

MAT454 Notes

Jad Elkhaleq Ghalayini

March 4 2020

Today, our goal is to prove the Riemann mapping theorem:

Theorem 1 (Riemann mapping theorem). *Any simply connected open $\Omega \subset \mathbb{C}$ except \mathbb{C} itself has a biholomorphic mapping onto the open unit disc D*

We will begin by proving a series of lemmas.

Lemma 1. *There is a biholomorphism of Ω onto a bounded open subset of \mathbb{C}*

Proof. Let $a \notin \Omega$ be a point, which exists as $\Omega \neq \mathbb{C}$. Then $\frac{1}{z-a}$ is a nonvanishing holomorphic function on the simply connected open set Ω , and so it has a primitive, some holomorphic function $g(z)$.

Now, a primitive of $\frac{1}{z-a}$ is like a branch of $\log(z-a)$, which means that

$$z - a = e^{g(z)}$$

One thing this tells us right away is that $g(z)$ is one to one, because $z-a$ is one to one and if the composition of a function with something else is one to one, then that function must be one to one.

Take a point $z_0 \in \Omega$. Since Ω is open and g is one to one, implying it is nonconstant, there is an open disc centered at $g(z_0)$ inside $g(\Omega)$. Now, I claim that if you look at this disc translated by $2\pi i$ it's outside of $g(\Omega)$, i.e. $E + 2\pi i \cap g(\Omega) = \emptyset$. Intuitively, this is the case because it's on a different branch of the log function. A cleaner way of saying this is that this is because $\exp \circ g$ is one-to-one, but \exp will take two translated points to the same point, yielding a contradiction. So then, how do we get a biholomorphism of Ω onto a bounded open subset of \mathbb{C} ? Well, we have that

$$h(z) = \frac{1}{g(z) - (g(z_0) + 2\pi i)}$$

is one to one and bounded on Ω . As g and h are biholomorphisms, there is a biholomorphism $h \circ g$ from Ω onto a bounded open set. \square

So as we have a biholomorphism from Ω to a bounded open subset of \mathbb{C} , we can assume $0 \in \Omega \subset D$ by simply making a translation and scaling appropriately. We're interested in defining a convenient normal family now, but it's going to come about in a very natural way. Let's look at the set of functions that are holomorphic in Ω and are biholomorphisms into D taking the origin to itself, i.e. let

$$\mathcal{A} = \{f \in \mathcal{H}(\Omega) : f \text{ is one to one, } f(0) = 0, |f(z)| < 1\}$$

We're going to find the element of this set with the largest possible derivative at the origin, which is going to force the image to be as large as possible. We'll show that that means it must be all of D . We first show that the maximum derivative at zero is actually taken on

Lemma 2. $\sup_{f \in \mathcal{A}} |f'(0)|$ is attained.

Proof. We note that the function from $\mathcal{H}(\Omega) \rightarrow \mathbb{C}$ taking f to $|f'(0)|$ is continuous. Hence, taking

$$\mathcal{B} = \{f \in \mathcal{A} : |f'(0)| \geq 1\} \supseteq \{\text{id}\} \neq \emptyset$$

it is enough to show that \mathcal{B} is compact. \mathcal{B} is a normal family, i.e. locally bounded, as it is bounded uniformly on all of Ω . That means the only thing we really have to show is that \mathcal{B} is closed, i.e.,

$$f \in \mathcal{H}(\Omega), \exists f_n \in \mathcal{B}, f = \lim_{n \rightarrow \infty} f_n \implies f \in \mathcal{B}$$

We trivially have that

$$f(0) = \lim_{n \rightarrow \infty} f_n(0) = \lim_{n \rightarrow \infty} 0 = 0$$

$$|f'(0)| = \lim_{n \rightarrow \infty} |f'_n(0)| \geq 1 \text{ since } [1, \infty) \text{ is closed}$$

So f is one to one because it's not constant. Then the only other thing to prove is $|f(z)| < 1$. So what about that? We know that $|f(z)| \leq 1$ since $(-\infty, 1]$ is closed, but $f(z) \neq 1$ at every point $z \in \Omega$ by the maximum modulus principle (as if it was this would imply f was constant, contradicting both the fact that it is one-to-one and that $f(0) = 0$). \square

We prove the second part of the above statement, completing the theorem

Lemma 3. *Let $g \in \mathcal{A}$. Then $g(\Omega) = D$ if and only if*

$$|g'(0)| = \sup_{f \in \mathcal{A}} |f'(0)|$$

Proof. • “Only if”: suppose

\square