Application of the Argument Principle to Functions Expressed as Mellin Transforms

Bjoern S. Schmekel *

Department of Physics, College of Studies for Foreign Diploma Recipients at the University of Hamburg, 20355 Hamburg, Germany

January 15, 2021

Abstract

We describe a numerical algorithm for evaluating the numbers of roots minus the number of poles contained in a region based on the argument principle with the function of interest being written as a Mellin transformation of a usually simpler function. Because the function to be transformed may be simpler than its Mellin transform whose roots are to be sought we express the final integrals in terms of the former accepting higher dimensional integrals. Nonlinear terms are expressed as convolutions approximating reciprocal values by exponential sums. As an example the final expression is applied to the Riemann Zeta function. The procedure is very inefficient numerically. However, depending on the function to be investigated it may be possible to find analytical estimates of the resulting integrals.

Keywords — Root-Finding Algorithms, Argument Principle, Mellin Transformation, Riemann Zeta Function

1 Introduction

Object of this paper is to compute the number of roots minus the number of poles enclosed by a closed contour $C$ using the argument principle

$$N_R - N_P = \frac{1}{2\pi i} \oint_C \frac{f'(s)}{f(s)} ds$$

(1.1)

where $f(z)$ is a meromorphic function on and inside of the contour $C$ which can be represented as an auxiliary function multiplied by the Mellin transform of another function

$$f(s) = K(s)Z(s) = K(s) \int_0^\infty z(t)t^{s-1} dt$$

(1.2)

assuming the Mellin transform exists on and inside of the contour $C$. The latter is chosen such that it does not run over any poles or roots. We are interested in cases where $z(t)$ is a simple function therefore expressing the final result in terms of $K(s)$ as well as $z(t)$ instead of $Z(s)$. Furthermore, we are looking for an expression such that eqn. [1.1] can be expressed as a multi-dimensional integral of $z(t)$. Obviously, the integrand in eqn. [1.1] is a nonlinear functional of $z(t)$ (and $K(s)$), so arriving at such a result is not completely straightforward. In a first step we deal with the nonlinearities by approximating the reciprocal value of $f(s)$ in the argument principle by an exponential sum $\sum_{j=1}^{N} \alpha_j e^{-\epsilon_j x}$, i.e.

$$\frac{1}{x} \approx \sum_{j=1}^{N} \alpha_j e^{-\epsilon_j x} \equiv I_1(x)$$

(1.3)

which is possible for $\Re(x) > 0$. We will have to ensure this condition is always met possibly adding a factor which changes sign when appropriate.

The exponential sum approximation has not been investigated too thoroughly for complex denominators. In fig. 1 The approximation breaks down for small values of $\Re(x)$ as expected, but accuracy is not impacted by an imaginary part as only $\Re(x) > 0$ is needed for convergence.

* bss28@cornell.edu
Figure 1: Real (above) and imaginary part (below) of the difference of $1/z$ and its exponential sum approximation with $z = x + iy$
Using the following complex sign function
\[
csgn(s) = \begin{cases} 
-1 & \Re(s) < 0 \\
1 & \Re(s) > 0 \\
sgn(\Im(s)) & \Re(s) = 0 
\end{cases}
\] (1.4)
and expanding the exponential function as a power series we obtain
\[
N_R - N_P = \frac{1}{2\pi i} \oint_C ds \left[ K^s(s)Z(s) + K(s)Z'(s) \right] \sum_{j=1}^{N} \sum_{k=0}^{n} \alpha_j csgn(f(s)) \frac{(-1)^k}{k!} c_j^k K^k(s)Z^k(s)csgn^k(f(s)) 
\]
\[
= \frac{1}{2\pi i} \sum_{j=1}^{N} \sum_{k=0}^{n} \oint_C ds csgn^{k+1} (f(s)) \frac{(-1)^k}{k!} c_j^k K^j(s)K^k(s)Z^{k+1}(s) 
\]
\[
+ \frac{1}{2\pi i} \sum_{j=1}^{N} \sum_{k=0}^{n} \oint_C ds csgn^{k+1} (f(s)) \frac{(-1)^k}{k!} c_j^k K^{k+1}(s)Z^k(s)Z^k(s) = \int_0^{2\pi} d\phi K(\phi) 
\] (1.5)
The powers of \(Z(s)\) can be expressed in terms of \(z(t)\) using the Mellin convolution theorem
\[
Z(s) = \int_0^\infty dt z(t) t^{s-1} 
\] (1.6)
\[
Z^2(s) = \int_0^\infty dt \int_0^\infty du_1 z(u_1) z \left( \frac{t}{u_1} \right) u_1^{-1} t^{s-1} 
\] (1.7)
\[
Z^3(s) = \int_0^\infty dt \int_0^\infty du_1 \int_0^\infty du_2 z(u_1) z \left( \frac{u_2}{u_1} \right) \left( \frac{t}{u_2} \right) u_1^{-1} u_2^{-1} t^{s-1} 
\] (1.8)
\[
Z^k(s) = \int_0^\infty dt \int_0^\infty du_1 \ldots \int_0^\infty du_{k-1} t^{k-1} z(u_1) z \left( \frac{t}{u_{k-1}} \right) u_1^{-1} \prod_{j=1}^{k-2} z \left( \frac{u_{j+1}}{u_j} \right) u_j^{-1} t^{s-1} 
\] (1.9)
In appendix B a short Maple program is given which can be used to test the formulas given above. Similarly, exploiting standard rules for the Mellin transform
\[
Z'(s) = \int_0^\infty dt \ln(t) z(t) t^{s-1} 
\] (1.10)
\[
Z'(s)Z(s) = \int_0^\infty dt \int_0^\infty du_1 \ln(u_1) z(u_1) z \left( \frac{t}{u_1} \right) u_1^{-1} t^{s-1} 
\] (1.11)
If the behavior of the csgn-function is non-trivial for the function to be investigated it may be approximated continuously by
\[
csgn(x) \approx \tanh(x/\epsilon) 
\] (1.13)
with precision increasing as \(\epsilon \to 0\) where \(\tanh\) can be represented as
\[
\tanh(x) = -\frac{2i}{\pi} \int_0^\infty \frac{2t}{t^2 + 1} dt 
\] (1.14)
The integral converges if \(-\pi/2 < \Im(x) < 0\).

2 Example: Riemann Zeta Function

We use the zeta function in the form \[5\ 6\]
\[
\zeta(s) = \frac{2^{s-1}}{(1 - 2^{1-s}) \Gamma(s+1)} \int_0^\infty \frac{t^s}{\cosh^2(t)} dt 
\] (2.1)
which converges for \(\Re(s) > -1\). The zeta function has been investigated using the argument principle before by many authors \[7\]. The representation in eqn. \[2.1\] is written in the form of eqn. \[2.2\]. Since the factor \(K(s)\) in front of the integral has no roots or poles by itself with the exception of the known pole at \(s = 1\) it would be sufficient to set \(K(s) = 1\). However, for the present purpose the full factor is retained in order to stay in a numerically favorable range achieving sufficient accuracy in the exponential sum approximation.
The second column is the integrand and factor of eqn. 1.1 with the final approximation. Integration is performed on a circle with radius $\rho$.

In the third column 1

This is approximated further by eqn. 1.9 and 1.12. Mathematica code producing the results in table 1 can be found in appendix C.

Making use of the exponential sum approximation and the expansion of the exponential function in a power series. The powers could be expressed in terms of Mellin convolutions of the simpler function. Because of the high dimension of the involved integrals the method may not be feasible for high precision. However, since depending on the function of interest the integrands may be simple it may be possible to come up with analytical estimates which may or may not exclude roots in a given region.

### Table 1: $R = 0.1$ and $z_0 = 0.57 + 1.57i$

| $\phi/2\pi$ | $\int_0^{2\pi} \frac{f(z_0 + R e^{i\phi})}{z_0 - z_0^*} \, d\phi$ | $f(z_0 + R e^{i\phi})$ | $f'(z_0 + R e^{i\phi})$ | $f''(z_0 + R e^{i\phi})$ | $K(\phi)$ |
|-------------|---------------------------------|-----------------|-----------------|-----------------|-------------|
| 0           | 0.0124820 + 0.0040853i          | 0.0155503 + 0.0200828i | 0.0155503 + 0.0200828i | 0.0155503 + 0.0200828i | K(ϕ)       |
| 1/8         | 0.0062568 + 0.0016734i          | -0.0200876 + 0.0219640i | -0.0200876 + 0.0219640i | -0.0200876 + 0.0219640i |             |
| 2/8         | -0.0021237 + 0.0012357i         | -0.0141669 + 0.0148087i | -0.0141669 + 0.0148087i | -0.0141669 + 0.0148087i |             |
| 3/8         | -0.0015151 + 0.00081128i        | -0.0027191 + 0.0033209i | -0.0027191 + 0.0033209i | 0.0027191 + 0.0033209i  |             |
| 4/8         | -0.0011950 + 0.00081128i        | -0.0209646 + 0.0121417i | -0.0209646 + 0.0121417i | 0.0209646 + 0.0121417i  |             |
| 5/8         | -0.0008067 + 0.0022558i         | -0.0059628 + 0.0263608i | -0.0059628 + 0.0263608i | 0.0059628 + 0.0263608i  |             |
| 6/8         | 0.0037589 + 0.0048824i          | 0.0186236 + 0.0229791i | 0.0186236 + 0.0229791i | 0.0186236 + 0.0229791i  |             |
| 7/8         | 0.0128362 + 0.0049048i          | 0.0287771 + 0.0013062i | 0.0287771 + 0.0013062i | 0.0287771 + 0.0013062i  |             |
| 8/8         | 0.0124820 + 0.0040853i          | 0.0155503 + 0.0200828i | 0.0155503 + 0.0200828i | 0.0155503 + 0.0200828i  |             |

In table 1 we contrast the integrand in the argument principle with various steps towards the final approximation. Integration is performed on a circle with radius $R = 0.1$ around $z_0 = 0.57 + 1.57i$ not enclosing any roots. The second column is the integrand and factor of eqn. 1.1 with the final $d\phi$-integration missing. The coefficients used in the exponential sum approximation can be found in table 2 in the appendix. In the third column $1/f(z)$ is approximated by $I_1(z)$ which contains the exponential sum approximation. This is approximated further by $I_2(z)$ where the exponential function has been expanded in a power series up to linear order. Finally, in the fifth column the powers of $Z$ are expressed by Mellin convolutions given by eqn. 1.3 and 1.4. Mathematica code producing the results in table 1 can be found in appendix C.

Looking at the error introduced in each step we find that the expression by Mellin convolutions (cf. column 4 and 5 in table 1) works fairly well. The largest error is introduced by the exponential sum approximation which could be reduced by using more exponential terms (higher value of $N$ in eqn. 1.3). Ultimately, for arbitrarily high precision the number of terms in the expansion of the exponential function needs to be increased as well, though (higher value of $n$). Each new term introduces integrals of one more dimension which makes them increasingly hard to evaluate numerically.

### 3 Conclusions

We presented a method which evaluates the number of roots minus the number of poles enclosed in a region using the argument principle focusing on function which can be expressed as Mellin transforms of simple functions. The method was devised to work with the latter (simpler) function which was made possible by making use of the exponential sum approximation and the expansion of the exponential function in a power series. The powers could be expressed in terms of Mellin convolutions of the simpler function. Because of the high dimension of the involved integrals the method may not be feasible for high precision. However, since depending on the function of interest the integrands may be simple it may be possible to come up with analytical estimates which may or may not exclude roots in a given region.
4 Acknowledgments

We acknowledge support by Wolfram Research having provided assistance with Mathematica and free maintenance thereof.

References

[1] Wolfgang Hackbusch. Computation of best $l^\infty$ exponential sums for $1/x$ by remez’algorithm. Computing and Visualization in Science, 20, 1–11 (2019).

[2] William McLean. Exponential sum approximations for $t^{-\beta}$, 2016, 1606.00123.

[3] William McLean. Exponential sum approximations for $t^{-\beta}$. Contemporary Computational Mathematics - A Celebration of the 80th Birthday of Ian Sloan, page 911–930.

[4] Gregory Beylkin and Lucas Monzón. Approximation by exponential sums revisited. Applied and Computational Harmonic Analysis, 28, 131 – 149 (2010). Special Issue on Continuous Wavelet Transform in Memory of Jean Morlet, Part I.

[5] Michael S. Milgram. Integral and series representations of riemann’s zeta function, dirichelet’s eta function and a medley of related results, 2012, 1208.3429.

[6] Michael S. Milgram. Integral and series representations of riemann’s zeta function and dirichlet’s eta function and a medley of related results. Journal of Mathematics, 2013, 1–17 (2013).

[7] Tomas Johnson and Warwick Tucker. Enclosing all zeros of an analytic function — a rigorous approach. Journal of Computational and Applied Mathematics, 228, 418 – 423 (2009).

A Coefficients for the exponential sum approximation

| $i$ | $\alpha_i$ | $c_i$ |
|-----|-------------|------|
| 1   | 0.048       | 0.017|
| 2   | 0.235       | 0.139|
| 3   | 0.852       | 0.627|
| 4   | 2.737       | 2.241|

Table 2: Coefficients for the exponential sum approximation with values taken from [1]

B Maple Code Testing Eqn. [1.8]

The following Maple code computes the third power of the integral in eqn. 2.1 for $s = 0.4$ without the factor in front using eqn. 1.8 and by taking the third power directly. The results are $0.4875296028$ and $0.4875296044$, respectively. For $s = 0.4 - 0.3i$ we obtain $0.4103824778 + 0.1549090396i$ and $0.4103824766 + 0.1549090398i$, respectively.

```maple
z:= unapply(t/cosh(t)^2, t);
integrand:= unapply(z(u2/u1)*z(t/u2)*z(u1)/u1/u2, u1, u2, t);
expr1:= Int(integrand(u1, u2, t), u1=0..infinity);
expr2:= Int(expr1, u2=0..infinity);
Int(expr2*t^(-s-1), t=0..infinity);
subs(s=0.4,%);
evalf(%);
Zalt:= unapply(Zeta(s)*(1-2^(-s))*(1-2^(-s)))*GAMMA(s+1)/2^(-s-1), s);
(Zalt(0.4))^(1/3);
```

1 Code tested using Maple 2019.2 for Mac OS X.
C  Mathematica Code Producing Table 1

\[
\begin{align*}
\text{Z1} [s_n] &:= \text{NIntegrate} \left[ z[t] \star t^{-s} \right], \{t , 0 , \text{inf}\}, \text{WorkingPrecision} \rightarrow 50, \ \text{AccuracyGoal} \rightarrow 5 \\
\text{Z2} [s_n] &:= \text{NIntegrate} \left[ z[t/u1] \star z[u1] / u1 \star t^{-s} \right], \{u1 , 0 , \text{inf}\}, \{t , 0 , \text{inf}\}, \text{WorkingPrecision} \rightarrow 50, \ \text{AccuracyGoal} \rightarrow 5 \\
\text{ZpZ0} [s_n] &:= \text{NIntegrate} \left[ \text{Log} [t] \star z[t] \star t^{-s} \right], \{t , 0 , \text{inf}\}, \text{WorkingPrecision} \rightarrow 50, \ \text{AccuracyGoal} \rightarrow 5 \\
\text{ZpZ1} [s_n] &:= \text{NIntegrate} \left[ \text{Log} [u1] \star z[t/u1] \star z[u1] / u1 \star t^{-s} \right], \{u1 , 0 , \text{inf}\}, \{t , 0 , \text{inf}\}, \text{WorkingPrecision} \rightarrow 50, \ \text{AccuracyGoal} \rightarrow 5 \\
\text{ExpApprox} [x_] &:= \text{Sum} [\text{I} * \text{Factorial}[k] * x^k, \{k , 0 , 1\}] \\
\text{InvApprox} [x_] &:= \text{Sum} [\text{alpha} [[j]] * \text{Csgn} [x] * \text{ExpApprox}[-c [[j]]] * x * \text{Csgn} [x]], \{j , 1 , \text{nj}\} \\
\text{InvApprox2} [x_] &:= \text{Sum} [\text{alpha} [[j]] * \text{Csgn} [x] * \text{ExpApprox}[-c [[j]]] * x * \text{Csgn} [x]], \{j , 1 , \text{nj}\} \\
\text{integrand1} [[\text{Phi} , k_], ] &:= \text{Sum} [\text{alpha} [[j]] * \text{Csgn} [f [z0 + R \text{Exp} [1 * \text{Phi}]]] \star (k+1) * (-1) \star k/\text{Factorial}[k] * c [[j]]] \star k^{\text{KP} [z0 + R \text{Exp} [1 * \text{Phi}]]} \star k^{\text{K} [z0 + R \text{Exp} [1 * \text{Phi}]]} \star k^{\text{Z1} [z0 + R \text{Exp} [1 * \text{Phi}]]} \star \text{RExp} [1 * \text{Phi}]] / (2 * \text{Pi} * 1), \{j , 1 , \text{nj}\} \\
\text{integrand2} [[\text{Phi} , k_], ] &:= \text{Sum} [\text{alpha} [[j]] * \text{Csgn} [f [z0 + R \text{Exp} [1 * \text{Phi}]]] \star (k+1) * (-1) \star k/\text{Factorial}[k] * c [[j]]] \star k^{\text{KP} [z0 + R \text{Exp} [1 * \text{Phi}]]} \star k^{\text{K} [z0 + R \text{Exp} [1 * \text{Phi}]]} \star k^{\text{Z2} [z0 + R \text{Exp} [1 * \text{Phi}]]} \star \text{RExp} [1 * \text{Phi}]] / (2 * \text{Pi} * 1), \{j , 1 , \text{nj}\} \\
\text{integrand3} [[\text{Phi} , k_], ] &:= \text{Sum} [\text{alpha} [[j]] * \text{Csgn} [f [z0 + R \text{Exp} [1 * \text{Phi}]]] \star (k+1) * (-1) \star k/\text{Factorial}[k] * c [[j]]] \star k^{\text{KP} [z0 + R \text{Exp} [1 * \text{Phi}]]} \star k^{\text{K} [z0 + R \text{Exp} [1 * \text{Phi}]]} \star k^{\text{Z2} [z0 + R \text{Exp} [1 * \text{Phi}]]} \star \text{RExp} [1 * \text{Phi}]] / (2 * \text{Pi} * 1), \{j , 1 , \text{nj}\} \\
\text{integrand4} [[\text{Phi} , k_], ] &:= \text{Sum} [\text{alpha} [[j]] * \text{Csgn} [f [z0 + R \text{Exp} [1 * \text{Phi}]]] \star (k+1) * (-1) \star k/\text{Factorial}[k] * c [[j]]] \star k^{\text{KP} [z0 + R \text{Exp} [1 * \text{Phi}]]} \star k^{\text{K} [z0 + R \text{Exp} [1 * \text{Phi}]]} \star k^{\text{Z2} [z0 + R \text{Exp} [1 * \text{Phi}]]} \star \text{RExp} [1 * \text{Phi}]] / (2 * \text{Pi} * 1), \{j , 1 , \text{nj}\} \\
\text{Table} [\text{N} * \text{RExp} [1 * \text{Phi}]] &:= \text{Set} [z0 + R \text{Exp} [1 * \text{Phi}]] / \text{Set} [z0 + R \text{Exp} [1 * \text{Phi}]] / (2 * \text{Pi} * 1), \{\text{Phi} , 0 , 2 * \text{Pi} , 2 * \text{Pi} / 8\} \\
\text{Table} [\text{N} * \text{RExp} [1 * \text{Phi}]] &:= \text{Set} [z0 + R \text{Exp} [1 * \text{Phi}]] / \text{Set} [z0 + R \text{Exp} [1 * \text{Phi}]] / \text{InvApprox} [\text{Set} [z0 + R \text{Exp} [1 * \text{Phi}]] / (2 * \text{Pi} * 1), \{\text{Phi} , 0 , 2 * \text{Pi} , 2 * \text{Pi} / 8\} \\
\text{Table} [\text{N} * \text{RExp} [1 * \text{Phi}]] &:= \text{Set} [z0 + R \text{Exp} [1 * \text{Phi}]] / \text{Set} [z0 + R \text{Exp} [1 * \text{Phi}]] / \text{InvApprox2} [\text{Set} [z0 + R \text{Exp} [1 * \text{Phi}]] / (2 * \text{Pi} * 1), \{\text{Phi} , 0 , 2 * \text{Pi} , 2 * \text{Pi} / 8\} \\
\text{Table} [\text{integrand1} [[\text{Phi} , 0] + \text{integrand2} [[\text{Phi} , 1] + \text{integrand3} [[\text{Phi} , 0] + \text{integrand4} [[\text{Phi} , 1], \{\text{Phi} , 0 , 2 * \text{Pi} , 2 * \text{Pi} / 8\} \\

\text{Remark:} \ 2 \text{Code tested using Mathematica 12.2.0.0 for Mac OS X}