EXACT COMPUTATION OF THE n-LOOP INVARIANTS OF KNOTS

STAVROS GAROUFALIDIS, ERIC SABO, AND SHANE SCOTT

ABSTRACT. The loop invariants of Dimofte-Garoufalidis is a formal power series with arithmetically interesting coefficients that conjecturally appears in the asymptotics of the Kashaev invariant of a knot to all orders in $1/N$. We develop methods implemented in `SnapPy` that compute the first 6 coefficients of the formal power series of a knot. We give examples that illustrate our method and its results.

CONTENTS

1. INTRODUCTION

1.1. **The Volume Conjecture to all orders in** $1/N$**.** The best known quantum invariant of a knot in 3-space is the Jones polynomial [Jon87]. The Kashaev invariant $\langle K \rangle_N$ of a knot K (for $N = 1, 2, \dots$) [Kas95] coincides with the evaluation of the Jones polynomial of a knot and its parallels at complex roots of unity [MM01]. The Volume Conjecture of Kashaev [Kas97] states that for a hyperbolic knot K,

$$\lim_{N \to \infty} \frac{1}{N} \log |\langle K \rangle_N| = \frac{\mathrm{Vol}(K)}{2\pi},$$

where $\mathrm{Vol}(K)$ is the hyperbolic volume of K. An extension of the Volume Conjecture to all orders in $1/N$ was proposed independently by Gukov and the first author [Guk05, Gar08]. Namely, for every hyperbolic knot K there exists a formal power series $\phi_K(\hbar) \in \mathbb{C}[\![\hbar]\!]$ such

that

$$\langle K \rangle_N \sim N^{3/2} e^{C_K N} \phi_K(2\pi i/N), \tag{1}$$

where C_K is the complexified volume of K divided by $2\pi i$,

$$\phi_K(\hbar) = \tau_K^{-\frac{1}{2}} \, \phi_{K,1}^+(\hbar), \tag{2a}$$

$$\phi_K^+(\hbar) \in 1 + \hbar F_K[\![\hbar]\!], \tag{2b}$$

$$\tau_K \in F_K, \tag{2c}$$

and F_K is the trace field of K.

1.2. **Ideal triangulations, shapes, and the loop invariants.** The left hand side of Equation (1) is concretely defined given a planar projection or an ideal triangulation of a knot, and is typically given by a finite state-sum where the summand is a ratio of quantum factorials. Examples of state-sum formulas for the Kashaev invariant of the 4_1, 5_2 and 6_1 knots are given in [Kas97, (2.2)-(2.4)].

On the other hand, the power series $\phi_K(\hbar)$ that conjecturally appears in the right hand side is not an explicit function of the knot. Numerical computations of the Kashaev invariant were performed by Zagier and G., and using numerical interpolation and a variety of guessing methods, it was possible to recognize the first few coefficients of the power series ϕ_K for several knots [GZ].

The main result of Dimofte-G. [DG13] was the construction of a power series $\phi_\gamma(\hbar)$ that depends on an ideal triangulation of a knot complement. For a detailed discussion on ideal triangulations and their gluing equations, see [Thu77, NZ85] and also [DG13, Sec.1.2]. Explicitly, an ideal triangulation $\mathcal{T}$ with N tetrahedra gives rise to a vector $z = (z_1, \ldots, z_N)$ of shapes that satisfy the Neumann-Zagier equations

$$\prod_{j=1}^{N} z_j^{\mathbf{A}_{ij}} (z_j'')^{\mathbf{B}_{ij}} = (-1)^{\nu_i}$$

for $i = 1, \ldots, N$ where $z_j'' = 1 - 1/z_j$. These equations are obtained as follows. Fix an edge of the ideal triangulation $\mathcal{T}$, and set the product of the shape parameters of all tetrahedra that go around the edge e equal to 1. If z is the shape of a tetrahedron that contains the fixed edge, then its contribution to the above product is z or z' or z'' according to the convention of Figure 1. Finally, replace $z' = -(zz'')^{-1}$ using the relation $zz'z'' = -1$. This gives rise to the equations (1.2), one for each edge of $\mathcal{T}$. Likewise, there is an equation os the same type for each peripheral curve.

In the above equations, we have removed one edge equation and replaced it with the meridian cusp equation. Neumann-Zagier [NZ85] prove that $(\mathbf{A}|\mathbf{B})$ is the upper part of a symplectic matrix. It follows that $(\mathbf{A}|\mathbf{B})$ has rank N and $\mathbf{A}\mathbf{B}^T$ is symmetric, where $\mathbf{B}^T$ is the transpose of $\mathbf{B}$. We will assume that $\mathbf{B}$ has nonvanishing determinant. Furthermore, we will assume that our triangulation $\mathcal{T}$ is such that there exists a solution to the gluing equations in $(\mathbb{C} \setminus \{0,1\})^N$ that recovers the complete hyperbolic structure of the hyperbolic knot K. In that case, z is a vector of algebraic numbers and $\mathbb{Q}(z_1, \ldots, z_N)$ is a number field (the shape field) which coincides with the invariant trace field and with the trace field of the knot [NR92, Thm.2.2,2.4]. Finally, one can choose a flatenning (f, f''), that is an integer
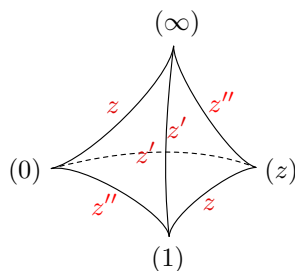
Figure 1. An ideal tetrahedron and its shape assignment.

solution of the linear equation $\mathbf{A}f + \mathbf{B}f'' = \nu$. This determines a Neumann-Zagier datum $\gamma = (A, B, \nu, f, f'', z)$, which in turn defines the power series $\phi_\gamma(\hbar)$. Of course, different ideal triangulations give rise to different Neumann-Zagier data, hence to potentially different formal power series $\phi_\gamma(\hbar)$. On the other hand, the left hand-side of Equation (1) depends only on the hyperbolic knot K. Although the topological invariance of $\phi_\gamma(\hbar)$ is not known, from the computational point of view, this gives an excellent consistency check of correctness of the code.

Equations (2a)-(2c) are manifest by the definition of $\phi_\gamma(\hbar)$. In [DG13] it was shown that τ_γ is a topological invariant, defined up to a sign. We may call τ_γ the 1-loop invariant. If we write

$$\phi_\gamma^+(\hbar) = \exp\left(\sum_{n=2}^\infty S_{\gamma,n}\hbar^{n-1}\right),$$

then $S_{\gamma,n}$ are the n-loop invariants of the γ. In [DG13] it was conjectured that $S_{\gamma,2}$ is well-defined up to addition of an integer multiple of $1/24$, and that $S_{\gamma,n}$ are topological invariants for $n \geq 3$.

The definition of $\phi_\gamma^+(\hbar)$ is given explicitly by formal Gaussian integration. It follows that $S_{\gamma,n}$ is a weighted sum of a finite set of Feynman diagrams with Feynman loop number at most n. The Feynman rules were explained in detail in [DG13, Sec.1.6-1.8] and the contributing Feynman diagrams for $n = 2$ and $n = 3$ were explicitly drawn. For $n > 3$, the number of Feynman diagrams gets large and drawings-by-hand is not advisable.

For the benefit of the reader, we recall the Feynman rules from [DG13, Sec.1.6-1.8]. By connected *Feynman diagram* G we mean a connected multigraph, possibly with loops and multiple edges. If G is a Feynman diagram, its *Feynman loop number* $L(G)$ is given by

$$L(G) = |V_1(G)| + |V_2(G)| + b_1(G)$$

where $|V_k(G)|$ is the number of k-valent vertices of G and $b_1(G)$ is the first betti number (also known as the number of holes) of G. It is easy to see that a connected Feynman diagram with loop number at most n has at most $2n - 2$ vertices and at most n holes. Hence, there are finitely many Feynman diagrams of loop number at most n.

Fix a Neumann-Zagier datum $\gamma = (\mathbf{A}, \mathbf{B}, \nu, f, f'', z)$ which we assume is non-degenerate, that is the propagator (defined below) makes sense. In each Feynman diagram G, the edges represent an $N \times N$ propagator

$$\Pi = \hbar\left(-\mathbf{B}^{-1}\mathbf{A} + \mathrm{diag}(1/(1-z))\right)^{-1}$$

n	2	3	4	5	6
g_n	6	40	331	3700	53758

Table 1. The number g_n of graphs that contribute to the n-loop invariant for $n = 2, \ldots, 6$.

while each k-vertex comes with an N-vector of factors $\Gamma_i^{(k)}$,

$$\Gamma_i^{(k)} = (-1)^k \sum_{p=\alpha_k}^{\alpha_k+n-L(D)} \frac{\hbar^{p-1}(-1)^p B_p}{p!} \mathrm{Li}_{2-p-k}(z_i^{-1}) + \begin{cases} -\frac{1}{2}(\mathbf{B}^{-1}\nu)_i & k = 1 \\ 0 & k \geq 2 \end{cases},$$

where $\alpha_k = 1$ (resp., 0) if $k = 1, 2$ (resp., $k \geq 3$). Here B_k is the k-th Bernoulli number ($B_1 = -1/2$, $B_2 = 1/6$) and $\mathrm{Li}_s(z) = \sum_{m=1}^{\infty} z^m/m^s \in \mathbb{Q}(z)$ is the s-polylogarithm function for s a nonpositive integer. The diagram G is then evaluated by contracting the *vertex factors* $\Gamma_i^{(k)}$ with propagators, multiplying by a standard *symmetry factor*, and taking the $\hbar^{n-1}$ part of the answer. In the end, $S_{\gamma,n}$ is the sum of evaluated diagrams, plus an additional *vacuum* contribution

$$\Gamma^{(0)} = \frac{B_n}{n!} \sum_{i=1}^{N} \mathrm{Li}_{2-n}(z_i^{-1}) + \begin{cases} \frac{1}{8} f \cdot \mathbf{B}^{-1}\mathbf{A}f & n = 2 \\ 0 & n \geq 3 \end{cases}.$$

1.3. **Our code.** Our goal is to give an exact computation for the n-loop invariants for $n = 1, \ldots, 6$ of a Neumann-Zagier datum of a `SnapPy` triangulation. Our method is implemented in `SnapPy`. We accomplished this in three steps.

(a) We wrote a Python method `generate_feynman_diagrams.py` that generates all Feynman diagrams that contribute to the n-loop invariant. The Feynman diagrams were generated by first generating trees, and then adding to them multiple edges or loops. The number of such diagrams is shown in Table 1. Observe that if G is a multigraph with corresponding simple graph $S(G)$ then $S(G)$ has at most $2n-2$ vertices and at most n holes, and $L(G)$ can be obtained from $S(G)$ by adding at most $n - L(S(G)) + |V_1(S(G))| + |V_2(S(G))|$ edges. Thus, all Feynman diagrams with Feynman loop number at most n can thus be generated by first generating all trees with at most $2n-2$ vertices then iteratively adding edges between pairs of vertices. Every edge added also adds an additional hole. If multigraph G has more than $n - |V_1(G)| - |V_2(G)|$ holes it cannot be the subgraph of a Feynman diagram with Feynman loop number at most n.

(b) We wrote a Python class `NeumannZagierDatum` which gives the Neumann-Zagier matrices and the exact value of the shape parameters that recover the geometric representation of an ideal triangulation. The exact computation of the shape parameters was done using the Ptolemy module [GGZ15, CDW] and the numerical computation is already implemented in `SnapPy`.

(c) We wrote Python classes `nloop_exact.py` and `nloop_num.py` which given a Neumann-Zagier datum γ and a natural number $n = 1, \ldots, 6$ computes $S_{\gamma,n}$ exactly (as an element of the trace field) or numerically to arbitrary precision.

To verify correctness of our code, we computed the n-loop invariants for $n = 1, \ldots, 5$ for different triangulations of each of a fixed knot, such as 5_2, $(-2, 3, 7)$ pretzel, 6_1, and 6_2. In all cases the results agreed (up to a sign when $n = 1$ and up to addition of $1/24$ times an integer when $n = 2$). This illustrates both the topological invariance of the n-loop invariants, and the correctness of our code.

1.4. **Usage.** The essence of our code lies in two Python classes `NeumannZagierDatum` and `nloop`. The former takes as input a manifold and generates the Neumann-Zagier datum $\gamma = (A, B, \nu, f, f'', z)$, and the latter takes as input Neumann-Zagier datum, an integer n, and a list of Feynman diagrams and returns the n-th loop invariant $S_{\gamma,n}$.

The `NeumannZagierDatum` class has three optional arguments `engine`, `verbose`, and `file_name`, which are set to `None`, `False`, and `None`, respectively, by default. The `engine` variable is passed as an option into the Ptolemy module and controls the method in which solutions to the Ptolemy variety are found. The preferred value for this variable for manifolds in `CensusKnots` is `engine="magma"`, which refers to the `Sage` interface to the `Magma` Computational Algebra System [BCP97]. If Magma is not available, `engine="None"` will attempt to compute solutions of the Ptolemy variety using `Sage`. Solutions for manifolds in `HTLinkExteriors` and `LinkExteriors` have been precomputed and are available with the Ptolemy module using `engine="retrieve"`, [GGZ15]. This option requires an internet connection, but will automatically switch to recomputing locally if the download is unsuccessful. The output of the Ptolemy module including the `retrieve` option are suppressed with `verbose=False` and are displayed with `verbose=True`.

To utilize the `NeumannZagierDatum` class use a terminal to navigate to the directory containing `nloop_exact.py`, available at [GSS], and load `Sage`. Once loaded, the class must first be initiated via

```
sage:  attach('nloop_exact.py')
sage:  M = Manifold('6_1')
sage:  D = NeumannZagierDatum(M, engine="retrieve").
```

To generate the Neumann-Zagier datum use

```
sage:  D.generate_nz_data().
```

This will assign a Python list $[A, B, \nu, f, f'', z, \text{embedding}]$ consisting of the Neumann-Zagier datum plus the embedding of the something in the something to the class variable `nz`. If the optional argument `file_name` is used, this variable will be saved as a `Sage` object file (*.sobj) in the current directory. To view the data simply use

```
sage:  D.nz.
```

The shape equations z and field embedding may be computed separately via

```
sage:  D.exact_shapes_via_ptolemy_lifted()
```

and

```
sage:  D.compute_ptolemy_field_and_embedding(),
```

respectively.

Once the Neumann-Zagier datum has been computed, one may use it to compute the n-loop invariants $S_{\gamma,n}$. First, load the Feynman diagrams you wish to use and choose an invariant you wish to calculate,

```
sage:  n = 2
```

```
sage:  diagrams = load('6diagrams.sobj')
sage:  E = nloop(D.nz, n, diagrams).
```
Here, we have chosen to calculate $S_{\gamma,2}$ using Feynman diagrams up to six loops. Note that the manifold M is not directly used when initiating the nloop class, as all the information about the manifold we need is encoded in the Neumann-Zagier datum D.nz. To compute the invariant use
```
sage:  E.one_loop()
```
if $n = 1$ or
```
sage:  E.nloop_invariant()
```
otherwise. To do this using a precomputed Neumann-Zagier datum Sage object file instead of defining D as above use
```
sage:  nz = load('nz_exact_6_1.sobj')
sage:  E = nloop(nz, n, diagrams).
```
The entire process described above has been streamlined into two automated functions for convenience. For example, to start with a specified manifold M and diagrams list, compute the Neumann-Zagier datum, and then compute the n-loop invariant, simply use
```
sage:  nloop_from_manifold(M, n, diagrams, engine="retrieve").
```
The NeumannZagierDatum optional arguments described above may be entered here as seen in the example. On the other hand, to start with a precomputed Neumann-Zagier datum Sage object file (loaded as nz) and a diagrams list then compute the n-loop invariant, simply use
```
sage:  nloop_from_nzdatum(nz, n, diagrams, engine="retrieve").
```
Also available at [GSS] is an almost identical version of our code, nloop_num.py, which produces numerical results to arbitrary precision instead of exact computations. The usage for this file is the same.

1.5. **Sample computations.** The results of our computations are available from [GSS], along with the code and data files.

To illustrate our method, consider the $6_1 = K4_1$ knot with trace field $F_{6_1} = \mathbb{Q}(x)$, where $x = -1.50410836415074 \cdots + i1.22685163774658 \ldots$ is a root of

$$x^4 + 2x^3 + x^2 - 3x + 1 = 0.$$

F_{6_1} is a number field of type $[0, 2]$ with discriminant 257, a prime number. It follows that the Bloch group $\mathcal{B}(F_{6_1})$ is a finitely generated abelian group of rank 2 [Sus90, Zic09]. The default SnapPy triangulation for $K4_1$ uses 4 ideal tetrahedra with shapes

$$z = \left(\frac{3}{2}x^3 + \frac{7}{2}x^2 + 3x - \frac{5}{2}, \ 2x^3 + 5x^2 + 5x - 3, \ -\frac{1}{2}x^3 - \frac{3}{2}x^2 - x + \frac{3}{2}, \ \frac{1}{2}x^3 + \frac{3}{2}x^2 + 2x + \frac{1}{2} \right).$$

A Neumann-Zagier datum $\gamma = (A, B, \nu, f, f'', z)$ is given by

$$A = \begin{pmatrix} 1 & 0 & -1 & 0 \\ -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix}, \ B = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 2 \end{pmatrix}, \ \nu = \begin{pmatrix} 1 \\ 2 \\ 1 \\ 2 \end{pmatrix}, \ f = \begin{pmatrix} 1 \\ 2 \\ 0 \\ 1 \end{pmatrix}, \ f'' = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

The n-loop invariants for $n = 1, \ldots, 6$ are given by:

$$\tau = -\frac{7}{2}x^3 - \frac{17}{2}x^2 - \frac{17}{2}x + 6$$

$$S_2 = \frac{46490}{198147}x^3 + \frac{231209}{396294}x^2 + \frac{473191}{792588}x - \frac{62777}{264196}$$

$$S_3 = \frac{570416}{16974593}x^3 + \frac{2833463}{33949186}x^2 + \frac{1122215}{16974593}x - \frac{1386486}{16974593}$$

$$S_4 = -\frac{2255130587026}{50451970187565}x^3 - \frac{91695358340911}{807231523001040}x^2 - \frac{85651263871967}{807231523001040}x + \frac{1596902056811}{20180788075026}$$

$$S_5 = -\frac{37040877003091}{1728820845093894}x^3 - \frac{330280282463219}{6915283380375576}x^2 - \frac{53499149965837}{1728820845093894}x + \frac{72838757049049}{1152547230062596}$$

$$S_6 = \frac{1449319256564305241317}{17984434859623040256945}x^3 + \frac{23592842410230239076799}{115100383101587457644448}x^2 +$$
$$\frac{110567432832899754708187}{575501915507937288222240}x - \frac{20008494585620168748319}{143875478876984322055560}$$

References

[BCP97] Wieb Bosma, John Cannon, and Catherine Playoust, *The Magma algebra system. I. The user language*, J. Symbolic Comput. **24** (1997), no. 3-4, 235–265, Computational algebra and number theory (London, 1993).

[CDW] Marc Culler, Nathan M. Dunfield, and Jeffery R. Weeks, *SnapPy*, http://www.math.uic.edu/t3m/SnapPy.

[DG13] Tudor Dimofte and Stavros Garoufalidis, *The quantum content of the gluing equations*, Geom. Topol. **17** (2013), no. 3, 1253–1315.

[Gar08] Stavros Garoufalidis, *Chern-Simons theory, analytic continuation and arithmetic*, Acta Math. Vietnam. **33** (2008), no. 3, 335–362.

[GGZ15] Stavros Garoufalidis, Matthias Goerner, and Christian Zickert, *The Ptolemy field of 3–manifold representations*, Algebr. Geom. Topol. **15** (2015), no. 1, 371–397.

[GSS] Stavros Garoufalidis, Eric Sabo, and Shane Scott, http://www.math.gatech.edu/~stavros/publications/nloop.data/index.html/.

[Guk05] Sergei Gukov, *Three-dimensional quantum gravity, Chern-Simons theory, and the A-polynomial*, Comm. Math. Phys. **255** (2005), no. 3, 577–627.

[GZ] Stavros Garoufalidis and Don Zagier, *Asymptotics of quantum knot invariants*, Preprint 2013.

[Jon87] V. F. R. Jones, *Hecke algebra representations of braid groups and link polynomials*, Ann. of Math. (2) **126** (1987), no. 2, 335–388.

[Kas95] Rinat M. Kashaev, *A link invariant from quantum dilogarithm*, Modern Phys. Lett. A **10** (1995), no. 19, 1409–1418.

[Kas97] R. M. Kashaev, *The hyperbolic volume of knots from the quantum dilogarithm*, Lett. Math. Phys. **39** (1997), no. 3, 269–275.

[MM01] Hitoshi Murakami and Jun Murakami, *The colored Jones polynomials and the simplicial volume of a knot*, Acta Math. **186** (2001), no. 1, 85–104.

[NR92] Walter D. Neumann and Alan W. Reid, *Arithmetic of hyperbolic manifolds*, Topology '90 (Columbus, OH, 1990), Ohio State Univ. Math. Res. Inst. Publ., vol. 1, de Gruyter, Berlin, 1992, pp. 273–310.

[NZ85]   Walter D. Neumann and Don Zagier, *Volumes of hyperbolic three-manifolds*, Topology **24** (1985), no. 3, 307–332.
[Sus90] Andrei Suslin, K_3 *of a field, and the Bloch group*, Trudy Mat. Inst. Steklov. **183** (1990), 180–199, 229, Translated in Proc. Steklov Inst. Math. **1**991, no. 4, 217–239, Galois theory, rings, algebraic groups and their applications (Russian).
[Thu77] William Thurston, *The geometry and topology of 3-manifolds*, Universitext, Springer-Verlag, Berlin, 1977, Lecture notes, Princeton.
[Zic09] Christian K. Zickert, *The extended Bloch group and algebraic K-theory*, 2009, arXiv:0910.4005, Preprint.

School of Mathematics, Georgia Institute of Technology, Atlanta, GA 30332-0160, USA
http://www.math.gatech.edu/~stavros
  E-mail address: stavros@math.gatech.edu

School of Mathematics, Georgia Institute of Technology, Atlanta, GA 30332-0160, USA
http://www.math.gatech.edu/users/esabo3
  E-mail address: esabo3@gatech.edu

School of Mathematics, Georgia Institute of Technology, Atlanta, GA 30332-0160, USA
http://www.math.gatech.edu/users/sscott42
  E-mail address: scottsha@gatech.edu