Computing Real Derivatives Via the Complex Plane: APPM 4360 project report

Maedée Trank-Greene[1], Logan Barnhart[1], and Jonathan Shaw[1]

[1]University of Colorado Boulder, Department of Applied Mathematics

May 1, 2023

Abstract

Motivated by the desire to accurately numerically differentiate analytic functions with unwieldy derivatives, we investigate the complex-step derivative approximation introduce by Martins et. al. [3]. The background of this method is grounded in the theory developed by Lyness and Moler, which considers approximating the derivative of a real-valued function by generalizing the argument of the function to the complex plane [2]. This theory is briefly presented, along with an example highlighting its practical limitations. Two applications of the complex-step derivative approximation are included as well, indicating the method's usefulness for approximating the first derivative of functions of one variable, although the method is not necessarily preferable in other applications, such as gradient descent.

1 Introduction

Robust and computationally cheap derivative calculations are important in many areas of science and engineering; in particular, sensitivity analysis of differential dynamical systems may benefit significantly from accurate numerical derivative approximations [3]. Continuous-time systems arising in nature are often too complicated for sensitivity analysis by hand and require approximating the derivatives of governing equations numerically. Finite difference methods offer easy implementation but can often be inaccurate because of floating point arithmetic related errors, and more sophisticated methods which do not fall victim to these kinds of errors may be too difficult to apply to certain systems. Another pitfall of finite difference methods is they decay very rapidly when trying to examine derivatives beyond the first order. In this paper, we review methods that utilize complex analysis to accurately calculate first and higher order derivatives along the real axis.

2 Using the complex plane to retrieve real derivative information

2.1 Generalizing real-valued functions to the complex plane: addressing the pitfalls of finite difference approximations

If the nth derivative of a C^n function of one variable is difficult to obtain analytically, the function's derivative at a discrete set of points may be approximated using numerical methods. Such functions certainly exist [2] and can present significant difficulties, depending on the required accuracy of the derivative approximation. A common scheme for approximating the derivative a function $f : \mathbb{R} \to \mathbb{R}$ at some point x is finite differences, which follows from Taylor's Theorem:

$$f(x+h) = f(x) + f'(x)h + \mathcal{O}\left(h^2\right) \Rightarrow f'(x) = \frac{f(x+h) - f(x)}{h} + \mathcal{O}\left(h\right).$$

While the approach above promises infinite accuracy as $h \to 0$, this is not the case in practice. As $h \to 0$, $f(x+h)$ and $f(x)$ are of similar order, and when subtracting the values using floating point arithmetic, significant roundoff errors are introduced. Lyness and Moler lay the theoretical groundwork for resolving this issue by generalizing the argument of f to the complex plane.

2.2 Deriving an analytic approximation of the n^{th} derivative

The theory presented by Lyness and Moler [2] is extensive, so only their primary result is outlined here. It is important to note that the analysis conducted is centered around the origin, but extending this to other points along the real axis is uncomplicated.

For a function $f : \mathbb{R} \to \mathbb{R}$, allow its argument to be generalized to the complex plane and introduce

$$g(r;t) = g(t) = \mathrm{Re} f(re^{i2\pi t}).$$

For a given function, $g(t)$, Let $R^{[m,1]}$ be the trapezoidal Riemann sum operator defined by

$$R^{[m,1]}g(t) = \frac{1}{m}\left[\frac{1}{2}g(0) + g(\frac{1}{m}) + \cdots + g(\frac{m-1}{m}) + \frac{1}{2}g(1)\right].$$

Additionally, let μ_n be the n^{th} Mobius Number defined by

$$\mu_n = \begin{cases} 1 & \text{if } n = 1, \\ (-1)^k & \text{if } n \text{ is a product of } k \text{ distinct prime numbers,} \\ 0 & \text{otherwise.} \end{cases}$$

The main conclusion from Lyness and Moler's work was the following theorem:

Theorem. If $f(x)$ is a real function of x and f(z) is analytic in a neighborhood that contains the circle $|z| = r$, then

$$a_n = \frac{f^{(n)}(0)}{n!} = \frac{1}{r^n} \sum_{m=1}^{\infty} \mu_m [R^{[mn,1]} g(t) - f(0)]. \tag{1}$$

A proof of this theorem has been included in subsection (5.1) of the appendix.

(Aside: although the interest lies in finding the n^{th} derivative of a function, it will be helpful in the analysis later to think of the problem as finding the n^{th} Taylor Series coefficient, which is why the notation a_n is adopted.)

2.3 Deriving the complex-step derivative approximation

Lyness and Moler [2] had shown that valuable information about the real axis can be extrapolated from the complex plane, but numerically their method can still be expensive, especially for lower order derivatives.

To improve upon this and the drawbacks of finite difference methods discussed in 2.1, consider generalizing the argument of a function $f : \mathbb{R} \to \mathbb{R}$ such that

$$f(z) = u(z) + iv(z).$$

Letting $z = x + iy$, we obtain

$$f(x + iy) = u(x + iy) + iv(x + iy).$$

Assuming that f is analytic, the Cauchy-Riemann equations hold. That is,

$$\begin{cases} u_x = v_y, \\ u_y = -v_x. \end{cases}$$

Using the first of the equations above and the formal definition of a derivative, we can write:

$$\frac{\partial u}{\partial x} = \lim_{h \to 0} \frac{v(x, i(y + h)) - v(x, iy)}{h}$$

Note that since the starting function was exclusively real-valued and we are focused on derivatives on the real axis, we can say that $f(x) = u(x)$, $y = 0$ and $v(x, 0) = 0$. Using these facts in the above expression reveals

$$f'(x) = \lim_{h \to 0} \frac{\mathrm{Im}[f(x + ih)]}{h}$$

Thus, we can approximate the derivative of f using a fixed h by using the following,

$$f'(x) \approx \frac{\mathrm{Im}[f(x + ih)]}{h}, \tag{2}$$

and note the approximation of $f'(x)$ does not suffer from floating point arithmetic-related errors.

2.3.1 Error analysis of the complex step approximation

Not only does the previous method not suffer from roundoff error, but it actually converges to the true derivative faster as h tends to zero. The same analysis as in Squire and Trapp [4] is presented below.

If a real function, $f(x)$, has an analytic extension to the complex plane, $f(z)$, then its Taylor series expansion about a real value a can be written as

$$f(x) = f(a) + (x-a)f'(a) + (x-a)^2\frac{f''(a)}{2} + (x-a)^3\frac{f'''(a)}{3!} + \dots$$

If we let $a = x$ and evaluate f at an imaginary value, $z = x + ih$, then we can see the expansion become

$$f(x+ih) = f(x) + ihf'(x) - h^2\frac{f''(x)}{2} - ih^3\frac{f'''(x)}{3!}\dots$$

If we take the imaginary part of both sides of the equation and divide by h, we get

$$\frac{\mathrm{Im}[f(x+ih)]}{h} = f'(x) - h^2\frac{f'''(x)}{3!}\dots$$

or, solving for $f'(x)$:

$$f'(x) = \frac{\mathrm{Im}[f(x+ih)]}{h} + h^2\frac{f'''(x)}{3!}\dots$$

Thus the error in the complex-step derivative approximation is $\mathcal{O}(h^2)$. So, not only does this method not suffer from round-off error, but it is also measurably more accurate than other strictly real methods.

3 Numerical experiments

In the following section, we investigate the stability of Lyness and Moler's theorital work and present two example applications of the complex-step derivative approximation introduced in [3].

3.1 Approximating the n^{th} derivative

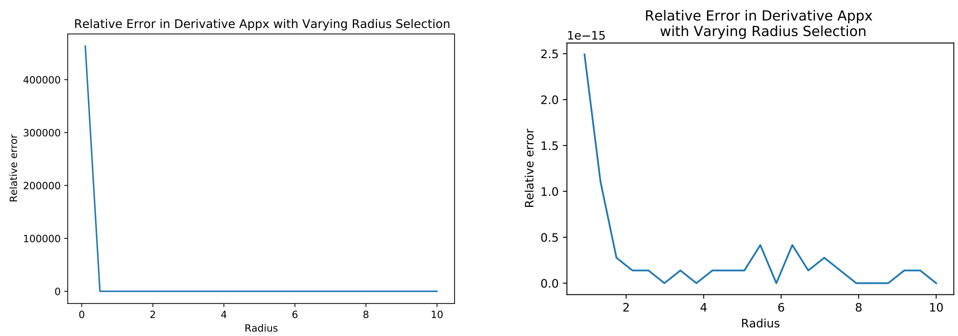
The expression introduced by Lyness and Moler in Theorem 1 in which a finite number of terms in the sum are evaluated is straightforward to implement. In the following, we explore the method's robustness through a small set of experiments.

Consider the following polynomial,

$$f(x) = x^{25} + 4x^{20} - 198x^{13} + 14x^4 - 2x^3, \tag{3}$$

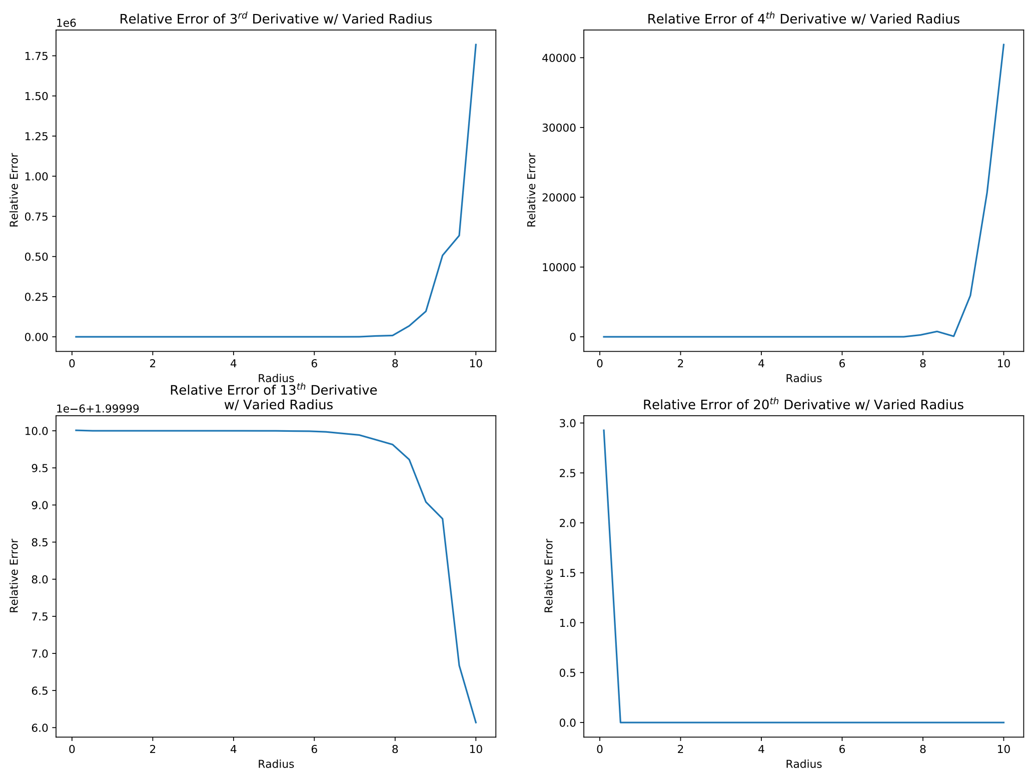
whose n^{th} order derivatives are easy to obtain, and note that the n^{th} derivative evaluated at $x = 0$ is either 0 if n is not one of the powers present in f, or it is $c_n n!$ where c_n is the coefficient of the n^{th} degree term in f.

We conduct an analysis of Lyness and Moler's method, and attempt to determine the method's sensitivity to radii choices and the number of terms retained in the sum, as well as accuracy for high order derivatives. The following figures illustrate the effects of radii choice for the $f^{(25)}(0)$ case, using the first 15 terms in the sum (see (1)).



The most notable thing is that while the method did yield some approximations that were within machine-epsilon of the correct answer, many radii selections resulted in approximations with tremendously large relative errors, as depicted above.

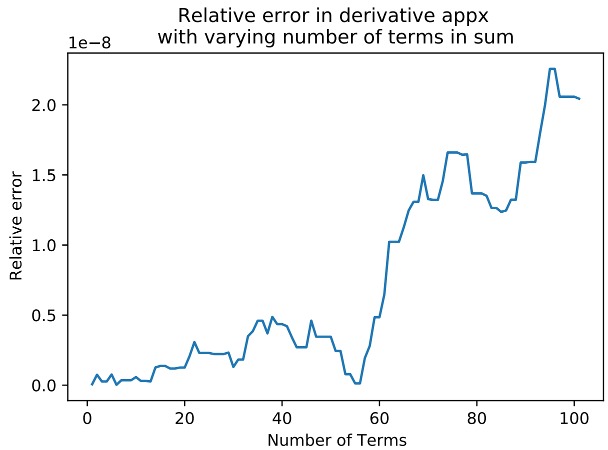
While relative error as a function of the chosen radius is important to consider, the order of the derivative should also be considered. The following figures illustrate the unpredictable nature of the relative error as a function of the radii for several choices of derivative orders.



It should be noted that most of the figures are accurate representations of performance *except* for the 13^{th} derivative. The scaling is skewed to show improvement, when in reality each relative error is within 6 digits of 2.0.

While there are some similar trends - for example, the figures for the 3^{rd} and 4^{th} derivatives and the 20^{th} and 25^{th} derivatives have similar shapes - it does not appear that there are any reliable changes in behavior as we scale upwards. Somehow we jump from relative error of 40000 in $f^{(4)}$ back down to a largest relative error of 3 in $f^{(20)}$.

Since this did not seem to guide us to a reliable way to calculate derivative approximations, we wondered if the variance was due to not using enough terms in the calculation. To assess this, we evaluated $f^{(25)}$ at a fixed radius - one which yielded moderate error in the first figure - and varied the number of terms retained in the sum from 1 to 100.



Again, we see a large amount of variance. In normal sum-dependant quadrature such as Riemann sums, an estimate generally improves as you include more terms, but interestingly the first term was highly accurate, while including a hundred terms was significantly less accurate.

There is one real takeaway from these results: this method's accuracy is too unpredictable to calculate higher order derivatives, as its performance is varies widely for a given derivative order as a function of the chosen radius and number of terms retained in the sum.

3.1.1 Improving recovery of the n^{th} derivative:

Analytically, the expression in Theorem 1 is very useful and marked an important point about the information that lies in the complex plane, but as we have seen, it suffers from numerical instabilities.

In 1981, B. Fornberg implemented an alternative approach that employs Fast Fourier Transforms and Richardson extrapolations to optimize the chosen radius r and the number of terms retained in the infinite sum, addressing the instabilities discussed in the previous section [1]. His method was not implemented here, however, as we only focus on the first derivative of analytic functions for the remainder of the paper. We encourage the reader to examine the work done by Fornberg in [1], as the algorithm presented allows for n^{th} order derivatives at arbitrary x locations to be calculated up to machine precision for $n = 1, 2, \ldots, 50$, and is very impressive.

3.2 The Complex Step Approximation

In the following section, we present two applications of the complex-step derivative approximation. In the first example, we select a function of one variable whose derivative is easily calculated to verify the effectiveness of the method, while the second example applies the method to a gradient descent problem.

3.2.1 Method verification: a function of one variable

The following numerical example will illustrate the difference in relative error from computing derivatives with finite difference methods and complex step derivatives with the following analytic function

$$f(x) = \frac{e^x}{\sin(x)^7 + \cos(x)^7}. \tag{4}$$

In this example, we computed the derivative at $x = 1.5$ and for the finite difference methods, we chose to only include the forward and central finite difference methods, as the backward difference was nearly identical to the forward difference.

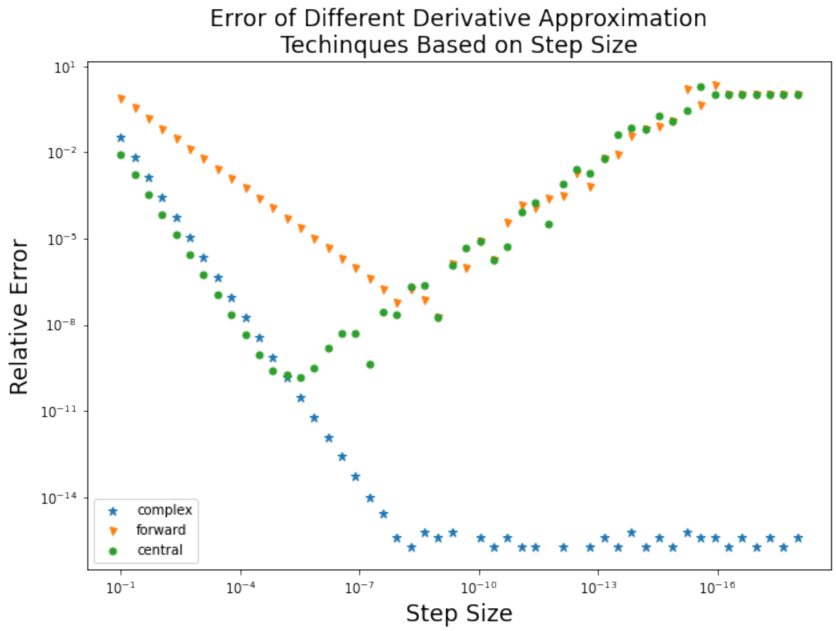
The following figure compares the relative error of the three methods as the step size decreased in size. The x-axis is inverted to better show the decrease in error as the step size decreases.

For reference, below is the exact derivative of $f(x)$.

$$f'(x) = \frac{e^x(\sin(x)^7 + \cos(x)^7) - e^x(7\sin(x)^6\cos(x) - 7\cos(x)^6\sin(x))}{(\sin(x)^7 + \cos(x)^7)^2} \tag{5}$$

The exact derivative of $f(x)$ at the point $x = 1.5$ was used to compute the relative error of the approximation methods.

As can be seen in the figure, the complex step derivatives retain an accuracy in approximating the derivative of $f(x)$ at $x = 1.5$ as the step size decreases. It decays quickly until around a step size of 10^{-8} and levels out to a relative error of 10^{-14}. The central finite difference decreases at the same rate as the complex step derivative until around a step size of 10^{-5}, and then jumps back up and levels out to a relative error of around 10 for smaller step sizes. Similar to the central finite difference, the forward finite difference decays until around a step size of 10^{-8} where it jumps back up and levels out with the central finite difference.



From the above example, it is clear to see how the complex step derivative is an optimal method when approximating derivatives, as it retains a small relative error as the step size decreases opposite to the finite difference methods.

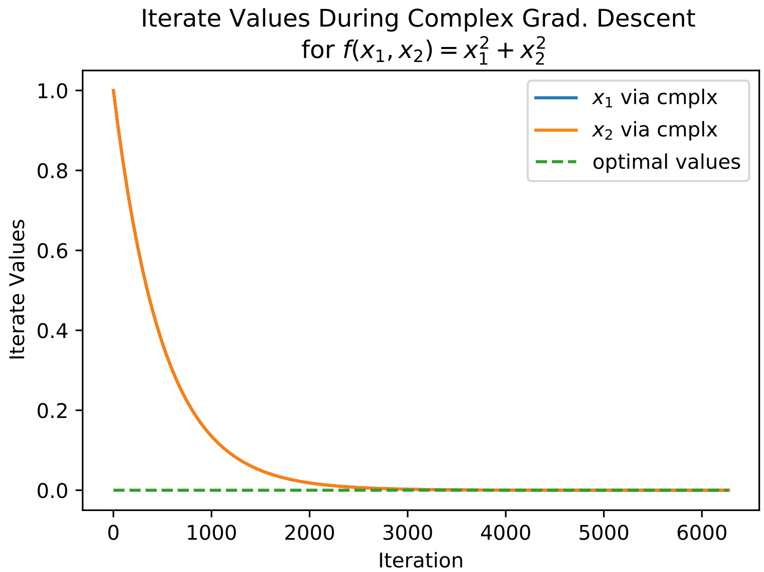
3.2.2 Applying the complex step to gradient descent

The final numerical experiment we wanted to explore was applying the complex step approximation to a frequently used application, gradient descent.

Our first test was on the relatively simple convex function

$$f(x_1, x_2) = x_1^2 + x_2^2.$$

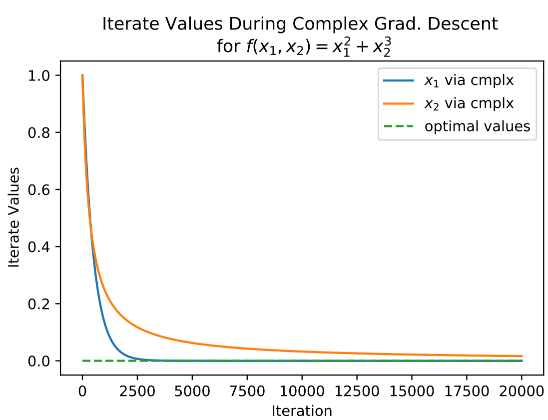
There are no difficult valleys or local minima to navigate around, and complex gradient descent converged to the optimal solution with high accuracy in a reasonable number of iterations in both dimensions.

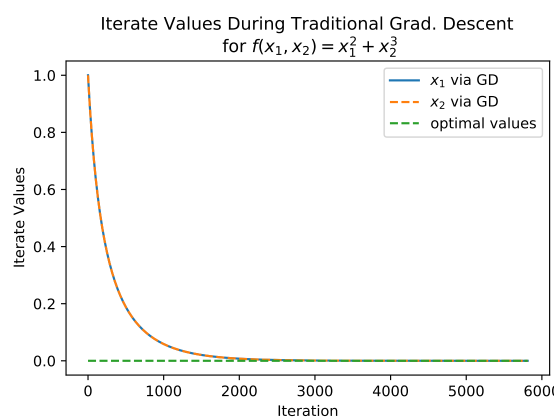


As nice, convex functions rarely occur in practice, so we additionally consider applying the complex-step to a non-convex function, given by

$$f(x_1, x_2) = x_1^2 + x_2^3.$$

While this function does not seem particularly complicated, poor initial guesses will result in gradient descent diverging, due to the cubic term, as $\min f = -\inf$, now. Given such a function, the goal is to converge to the saddle point - where the gradient is identically zero. Given the right initialization, true gradient descent will converge, so we wanted to see how the complex approximation performed.





In the convex dimension, x_1, we converged at a similar rate compared to true gradient descent, but the non-convex variable seems to converge noticeably slower. This is intriguing, because this means for some reason, the gradient supplied by the complex step approximation is an underestimate of the true gradient.

4 Conclusion

We have seen that valuable information about real-valued functions can be extrapolated from surrounding behavior on the complex plane. This is incredibly valuable if it can be extracted in an efficient manner. More rudimentary methods, like a pure implementation of the theorem from Lyness and Moler's paper [2], were not robust enough to be applied to many real-world problems. Thanks to improvements such as Fornberg's reimplementation [1] and a simplified way to calculate the first derivative [3], we have been able to reliably use the complex plane for real world tasks.

At a higher level, this can be used to computationally, or analytically simplify gradient descent for "well-behaved" functions, but it can be implemented in ways to efficiently solve difficult optimization problems [3] and may even have extensions in numerically solving ODEs. The complex step approximation has a swathe of potential applications, and although it seems less practical, easy-to-compute high order derivatives can be very valuable when using more sophisticated implementations.

References

[1] Bengt Fornberg. "Numerical Differentiation of Analytic Functions". In: *ACM Transactions on Mathematical Software* 7.4 (Dec. 1981), pp. 512–526. URL: `https://www.colorado.edu/amath/sites/default/files/attached-files/acm_81_1.pdf`.

[2] J. N. Lyness and C. B. Moler. "Numerical Differentiation of Analytic Functions". In: *SIAM Journal on Numerical Analysis* 4.2 (1967), pp. 202–210. DOI: `10.1137/0704019`. eprint: `https://doi.org/10.1137/0704019`. URL: `https://doi.org/10.1137/0704019`.

[3] Joaquim Martins, Peter Sturdza, and Juan Alonso. "The Complex-Step Derivative Approximation". In: *ACM Transactions on Mathematical Software* 29 (Sept. 2003), pp. 245–262. DOI: `10.1145/838250.838251`.

[4] William Squire and George Trapp. "Using Complex Variables to Estimate Derivatives of Real Functions". In: *SIAM Review* 40.1 (1998), pp. 110–112. URL: `https://researchrepository.wvu.edu/cgi/viewcontent.cgi?article=1425&context=faculty_publications`.

5 Appendix

5.1 Proof of (1)

Cauchy's integral formula can be generalized to the n^{th} derivative evaluated at $z = 0$ as follows

$$a_n = \frac{f^{(n)}(0)}{n!} = \frac{1}{2\pi i} \int_C \frac{f(z)}{(z)^{n+1}} dz. \tag{6}$$

Letting the contour C be the circle centered at the origin with radius r, or $|z| = r$, and using the direct substitution $z = re^{2\pi i\theta}$, (6) becomes

$$a_n = \frac{1}{r^n}\int_0^1 f(re^{2\pi i\theta})e^{2\pi in\theta}d\theta \tag{7}$$

and notice that for $n = 0$,

$$a_0 = f(0) = \int_0^1 f(re^{2\pi i\theta})d\theta.$$

Now letting $g(z) = z^n f(z)$ and looking at the first Taylor coefficient of g using this formula,

$$g(0) = 0 = \frac{1}{r^n}\int_0^1 f(re^{2\pi i\theta})e^{2\pi in\theta}d\theta \tag{8}$$

Equations (3) and (4) can now be combined to have one of the following two forms:

$$a_n = \frac{2}{r^n}\int_0^1 f(re^{2\pi i\theta})\cos 2\pi n\theta d \tag{9}$$

$$a_n = \frac{-2i}{r^n}\int_0^1 f(re^{2\pi i\theta})\sin 2\pi n\theta d \tag{10}$$

The above expressions for a_n closely resemble the coefficients of a Fourier series, which is why the Poisson summation formula can be used to relate the trapezoidal Riemann sum operator with the expressions as follows

$$R^{[n,1]}f(re^{2\pi i\theta}) = \sum_{k=-\infty}^{\infty}\int_0^1 f(re^{2\pi i\theta})e^{2\pi ink\theta}d\theta \tag{11}$$

$$= \int_0^1 f(re^{2\pi i\theta})d\theta + 2\sum_{k=1}^{\infty}\int_0^1 f(re^{2\pi i\theta})\cos 2\pi ink\theta d\theta \tag{12}$$

Now letting b_n be the signed error in the trapezoidal sum, or

$$b_n = R^{[n,1]}f(re^{2\pi i\theta}) - \int_0^1 f(re^{2\pi i\theta})d\theta \tag{13}$$

$$= R^{[n,1]}f(re^{2\pi i\theta}) - f(0) \tag{14}$$

And using (8) combined with the earlier expression for a_0, b_n is just

$$= \sum_{k=1}^{\infty} 2\int_0^1 f(re^{2\pi i\theta})\cos 2\pi nk\theta d\theta \tag{15}$$

$$= \sum_{k=1}^{\infty} r^{kn}a_{kn} \tag{16}$$

Now consider (12) for $n \in \mathbb{N}$ - inverting the resulting *set* of equations yields

$$r^n a_n = \sum_{k=1}^{\infty}\mu_k b_{kn}$$
$$\forall n \in \mathbb{N}$$

Where μ_l is the l^{th} Mobius number.

If the expression on the right hand side of (10) is substituted in for each b_n in the above equation, both sides are divided by r^n, and the real part is taken (since $f(x)$ is a real function) then one arrives directly at Theorem 1.