COMPUTING RIEMANN THETA FUNCTIONS IN SAGE

CHRIS SWIERCZEWSKI
UNIVERSITY OF WASHINGTON
DEPARTMENT OF APPLIED MATHEMATICS
CSWIERCZ@AMATH.WASHINGTON.EDU

The Kadomtsev-Petviashvili (KP) equation

$$(-4u_t + 6uu_x + u_{xxx})_x + 3\sigma^2 u_{yy} = 0, \quad \sigma^2 = \pm 1 \tag{1}$$

describes the propagation of two-dimensional shallow water waves. The non-linear Schrödinger (NLS) equation

$$i\psi_t = -\tfrac{1}{2}\psi_{xx} + \kappa|\psi|^2\psi \tag{2}$$

has applications in optics and water waves. One thing these two equations have in common is that they both have analytic solutions in terms of the Riemann theta function. Understanding the function requires drawing facts from complex analysis, linear analysis, combinatorics, algebra, and even number theory. Riemann theta also has application in topics in number theory, algebraic geometry, integerable differential equations, and combinatorics.

However, even though Riemann knew of the function in the late 1800's, it was not until recently with the work of Deconinck and van Hoeij that we have been able to plot one, let alone use it in a computational setting to solve equations such as KP and NLS.

In this paper we will introduce the theory of Riemann theta functions and summarize the work of Bobenko, Deconinck, Heil, van Hoeij, and Schmies which allows us to compute Riemann theta functions to any desired numerical accuracy. We will also discuss the Sage implementation in particular which was initiated by Nick Alexander at UC Irvine and is now taken on by the author. Studying and implementing methods for computing the Riemann theta function key components of the author's thesis work.

1. Introduction to Riemann Theta Functions

Let us start with the definition of the Riemann theta function:

Definition 1. (Riemann Theta Function) *Let $\Omega \in \mathbb{C}^{g\times g}$ be a Riemann matrix; i.e. Ω is symmetric and $\mathrm{Im}(\Omega)$ is strictly positive definite. Then the Riemann theta function, $\theta : \mathbb{C}^g \to \mathbb{C}^g$ is defined as*

$$\theta(z|\Omega) = \sum_{n\in\mathbb{Z}^g} e^{2\pi i(\langle n,\Omega n\rangle + \langle n,z\rangle)}.$$

This series converges absolutely in z and Ω and uniformly on compact sets. Note that the function is periodic with period 1 in each z component; that is, $\theta(z+m|\Omega) = \theta(z|\Omega), \forall m \in \mathbb{Z}^g$.

Date: 10 December 2010.

1.1. **A Natural Construction of Riemann Theta Functions.** One of the necessary parameters is the choice of Riemann matrix Ω. In this section we describe how to compute the Riemann matrix corresponding to a complex plane algebraic curve.

The set of all $x, y \in \mathbb{C}$ such that

$$f(x,y) = a_d(x)y^d + \cdots + a_1(x)y + a_0(x) = 0$$

defines a d-sheeted algebraic covering $y(x)$ of the Riemann sphere. The behavior at the branch points of $y(x)$ determines the topology of the curve's corresponding Riemann surface. The genus of the Riemann surface can be computed explicitly via the Hurwitz formula.

Theorem 2. (Hurwitz Formula) *Let e_P by the ramification index of the covering $y(x)$ at a point P on the Riemann surface Γ. ($e_P =$ the number of sheets of $y(x)$ that meet at P.) Define the total branching number*

$$W = \sum_{P \text{ branch point of } \Gamma} (e_P - 1).$$

The genus g of the Riemann surface is given by the Hurwitz formula

$$g = \frac{W}{2} - d + 1$$

The genus of a curve's Riemann surface tells us a lot about the covering's behavior. To further develop the geometry of a Riemann surface we need to consider homotopy theory and the surface's homology group. For an introduction to homotopy theory and the homology of a surface in full generality, see Munkres. [**?**] With that said, given a genus g surface, Γ, there are $2g$ nonhomologous cycles $a_1, \ldots, a_g, b_1, \ldots, b_g$ such that

$$a_i \circ a_j = 0,\ b_i \circ b_j = 0,\ a_i \circ b_j = \delta_{ij};$$

where δ_{ij} is the Kronecker delta and $a \circ b = 0$ if and only if the cycles a and b do not intersect. These $2g$ cycles form a basis for the homology group of the Riemann surface Γ.

The cohomology of a Riemann surface is given in terms of its basis $\{\omega_1, \ldots, \omega_g\}$ of holomorphic differentials of the form

$$\omega_k = \frac{P_k(x,y)}{\partial_y f(x,y)} dx$$

Definition 3. (Holomorphic Differential) *A holomorphic (resp. meromorphic) differential ω on a Riemann surface Γ is a family $\{(U_i, z_i, \omega_i)\}$ such that $\{(U_i, z_i)\}$ is a holomorphic covering of Γ and $\omega_i = f_i(z_i)dz_i$ where $f_i : U_i \to \mathbb{C}$ is a holomorphic (resp. meromorphic) function and where the chain rule is preserved under coordinate transformations $\phi_{ij} : U_i \cap U_j \to U_i \cap U_j$.*

Note that the denominator $\partial_y f(x,y)$ vanishes at the branch points and singular points of the covering $y(x)$ whereas the differential form dx vanishes at the branch points. Hence, for ω_k to be a holomorphic differential we need to find a $P_k(x,y)$ that vanishes at the singular points.

A *period matrix*, $\tilde{\Omega}$, is obtained by integrating the normalized differentials on the elements of the Riemann surface's homology basis.

$$\tilde{\Omega} = (A\ B) \in \mathbb{C}^{g \times 2g}$$

where

$$A = (A_{ij})_{i,j=1}^{g}, \quad A_{ij} = \oint_{a_j} \omega_i$$
$$B = (B_{ij})_{i,j=1}^{g}, \quad B_{ij} = \oint_{b_j} \omega_i.$$

However, with an appropriate choice of holomorphic differentials (basis of the cohomology of Γ) such that $\oint_{a_j} \omega_i = \delta_{ij}$, we have

$$\tilde{\Omega} = (I\ \Omega)$$

where Ω is the desired Riemann matrix appearing in the Riemann theta function and $\tilde{\Omega}$ is called the normalized period matrix of the Riemann surface.

The normalized period matrix encodes much of the geometric information about a curve's Riemann surface . The column space of Γ describes a lattice Λ in $\mathbb{C}^g$ and is given by

$$\Lambda = \{v \in \mathbb{C}^g \,|v = m + \Omega n; m, n \in \mathbb{Z}^g\}$$

The *fundamental parallelepiped*, also known as the *Jacobian*, of the normalized period matrix is the quotient space

$$J(\Gamma) = \mathbb{C}^g / \Lambda.$$

In the case when $g = 1$, Λ is simply a lattice in $\mathbb{C}$ and is isomorphic to the complex torus. This particular case describes the geometry of an elliptic curve.

2. Computing Riemann Theta Functions

The ability to compute a Riemann matrix corresponding to a plane algebraic curve as well as the Riemann theta function itself for a given Riemann matrix is already implemented in Maple's `algcurves` package. This work is the result of the work of Deconinck and van Hoeij. For several reasons, the authors of this Maple implementation would like to see their code ported to Sage where they and future developers, such as myself, can have tighter control over its performance and interaction with other components of the software.

Given a complex plane algebraic curve, computing the curve's Riemann theta function requires several components:

- *Compute the Riemann matrix, Ω. This requires several components:*
  - Compute the monodromy group about all of the branch points. The monodromy group is a permutation group describing which sheets are connected at each branch point and thus the geometry of the Riemann surface.
  - Compute the basis of the Riemann surface homology $\{a_k, b_k\}$ using the monodromy group.
  - Compute the basis of the cohomology $\{\omega_k\}$ by numerically integrating about the elements of the homology.
- *Given Ω, determine the number of terms needed in $\theta(z|\Omega)$ to achieve a desired accuracy.* See *"Computing Riemann Theta Functions"* by Bobenko, Deconinck, Heil, van Hoeij, and Schmies for a uniform approximation formula with arbitrary precision. Here, we will outline the steps needed to compute the pointwise approximation to $theta(z|\Omega)$ for a given z:
  - Split z and Ω into their real and imaginary parts.

  - Use Equation (6) in *"Computing Riemann Theta Functions"* to split the exponential growth terms off from the oscillatory part of $\theta(z|\Omega)$.
  - Given an error bound, ϵ, determine the set of integer points S_R needed to achieve this desired accuracy: this involves Cholesky decomposotions, floating point LLL, and the inverse incomplete gamma function.
  - Compute the sum and combine the exponential and oscillatory terms.

.

As of the writing of this paper the author has only implemented the evaluation of $\theta(z|\Omega)$ and its derivatives for a given Riemann matrix, Ω. Furthermore, this implementation only uses the pointwise approximation of θ. There are several components that should be written before this code is implemented in Sage:

- unfiorm approximation fomulas for θ: this allows us to use the same bounding ellipse for all points z. Cacheing these points will result in a speed improvement when evaluating θ on a large domain; for example, when plotting θ.
- Cythonized summation: the number of points needed in S_R grows exponentially with the genus, g. For example, in order to evaluate θ corresponding to a genus $g = 6$ curve with 30 bits of precision one requires on the order of 10^5 integer points. There are good reasons for looking at large genus curves so we should squeeze out as much performance as possible in our computation.
- Riemann theta with characteristic: Riemann theta with characteristic, denoted $\theta[\alpha, \beta](z|\Omega)$, is a commonly used modification of θ and hence should be made easily accessible.

The construction of Ω from a plane algebraic curve is part of the author's thesis work. It is a much more difficult computation than the evaluation of θ. Development of a Sage implemention will commence once Riemann theta is in suitable condition and ready for review.