Calculus on Riemann Surfaces in Python

Chris Swierczewski

cswiercz@uw.edu
University of Washington
Department of Applied Mathematics

Symbolic Computation Seminar
North Carolina State University
March 2013

March 19, 2013

Table of Contents

Acknowledgements

Collaborators:

- Bernard Deconinck: advisor, co-author of `algcurves` for Maple.
- Grady Williams: undergraduate at UW, computing Riemann theta functions using CUDA.

Primary references:

- Mumford, *"Tata Lectures on Theta I,II"*
- Bobenko, Klein, *"Computational Approach to Riemann Surfaces"*
  - Chapter 2 by Deconinck and Patterson, *"Computing with Plane Algebraic Curves and Riemann Surfaces"*

Motivation

- Abelian functions: higher genus versions of elliptic functions. Periodic functions on $\mathbb{C}^g/\Lambda$).
    - Any Abelian function can be expressed as a rational function of the Riemann theta function $\theta : \mathbb{C}^g \times \mathfrak{h}_g \to \mathbb{C}$ and its derivatives.

$$\theta(z, \Omega) = \sum_{n \in \mathbb{Z}^g} e^{2\pi i \left(\frac{1}{2} n \cdot \Omega n + n \cdot z \right)}$$

    - Algorithm for computing the Riemann theta function by Deconinck, et. al.
- Period matrices corresponding to complex plane algebraic curves appear in Abelian function theory.

Motivation

Additional applications of period matrices and Riemann theta functions:

- Periodic solutions to integrable partial differential equations. (Dubrovin)
- Bitangents of complex plane quartics. (Vinzant, Plaumann, Sturmfels)
- Linear matrix representations of Helton-Vinnikov curves. (Vinzant, Plaumann, Sturmfels)
- Igusa polynomials and Siegel modular forms. (Lauter, Yang)

Riemann Surfaces

Let $f \in \mathbb{C}[x,y]$. The Riemann surface, X, corresponding to f is the desingularization and compactification of

$$X = \{(x,y) \in \mathbb{C}^2 \ : \ f(x,y) = 0\}. \tag{1}$$

Interpret X as a y-covering of the Riemann sphere $\mathbb{C}_x^*$. If

$$f(x,y) = a_n(x)y^n + \cdots + a_1(x)y + a_0(x)$$

then the covering is n-sheeted. Above all but finitely many $x \in \mathbb{C}_x^*$ (the branch points of f) there are n distinct values of $y \in \mathbb{C}_y$.

- "Fibre at x" or "y-roots at x".

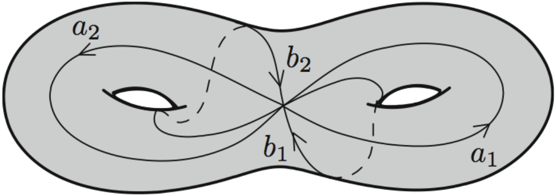
Riemann Surfaces

X is homeomorphic a complex g-torus. The homology group of X

$$H_1(X, \mathbb{Z})$$

is the space of all closed paths (cycles) on X. Basis consists of $2g$ cycles $\{a_1, \ldots, a_g, b_1, \ldots, b_g\}$ with the intersection properties

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



Dual space of holomorphic differentials on X

$$\Gamma(X, \Omega^1)$$

has dimension g.

Period Matrices and Riemann Matrices

Choose the normalized basis of $\Gamma(X, \Omega^1)$ where.

$$\oint_{a_i} \omega_j = \delta_{ij}, \quad \oint_{b_i} \omega_j = \Omega_{ij} \in \mathbb{C}.$$

Period matrix

$$\begin{pmatrix} I_{g\times g} & \Omega \end{pmatrix} \in \mathbb{C}^{g\times 2g}$$

where $\Omega \in \mathbb{C}^{g\times g}$ is a *Riemann matrix:*

- $\Omega = \Omega^T$,
- Im $\Omega \succ 0$.

The period matrix is used to define the Jacobian of a Riemann surface.

$$\mathrm{Jac}(X) := \mathbb{C}^g / (I_{g\times g}\mathbb{Z}^g + \Omega\mathbb{Z}^g)$$

Software: abelfunctions

abelfunctions is a Python library for computing with Riemann surfaces and (eventually) Abelian functions.

- A Python + C port of Deconinck, van Hoeij's algcurves package in Maple.
- Open-source and free.
- Uses numpy, scipy, sympy, networkx.

Available on GitHub:
https://github.com/cswiercz/abelfunctions

github SOCIAL CODING

Software: Path to Riemann Matrices

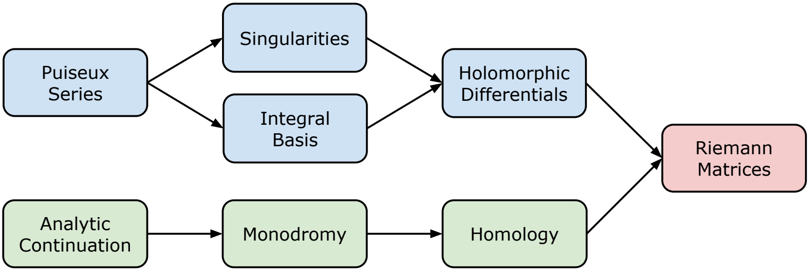


Table of Contents

Puiseux Series

Taylor and Laurent series locally describe single-valued functions at a point. Puiseux series are a multi-valued extension of these. Can locally describe

- regular points,
- branch points,
- singular points.

"point" $=$ a value in the complex x-plane. "place" a location on the Riemann surface, X.

Puiseux Series: Definition

A place P on a Riemann surface X lying above some $x = \alpha$ is determined by expansions of the form

$$P = \begin{cases} x = \alpha + \lambda t^e, \\ y = \sum_{k=0}^{\infty} \beta_k t^{n_k} \end{cases} \tag{2}$$

where $\cdots < n_k < n_{k+1} < \cdots$ with only finitely many negative exponents. Solve for $t = t(x)$:

$$y = y(x) = \sum_{k=0}^{\infty} \tilde{\alpha}_k (x - \alpha)^{n_k/e}.$$

Local description of y as a function of x on X.

- Algorithm by D. Duval using Newton polygons.

Demo

Puiseux series at places on complex plane algebraic curves.

Singularities

Must resolve the singularities of an algebraic curve to obtain corresponding Riemann surface. We use Puiseux series to do so.

- Puiseux series define local behavior. $\Longrightarrow$ How to go around / pass through singular points.
- Use to determine coordinate chart at each singular point, providing the manifold structure.

Definition: Singularities

Let $F(X, Y, Z) \in \mathbb{C}[X, Y, Z]$ be the homogenization of $f(x, y) \in \mathbb{C}[x, y]$. The singular points, $P \in P^2\mathbb{C}$, occur where

$$\partial_X F(P) = \partial_Y F(P) = \partial_Z F(P) = 0$$

Singularity Data

Important pieces of information corresponding to singular points $P \in P^2\mathbb{C}$.

- **multiplicity**: the sum of the ramification indices of the Puiseux series at P.
- **delta invariant**: the number of double points at P. Appears in the genus formula: if d is the total degree of f then the genus of the curves corresp. Riemann surface is

$$g = (d-1)(d-2)/2 - \sum_{P \in X} \delta_P$$

- **branching number**: the number of Puiseux series at P.

Demo

Singularities. Computing the genus of a curve.

Differentials

Holomorphic differentials on $X : f(x,y) = 0$ are all of the form

$$\omega_k = \frac{P_k(x,y)}{\partial_y f(x,y)} dx, \quad P_k(x,y) = \sum_{i+j \leq d-3} c_{kij} x^i y^j$$

where d is the degree of f and $P_k(x,y)$ are the “adjoint polynomials” of f are chosen such that ω_k has no poles on X.

- If X is nonsingular then all polynomials $P_k(x,y)$ of degree $\leq d-3$ give rise to a holomorphic differential. Consistent with genus formula:

$$g = (d-1)(d-2)/2$$

- If X has singularities then more conditions are imposed on the $P_k(x,y)$'s. (P_k must, at least, vanish at the singularities.)

Algorithm

Based on observation by Mñuk: let

$$\mathcal{O}_{A(X)} = \overline{\mathbb{C}[x,y]/(f)} \subset \mathbb{C}(x,y).$$

Then,

$$\mathrm{Adj}(X) = \left\{ P(x,y) \in \mathbb{C}[x,y] \,\|\, \mathcal{O}_{A(X)} \cdot P(x,y) \subset \mathbb{C}[x,y] \right\}.$$

$\mathcal{O}_{A(X)}$ is finite dimensional over $\mathbb{C}[x,y]$. Let, $\{\beta_1, \ldots, \beta_m\}$ be a basis. Then Mñuk's theorem is equivalent to requiring

$$\beta_j P(x,y) \in \mathbb{C}[x,y], \forall j = 1, \ldots, m.$$

Algorithm for computing β_j is due to van Hoeij and uses Puiseux series.

Demo

Integral bases of algebraic functions fields and bases of the space of holomorphic differentials.

Table of Contents

Continuing y-Roots Along x-Paths

The y-roots / fibre $\mathbf{y} = \{y_1, \ldots, y_n\} \subset \mathbb{C}_y$ of

$$f(x,y) = \sum_{j=0}^{n} a_j(x) y^j = 0$$

are continuous as a function of $x \in \mathbb{C}_x$. Much of the "geometric side" involves continuing a fibre $\mathbf{y}$ along a path $\gamma \subset \mathbb{C}_x$.

- In particular, we select a "base point" $a \in \mathbb{C}_x$ and an ordering of the y-roots at that point. (y_j is on sheet j of the Riemann surface.)

Continuing y-roots Along x-Paths

Two-step process:

1. Taylor step.
    - A proper step size is determined based on the "movement" of all of the roots.

   Let $\mathbf{y^i} = (y_1^i, \ldots, y_n^i)$ be the fibre at $x_i \in \mathbb{C}_x$ and $dx_i = x_{i+1} - x_i$. Then

$$y_j^{approx} = y_j^i - dx_i \frac{f(x_i, y_j^i)}{\partial_y f(x_i, y_j^i)}$$

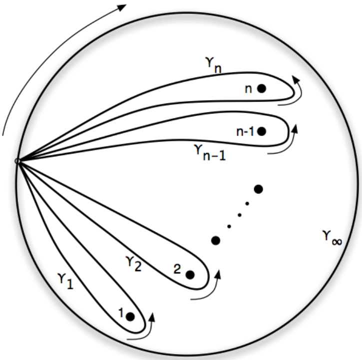
2. Compute roots y_j^{i+1} at x_{i+1}.
3. Determine if Taylor approximates y_j^{approx} are "close enough" to the new roots.
    - If so, match approximates with new roots and iterate.
    - If not, take smaller step and try again.

Monodromy Group

Let $b_1, \cdots, b_N \in \mathbb{C}_x^*$ be the branch points of f. The **monodromy group of** f is the fundamental group

$$\pi_1(\mathbb{C}_x^* \backslash \{b_1, \ldots, b_N\}, a).$$

where $a \in \mathbb{C}_x$ is a fixed point. (Not a branch point.)



Specifically, we compute the fibre $\mathbf{y} = \{y_1, \ldots, y_n\}$ at $x = a$ and see how the roots are permuted when we analytically continue around each branch point b_i.

Monodromy: Algorithm

1. Fix a base point $a \in \mathbb{C}_x$ and an ordering of the fibre $\mathbf{y}$ above it. (Fixed for all future comptuations.)
2. Fix an ordering of the branch points, b_i.
3. Compute a minimal spanning tree with the branch points as the nodes. The root is the branch point closest to a. Draw circular paths around each branch point.
   - For numerical accuracy purposes we stay away from the branch points.

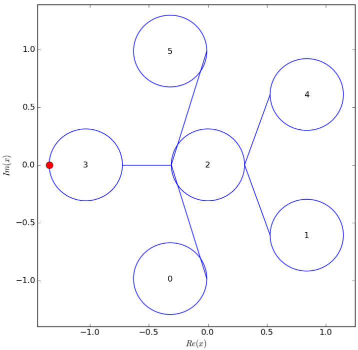


Figure: The collection of all x-paths over which we analytically continue the "base fibre".

Monodromy: Algorithm

4 Determine appropriate monodromy group paths, given the above ordering.
   - The path around branch point b_i must lie below the path around branch point b_{i+1}.

5 Analytically continue the base fibre along each path.

6 When returning to the base point, record how the fibre elements / sheets were permuted.

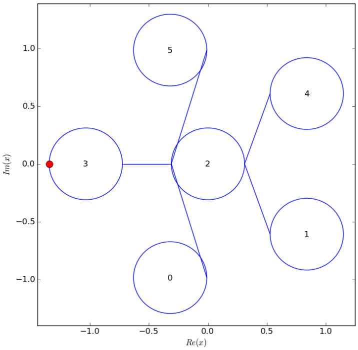


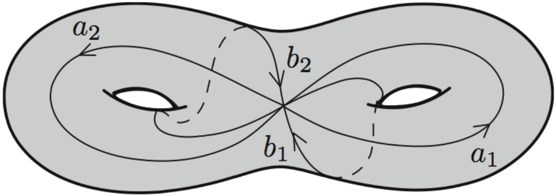
Figure: The collection of all x-paths over which we analytically continue the "base fibre".

Demo

The monodromy group $\pi_1\left(\mathbb{C}_x^*\backslash\{b_0,\dots,b_N\},a\right)$

Homology $H_1(X, \mathbb{Z})$

Monodromy says how to get from one sheet to another on X by analytically continuing along closed paths in $\mathbb{C}_x$. But we want closed paths on the Riemann surface, itself.



Algorithm due to Tretkoff and Tretkoff takes monodromy information and returns a- and b-cycles.

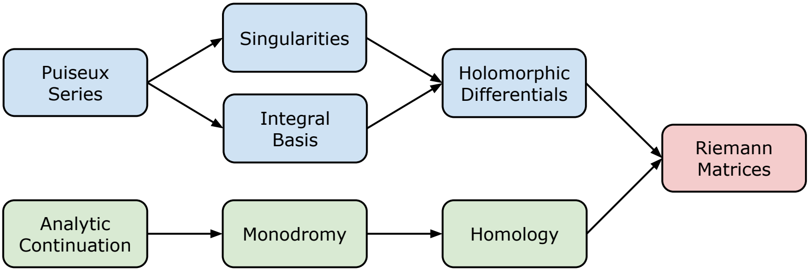
- "Base place" $P_0 = (a, y_1)$.
- Linear combinations of intermediate "c-cycles".

Demo

Homology basis of X: $H_1(X, \mathbb{Z})$

Table of Contents

Riemann Matrices: Putting It Together



Riemann Matrices

Compute

$$A_{ij} = \oint_{a_i} \omega_j \quad B_{ij} = \oint_{b_i} \omega_j$$

using Gauss-Legendre quadrature. In particular, parameterize the path $\gamma \in H_1(X, \mathbb{Z})$ by $t \in [0,1]$ and compute

$$\int_\gamma \omega = \int_0^1 \omega\left(x(t), y(x(t))\right) x'(t) dt.$$

Set,

$$\Omega = A^{-1} B.$$

(This Ω is equal to that which is chosen by normalizing our basis of holomorphic differentials.)

Demo

Period matrices and Riemann matrices.

Riemann Theta Functions

Building block of Abelian functions: $\theta : \mathbb{C}^g \times \mathfrak{h}_g \to \mathbb{C}$.

$$\theta(z, \Omega) = \sum_{n \in \mathbb{Z}^g} e^{2\pi i \left(\frac{1}{2} n \cdot \Omega n + n \cdot z \right)}$$

Algorithm by Deconinck, *et. al.*: separates doubly exponentially growing part and "oscillatory part".

- Fast Python + C implementation. (Multiprecise on the way.)
- Optional CUDA-enabled implementation by S. and Williams.

Riemann Theta Functions: Example

We plot real and imaginary parts of the oscillatory part of

$$\theta\left([x+iy,0,0],\Omega\right)$$

for $x \in [0,1], y \in [0,3]$ with $N = 65536$ $z = [x+iy,0,0]$ values and for

$$\Omega = \begin{pmatrix} -\frac{1}{2}+i & \frac{1}{2}-\frac{1}{2}i & -\frac{1}{2}-\frac{1}{2}i \\ \frac{1}{2}-\frac{1}{2}i & i & 0 \\ -\frac{1}{2}-\frac{1}{2}i & 0 & i \end{pmatrix}$$

Demo

CUDA evaluation of a slice of a genus $g = 3$ Riemann theta function.

Thank You

email
cswiercz@uw.edu

abelfunctions
https://github.com/cswiercz/abelfunctions

Future Work

- Performance improvements.
  - csympy by Ondřej Čertík.
  - Use Cython and GMP for numerical portions.
- Abel Map $A : X \to \text{Jac}(X)$

$$A(P) = \left(\int_{P_0}^{P} \omega_1, \ldots, \int_{P_0}^{P} \omega_g \right) \tag{3}$$

  (Ph.D. thesis work of Matthew Patterson.)
- Vector of Riemann Constants: $2K \equiv -A(C)$ where C is the *canonical class* divisor. (The equivalence class of all divisors of holomorphic differentials.)
  (Ph.D. thesis work of Matthew Patterson.)

Future Work

- Fay's Prime Form: Let $\alpha, \beta \in [0,1)^g$ be such that

$$\left(\frac{\partial}{\partial z_1}\theta[\alpha,\beta](0,\Omega), \ldots, \frac{\partial}{\partial z_g}\theta[\alpha,\beta](0,\Omega)\right) \neq \mathbf{0}.$$

Define the holomorphic differential

$$\zeta = \sum_{j=1}^{g} \omega_j \frac{\partial}{\partial z_j}\theta[\alpha,\beta](0,\Omega).$$

Then the prime form $E : X \times X \to \mathbb{C}$ is defined by

$$E(P_1, P_2) = \theta[\alpha,\beta]\left(\int_{P_1}^{P_2} \omega, \Omega\right) \bigg/ \sqrt{\zeta(P_1)}\sqrt{\zeta(P_2)}$$

where $\omega = (\omega_1, \ldots, \omega_g)$.