$ or $E^*_{\sigma_2}$ must belong to some tetrahedron in the bag $B_{\sigma_i}$ and therefore also the parent bag $B_\tau$, and so $|E^*_{\sigma_1} \cup E^*_{\sigma_2}| \le 6(k+1)$. From the discussion above, it follows that every bag can be processed in time $O\left((r-1)^{6(k+1)} \cdot k^2 \log r\right)$, and so:

**Theorem 1** *Given a generalised triangulation $\mathfrak{T}$ of a 3-manifold with n tetrahedra, and a nice tree decomposition of $\mathfrak{T}$ with width k and $O(n)$ bags, Algorithm 1 computes* $\mathrm{TV}_r(\mathfrak{T})$ *in* $O(n \cdot (r-1)^{6(k+1)} \cdot k^2 \log r)$ *arithmetic operations in* $\mathbb{Q}(\zeta)$.

Theorem 1 shows that, if we have access to a tree decomposition of small width, then computing $\mathrm{TV}_r$, $r$ fixed, becomes a linear time procedure, even for large inputs. This of course is the main benefit of fixed-parameter tractability. In our setting, however, we have an added advantage: $\mathrm{TV}_r$ is a topological invariant, and does not depend on our particular choice of triangulation.

Therefore, if we are faced with a large treewidth triangulation, we can *retriangulate* the manifold (for instance, using bistellar flips and related local moves), in an attempt to make the treewidth smaller. While this is not always possible in theory (Huszár 2018), it is extremely effective in practice, as seen in Sect. 6.

Once we have a triangulation with dual graph of small treewidth, our approach relies on our ability to actually compute explicit small-width tree decompositions efficiently. Computing the treewidth of an arbitrary graph is NP-hard (Arnborg et al. 1987), but fixed-parameter tractable with respect to the treewidth (Bodlaender 1996). Consequently, together with Theorem 1, computing the Turaev–Viro invariant $\mathrm{TV}_r(\mathfrak{T})$ of a triangulation $\mathfrak{T}$ is fixed-parameter tractable in the treewidth of the dual graph of $\mathfrak{T}$. The running time of the exact algorithm (Bodlaender 1996) is $k^{O(k^3)}n$ for treewidth-$k$ graphs, which makes it intractable in practice. However, efficient approximation algorithms for tree decomposition exist (Amir 2010; Bodlaender et al. 2016; Feige et al. 2008). In particular, the 5-approximation algorithm of Bodlaender et al. (2016) has complexity $2^{O(k)}n$, and consequently produces tree decompositions of width reasonably close to the actual treewidth while maintaining a singly exponential complexity in the treewidth $k$. In particular, this yields an algorithm for the overall computation of the Turaev–Viro invariant $\mathrm{TV}_r(\mathfrak{T})$, $r$ fixed, with singly exponential complexity in $k$, and a linear complexity in $n$, on a triangulation $\mathfrak{T}$ with $n$ tetrahedra and treewidth $k$.

In the experimental section of this article, we use the polynomial time `GreedyFillIn` heuristic (Bodlaender and Koster 2010) which constructs tree decompositions that, in practice, are close to the actual treewidth, see Sect. 6.

Note that if—for one reason or another—we fail to obtain a small-width tree decomposition of the dual graph of our input triangulation, every tree node has $|E^*_\tau| \leq \ell$, where $\ell$ is the number of edges in the triangulation. Therefore the time complexity of Algorithm 1 reduces to $O(n \cdot (r-1)^{\ell} \cdot \ell^2 \log r)$, which is only a little slower than the backtracking algorithm (Lemma 1). This is in sharp contrast to many FPT algorithms from the literature, which—although fast for small parameters—suffer from extremely poor performance when the parameter becomes large.

## 5 Generalisation to the Turaev–Viro–Barrett–Westbury model

The FPT algorithm described in Sect. 4, generalises to every 3-manifold invariant which can be defined using a triangulation $\mathfrak{T}$ of the manifold and a state-sum over "local data" on $\mathfrak{T}$, that is, colourings of the edges of $\mathfrak{T}$, of the form of Eqs. (1) and (2)—for an arbitrary set of weights $|\cdot|_\theta$.

In this section we describe a very large family of 3-manifold invariants satisfying this condition, and due to Barrett and Westbury (1996). More precisely, the *Turaev–Viro–Barrett–Westbury model* defines a 3-manifold invariant for every (*finite semisimple*) *spherical category*, and substantially generalises the Turaev–Viro construction given in Sect. 2.1.

Roughly speaking, a spherical category is a category which admits the notion of local data (edge colourings)—stored in triangles and tetrahedra of the triangulation—from which a state-sum can be formed. The value of the state sum can then be shown to be invariant under bistellar flips on the triangulation. This is done by using a generalised version of what is called the *Biedenharn–Elliott identity* (Turaev 2010, Theorem VI.5.4.1). In particular this proves that the state sum defines a topological invariant.

For our purposes, the following properties of a spherical category $\mathcal{C}$ are important (we refer to Barrett and Westbury 1996 for a precise definition of a spherical category):

- $\mathcal{C}$ is equipped with a tensor product of objects $(a, b) \mapsto a \otimes b$ and of morphisms $(f, g) \mapsto f \otimes g$;
- $\mathcal{C}$ admits a *trace function* $\mathrm{tr} : \mathrm{Hom}(a, a) \to \mathbb{F}$, taking values in a fixed field $\mathbb{F}$, for every object $a$. Given two morphisms $f : a \to a$ and $g : b \to b$, the trace function satisfies $\mathrm{tr}(f \otimes g) = \mathrm{tr}(f)\mathrm{tr}(g)$, where the multiplication is in $\mathbb{F}$;
- $\mathcal{C}$ admits a finite set of nonequivalent non-zero simple objects $J$; and
- every Hom set is a finitely generated abelian group, and $\mathcal{C}$ defines an *evaluation map* $\mathrm{ev} : \mathrm{Hom}(a, b) \otimes \mathrm{Hom}(b, a) \to \mathbb{F}$ by $\mathrm{ev}(f \otimes g) = \mathrm{tr}(f \circ g)$.

A topological invariant is formed by taking a state-sum over all colourings of the edges $E$ of the triangulation $\mathfrak{T}$ by simple objects of $J$. For each such colouring $\theta : E \to J$ we associate a value $Z(\mathfrak{T}, \theta) \in \mathbb{F}$, computed in the following fashion. Fix an orientation of $\mathfrak{T}$, and let $T$ be a tetrahedron of the triangulation whose triangular faces are $t_1, t_2, t_3$ and $t_4$. Assign to $T$ the space $V(T, \theta) = \bigotimes_{i=1\dots4} H(t_i, \theta)$, where, for a triangle $t_i$ coloured $(a, b, c) \in J^3$ by $\theta$, $H(t_i, \theta)$ is either $\mathrm{Hom}(b, a \otimes c)$ or $\mathrm{Hom}^*(b, a \otimes c) \cong \mathrm{Hom}(a \otimes c, b)$, depending on whether the orientations of $(a, b, c)$ and $T$ agree or not. Define the space $V(\mathfrak{T}, \theta)$ to be the non-ordered tensor product of the $V(T, \theta)$ taken over all tetrahedra $T$ of $\mathfrak{T}$. Finally, every coloured tetrahedron $T \in \mathfrak{T}$ is assigned an element $Z(T, \theta) \in V(T, \theta)$, called a *6j-symbol*.

Note that, because every triangle is shared by exactly two tetrahedra with opposite orientations, every triangle coloured $(a, b, c)$ induces the appearance of spaces $\mathrm{Hom}(b, a \otimes c)$ and $\mathrm{Hom}(a \otimes c, b)$ in the tensor product $V(\mathfrak{T}, \theta)$. The state-sum is then defined by

$$Z(\mathfrak{T}) = K^{-\#V} \sum_{\theta: E \to J} Z(\mathfrak{T}, \theta) \prod_{e \in E} \dim_q(\theta(e)) \tag{3}$$

where

- $K$ is a constant depending solely on the category $\mathcal{C}$, called the *dimension* of $\mathcal{C}$;
- $\dim_q(a)$ is the quantum dimension of the object $a$, defined by $\mathrm{tr}(1_a)$;
- $Z(\mathfrak{T}, \theta) := \mathrm{ev}(\bigotimes_T Z(T, \theta))$ is the evaluation of $V(\mathfrak{T}, \theta)$, i.e., the product (in $\mathbb{F}$) over all triangles of $\mathfrak{T}$ of the evaluation map on $\mathrm{Hom}(b, a \otimes c) \otimes \mathrm{Hom}(a \otimes c, b) \to \mathbb{F}$ applied to the element given by the tensor product of the 6j-symbols.

As an example, we recover the Turaev–Viro invariant $\mathrm{TV}_r$ from this construction. Using the notations of Sect. 2.1, the simple objects of $J$ are labelled by half integers from 0 to $(r-2)/2$, and every state space $H(t, \theta)$ of a triangle whose edges are coloured with objects $a, b, c \in J$ is either the space 0, if $a$, $b$ and $c$ form a non admissible triple (in the sense of Sect. 2.1), or $\mathbb{C}$ otherwise. In Eq. (1), the product of vertex weights $|v|_\theta$ gives $K^{-\#V}$ in Eq. (3) (vertex weights are independent of $\theta$). The product of edge weights $|e|_\theta$ yields the product of $\dim_q(e)$ (edge weights are independent of the triangulation).

The dynamic programming approach of the fixed-parameter tractable algorithm of Sect. 4 is independent of the weight system in the state-sum formula of the invariant. Consequently, it adapts directly to the general definition of Turaev–Viro–Barrett–Westbury invariants.

**Corollary 2** *Given a generalised triangulation $\mathfrak{T}$ of a 3-manifold with n tetrahedra, a nice tree decomposition of $\mathfrak{T}$ of width k and $O(n)$ bags, and a finite semisimple spherical category $\mathcal{C}$, there exists a fixed-parameter tractable algorithm with respect to k to compute the $\mathcal{C}$-Turaev–Viro–Barrett–Westbury invariant of $\mathfrak{T}$ in $O(n \cdot (r-1)^{6(k+1)} \cdot k^2 \log r)$ arithmetic operations in the field $\mathbb{F}$ of $\mathcal{C}$, provided $\mathcal{C}$ admits r distinct inequivalent simple objects, and all weights in Eq. 3 are precomputed.*

## 6 Implementation and experimentation

In this section we describe an implementation of Algorithm 1 (the fixed-parameter tractable algorithm), and compare it to the backtracking algorithm via exhaustive experimentation.

The FPT algorithm is implemented in the open-source software package *Regina* (Burton et al. 1999): the source code is available from *Regina*'s public git repository. For consistency we compare it to *Regina*'s long-standing implementation of the backtracking algorithm.[2]

In our implementation, we do not compute treewidths precisely (an NP-complete problem)—instead, we implement the quadratic-time `GreedyFillIn` heuristic (Bodlaender and Koster 2010), which is reported to produce small widths in practice (van Dijk et al. 2006). This way, costs of building tree decompositions are insignificant (but included in the running times). For both algorithms, we use relatively naïve implementations of arithmetic in cyclotomic fields—these are asymptotically slower than described in Sect. 3, but have very small constants.

We use two data sets for our experiments, both taken from large census databases of 3-manifolds to ensure that the experiments are comprehensive and not cherry-picked.

The first census contains all 13,400 closed prime orientable manifolds that can be formed from $n \leq 11$ tetrahedra (Burton 2011; Matveev 2003). This simulates "real-world" computation—the Turaev–Viro invariants were used to build this census. Since the census includes all minimal triangulations of these manifolds, we choose the representative whose heuristic tree decomposition has smallest width (since we are allowed to retriangulate).

The second data set contains some of the 11,031 triangulations from the (much larger) Hodgson–Weeks census of closed hyperbolic manifolds (Hodgson and Weeks 1994). This shows performance on larger triangulations, with $n$ ranging from 9 to 20.

Figures 3 and 4 compare the performance of both algorithms for each data set. All running times are for $\mathrm{TV}_7$ at $\zeta = e^{i\pi/7}$ (the largest $r$ for which the experiments were feasible), and are measured on a single 3 GHz Intel Core i7 CPU. Both plots use a log-log scale with one data point per input triangulation. The results are striking: the FPT algorithm runs faster in over 99% of cases, including most of the cases with largest treewidth. In the worst example the FPT algorithm runs 3.7× slower than the backtracking, but both data sets have examples that run $> 440\times$ faster. It is also pleasing to see a clear impact of the treewidth on the performance of the FPT algorithm, as one would expect.

## 7 An alternate geometric interpretation

In this section, we give a geometric interpretation of admissible colourings on a triangulation of a 3-manifold $\mathfrak{T}$ in terms of *normal arcs*, i.e., straight lines in the interior of a triangle which are pairwise disjoint and meet the edges of a triangle, but not its vertices (see Fig. 5). More precisely, we have the following

**Theorem 2** *Given a 3-manifold triangulation $\mathfrak{T}$, and $r \geq 3$, an admissible colouring of the edges of $\mathfrak{T}$ with $r-1$ colours corresponds to a system of normal arcs in the 2-skeleton with $\leq r-2$ arcs per triangle forming a collection of cycles on the boundary of each tetrahedron of $\mathfrak{T}$.*

[2] The *Manifold Recogniser* (Matveev et al. 2012) also implements a backtracking algorithm, but it is not open-source and so comparisons are more difficult.

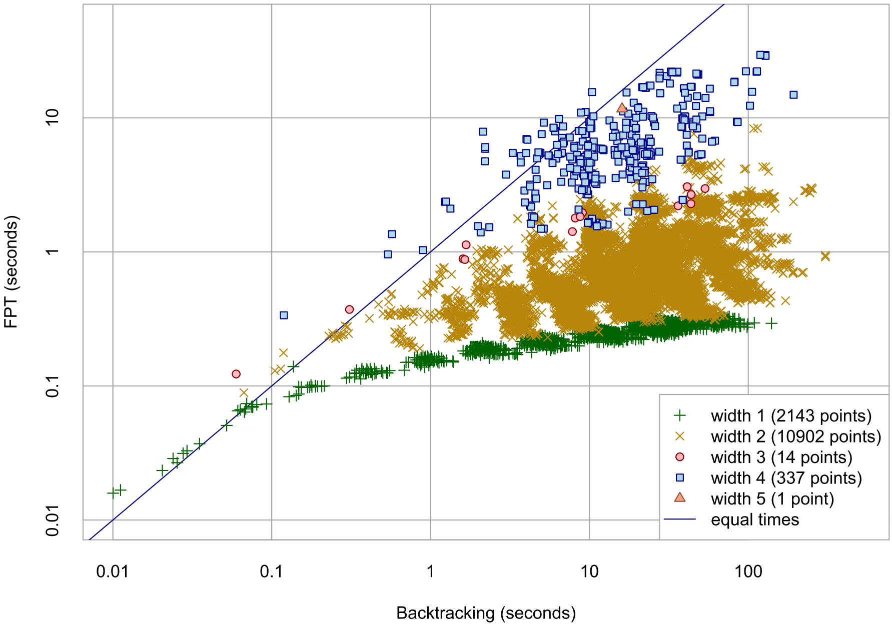


**Fig. 3** Running times for the 13,400 closed census manifolds. The term "width *k*" (an upper bound for the actual treewidth) denotes the width of the tree decomposition for the triangulation obtained from `GreedyFillIn` heuristic, and used for the FPT algorithm

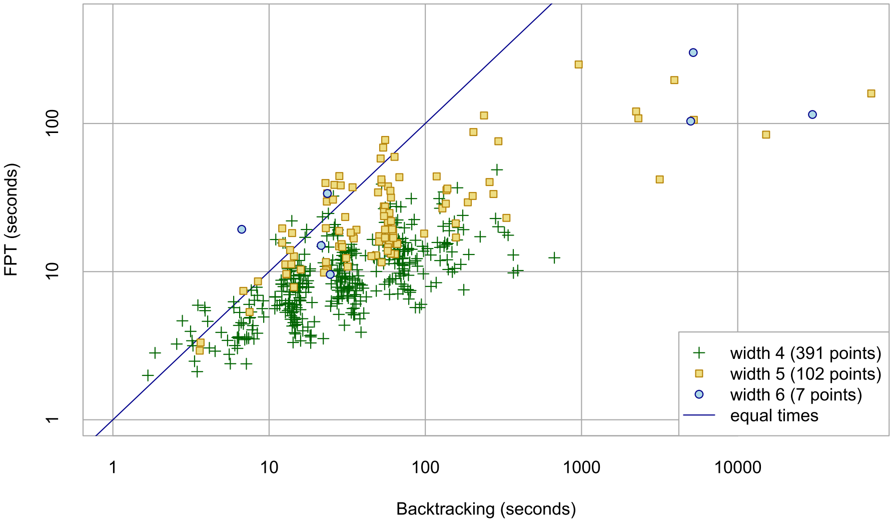
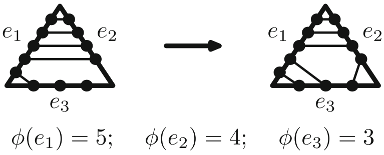


**Fig. 4** Running times for the first 500 Hodgson–Weeks census manifolds. The term "width *k*" (an upper bound for the actual treewidth) indicates the width of the tree decomposition for the triangulation obtained from `GreedyFillIn` heuristic, and used for the FPT algorithm

**Fig. 5** Constructing a system of normal arcs from edge colourings



***Proof*** Following the definition of an admissible colouring from Sect. 2.1, the colours of the edges $e_1$, $e_2$, $e_3$ of a triangle $f$ of $\mathfrak{T}$ must satisfy the parity condition and the triangle inequalities.

For a colouring $\theta(e)$ of an edge $e$, we define $\phi(e) = 2\theta(e)$ which is an integer; we also use the term "colouring" for $\phi$. We interpret the colourings $\phi(e_1), \phi(e_2), \phi(e_3) \in \{0, 1, \ldots, r-2\}$ as the number of intersections of normal arcs with the respective edges of the triangulation (see Fig. 5). Without loss of generality, let $\phi(e_1) \geq \phi(e_2) \geq \phi(e_3)$. We construct a system of normal arcs by first drawing $\phi(e_2)$ arcs between edge $e_1$ and $e_3$ and $\phi(e_1)-\phi(e_2)$ arcs between edge $e_1$ and $e_3$. This is always possible since $\phi(e_1) \leq \phi(e_2) + \phi(e_3)$ by the triangle inequality. Furthermore, the parity condition ensures that an even number of unmatched intersections remains which, by construction, all have to be on edge $e_3$. If this number is zero we are done. Otherwise we start replacing normal arcs between $e_1$ and $e_2$ by pairs of normal arcs, one between $e_1$ and $e_3$ and one between $e_2$ and $e_3$ (see Fig. 5). In each step, the number of unmatched intersection points decreases by two. By the assumption $\phi(e_2) \geq \phi(e_3)$, this yields a system of normal arcs in $f$ which leaves no intersection on the boundary edges unmatched. This system of normal arcs is unique for each admissible triple of colours. By the upper bound constraint, we get at most $r-2$ normal arcs on $f$.

Looking at the boundary of a tetrahedron $t$ of $\mathfrak{T}$ these normal arcs form a collection of closed cycles. To see this, note that each intersection point of a normal arc in a triangle with an edge is part of exactly one normal arc in that triangle and that there are exactly two triangles sharing a given edge.

Now, let $\mathfrak{T}$ be a closed $n$-tetrahedron 3-manifold triangulation, $t$ a tetrahedron of $\mathfrak{T}$, $f_1$ and $f_2$ two triangles of $t$ with common edge $e$ of colour $\phi(e)$, and $a_i$ and $b_i$ the respective non-negative numbers of the two normal arc types in $f_i$ meeting $e$, $i \in \{1, 2\}$. Since the system of normal arcs on $t$ forms a collection of cycles on the boundary of $t$, we must have $a_1 + b_1 = a_2 + b_2 \leq r - 2$, giving rise to a total of $6n$ linear equations and $12n$ linear inequalities on $6n$ variables which all admissible colourings on $\mathfrak{T}$ must satisfy. Thus, finding admissible colourings on $\mathfrak{T}$ translates to the enumeration of integer lattice points within the polytope defined by the above equalities and inequalities.

Now, if we drop the upper bound constraint above, we get a cone. Computing the Hilbert basis of integer lattice points of this cone yields a finite description of all admissible colourings for any $r \geq 3$ and, thus, the essential information to compute $\mathrm{TV}_r(\mathfrak{T})$ for arbitrary $r$. Transforming this approach into a practical algorithm is work in progress.

**Acknowledgements** We would like to thank the anonymous referees for valuable suggestions which helped to improve the paper.

## References

Alagic, G., Lo, C.: Quantum invariants of 3-manifolds and NP vs. #P. Quantum Inf. Comput. **17**(1–2), 125–146 (2017)

Alagic, G., Jordan, S.P., König, R., Reichardt, B.W.: Estimating Turaev–Viro three-manifold invariants is universal for quantum computation. Phys. Rev. A **82**(4), 040302(R) (2010)

Amir, E.: Approximation algorithms for treewidth. Algorithmica **56**(4), 448–479 (2010). https://doi.org/10.1007/s00453-008-9180-4

Arnborg, S., Corneil, D.G., Proskurowski, A.: Complexity of finding embeddings in a k-tree. SIAM J. Algebraic Discrete Methods **8**(2), 277–284 (1987)

Barrett, J.W., Westbury, B.W.: Invariants of piecewise-linear 3-manifolds. Trans. Am. Math. Soc. **348**, 3997–4022 (1996)

Bodlaender, H.L.: A linear-time algorithm for finding tree-decompositions of small treewidth. SIAM J. Comput. **25**(6), 1305–1317 (1996). https://doi.org/10.1137/S0097539793251219

Bodlaender, H.L., Koster, A.M.C.A.: Treewidth computations. I. Upper bounds. Inf. Comput. **208**(3), 259–275 (2010). https://doi.org/10.1016/j.ic.2009.03.008

Bodlaender, H.L., Drange, P.G., Dregi, M.S., Fomin, F.V., Lokshtanov, D., Pilipczuk, M.: A $c^k$ n 5-approximation algorithm for treewidth. SIAM J. Comput. **45**(2), 317–378 (2016). https://doi.org/10.1137/130947374

Burton, B.A.: Structures of small closed non-orientable 3-manifold triangulations. J. Knot Theory Ramif. **16**(5), 545–574 (2007)

Burton, B.A.: Detecting genus in vertex links for the fast enumeration of 3-manifold triangulations. In: ISSAC 2011: Proceedings of the 36th International Symposium on Symbolic and Algebraic Computation, pp. 59–66. ACM, New York (2011)

Burton, B.A., Budney, R., Pettersson, W., et al.: Regina: software for 3-manifold topology and normal surface theory (1999–2014). http://regina.sourceforge.net/

Burton, B.A., Downey, R.G.: Courcelle's theorem for triangulations (2014) **(preprint)**. arXiv:1403.2926

Burton, B.A., Maria, C., Spreer, J.: Algorithms and complexity for Turaev–Viro invariants. In: Automata, Languages, and Programming—42nd International Colloquium, ICALP 2015, Kyoto, Japan, July 6–10, 2015, Proceedings, Part I, pp. 281–293 (2015)

Cantor, D.G., Kaltofen, E.: On fast multiplication of polynomials over arbitrary algebras. Acta Inform. **28**(7), 693–701 (1991). https://doi.org/10.1007/BF01178683

Feige, U., Hajiaghayi, M., Lee, J.R.: Improved approximation algorithms for minimum weight vertex separators. SIAM J. Comput. **38**(2), 629–657 (2008). https://doi.org/10.1137/05064299X

Hodgson, C.D., Weeks, J.R.: Symmetries, isometries and length spectra of closed hyperbolic three-manifolds. Exp. Math. **3**(4), 261–274 (1994)

Huszár, K., Spreer, J., Wagner, U.: On the treewidth of triangulated 3-manifolds. In: 34nd International Symposium on Computational Geometry (SoCG 2018), Leibniz International Proceedings in Informatics (LIPICS), vol. 99, pp. 46:1–46:15 (2018)

Kauffman, L.H., Lins, S.: Computing Turaev–Viro invariants for 3-manifolds. Manuscripta Math. **72**(1), 81–94 (1991)

Kirby, R., Melvin, P.: Local surgery formulas for quantum invariants and the Arf invariant. In: Proceedings of the Casson Fest. Geometry and Topology Monographs, vol. 7, pp. 213–233. Geometry and Topology Publishers, Coventry (2004). https://doi.org/10.2140/gtm.2004.7.213

Kleiner, B., Lott, J.: Notes on Perelman's papers. Geom. Topol. **12**(5), 2587–2855 (2008)

Kloks, T.: Treewidth: Computations and Approximations, vol. 842. Springer, Berlin (1994)

Maria, C., Spreer, J.: A polynomial time algorithm to compute quantum invariants of 3-manifolds with bounded first Betti number. In: Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA 2017), pp. 2721–2732. ACM, New York (2017). arXiv:1607.02218

Markov, I.L., Shi, Y.: Simulating quantum computation by contracting tensor networks. SIAM J. Comput. **38**(3), 963–981 (2008). https://doi.org/10.1137/050644756

Matveev, S.: Algorithmic Topology and Classification of 3-Manifolds. Algorithms and Computation in Mathematics, vol. 9. Springer, Berlin (2003)

Matveev, S., et al.: Manifold recognizer. http://www.matlas.math.csu.ru/?page=recognizer. Accessed Aug 2012

Munkres, J.R.: Elements of Algebraic Topology. Addison-Wesley, Menlo Park (1984)

Robertson, N., Seymour, P.D.: Graph minors. II. Algorithmic aspects of tree-width. J. Algorithms **7**(3), 309–322 (1986)

Toda, S.: PP is as hard as the polynomial-time hierarchy. SIAM J. Comput. **20**(5), 865–877 (1991). https://doi.org/10.1137/0220053

Turaev, V.G.: Quantum Invariants of Knots and 3-Manifolds. de Gruyter Studies in Mathematics, vol. 18, revised edn. Walter de Gruyter & Co., Berlin (2010). https://doi.org/10.1515/9783110221848

Turaev, V.G., Viro, O.Y.: State sum invariants of 3-manifolds and quantum $6j$-symbols. Topology **31**(4), 865–902 (1992)

Valiant, L.G.: The complexity of computing the permanent. Theor. Comput. Sci. **8**, 189–201 (1979). https://doi.org/10.1016/0304-3975(79)90044-6

van Dijk, T., van den Heuvel, J.P., Slob, W.: Computing treewidth with LibTW (2006). http://www.treewidth.com/treewidth/

Walker, K.: On Witten's 3-manifold invariants (1991). http://canyon23.net/math/