

Computer Programs in Physics



TRIQS/Nevanlinna: Implementation of the Nevanlinna Analytic Continuation method for noise-free data

Sergei Iskakov ^{a,*}, Alexander Hampel ^b, Nils Wentzell ^b, Emanuel Gull ^a^a Department of Physics, University of Michigan, Ann Arbor, MI 48109, USA^b Center for Computational Quantum Physics, Flatiron Institute, 162 5th Avenue, New York, NY 10010, USA

A B S T R A C T

We present the TRIQS/Nevanlinna analytic continuation package, an efficient implementation of the methods proposed by J. Fei et al. (2021) [53] and (2021) [55]. TRIQS/Nevanlinna strives to provide a high quality open source (distributed under the GNU General Public License version 3) alternative to the more widely adopted Maximum Entropy based analytic continuation programs. With the additional Hardy functions optimization procedure, it allows for an accurate resolution of wide band and sharp features in the spectral function. Those problems can be formulated in terms of imaginary time or Matsubara frequency response functions. The application is based on the TRIQS C++/Python framework, which allows for easy interoperability with other TRIQS-based applications, electronic band structure codes and visualization tools. Similar to other TRIQS packages, it comes with a convenient Python interface.

Program summary

Program Title: TRIQS/Nevanlinna

CPC Library link to program files: <https://doi.org/10.17632/4cbzfy5rds.1>Developer's repository link: <https://github.com/TRIQS/Nevanlinna>

Licensing provisions: GPLv3

Programming language: C++/Python

External routines/libraries: TRIQS 3.2 [1], Boost >= 1.76.0, Eigen >= 3.4.0, cmake >= 3.20.

Nature of problem: Finite-temperature field theories are widely used to study quantum many-body effects and electronic structure of correlated materials. Obtaining physically relevant spectral functions from results in the imaginary time/Matsubara frequency domains requires solution of an ill-posed analytic continuation problem as a post-processing step.

Solution method: We present an efficient C++/Python open-source implementation of the Nevanlinna/Caratheodory analytic continuation.

1. Introduction

The single-particle spectral function of interacting quantum many-body systems is an object of prime importance in many-body theory, since it reveals excitation information of a quantum system. This information can often be directly compared to experiment. Up to a prefactor, the spectral function corresponds to the imaginary part of the retarded real-frequency Green's function [2]. Finite-temperature field-theoretical methods, such as path-integral [3] and lattice quantum Monte Carlo [4,5], self-consistent finite-temperature perturbation theory [6–12], non-perturbative embedding theories [13–18], diagrammatic Monte Carlo methods [19,20] or continuous-time Monte Carlo impurity solvers [21–24] provide direct access to the so-called imaginary-time or Matsubara frequency Green's functions, which are closely related to the retarded real-frequency Green's function. The problem of numerical analytic continuation then consists of obtaining retarded real-frequency Green's functions from the Matsubara and imaginary time quantities obtained in the aforementioned computational methods. A naive solution of the analytic continuation equations

* The review of this paper was arranged by Prof. Blum Volker.

** This paper and its associated computer program are available via the Computer Physics Communications homepage on ScienceDirect (<http://www.sciencedirect.com/science/journal/00104655>).

* Corresponding author. Current address: Department of Physics, University of Michigan, Ann Arbor, MI 48109, USA.

E-mail addresses: siskakov@umich.edu (S. Iskakov), ahampel@flatironinstitute.org (A. Hampel), nwentzell@flatironinstitute.org (N. Wentzell), egull@umich.edu (E. Gull).

is ill conditioned, in the sense that a small change of the input Matsubara quantities results in a large change of the resulting spectral function [25].

In the presence of noise, numerous numerical approaches, including the Maximum Entropy method [25–29] and variants [30], the stochastic analytic continuation [31–36] and variants [37], as well as sparse modeling [38–40], the Prony method [41], machine learning [42,43], and projection, pole estimation, and semidefinite relaxation approaches [44], and Padé approximants [45,46] have been successful. Extensions and generalizations have also appeared to the problem of matrix-valued spectral functions [47], non-positive spectral functions [48] and superconductivity [49–52].

In the absence of noise, an interpolation of the input data with an appropriate function, rather than a fit, is desired. Apart from coinciding with input values, the interpolating function should also respect the analytic properties of Green's functions, in particular causality. The space of scalar causal response functions is spanned by so-called Nevanlinna functions [53,54], while their matrix-valued generalizations are known as Carathéodory functions [55,56] (in other contexts, such functions are also known as Herglotz, Pick, or Riesz functions). This function space is very restrictive, in the sense that physical response functions only exist for data that fulfills the so-called Pick criterion [57]. A closed-form continued fraction interpolation algorithm that parametrizes all possible causal response functions in terms of a free Nevanlinna function has been provided by Schur [58].

Nevanlinna analytic continuation by now has found applications in domains ranging from real-materials simulations [12,59] to high energy physics [60]. Generalizations to situations with noise [44], to bosonic response functions [61,60], and open-source implementations of the Nevanlinna method Julia have appeared [62]. Nevertheless, there is a need for a generic, well-documented and efficient implementation of the method for scalar and matrix-valued response functions, especially in the situation for continuous spectral functions where a function optimization is required to obtain smooth spectra. This paper provides such an implementation based on the TRIQS software library for interacting quantum systems [1], designed as a fully integrated application in the TRIQS ecosystem, but also usable standalone.

2. Theory

We start the method description by introducing the fermionic imaginary time Green's function

$$G_{\sigma,ij}(\tau) = -\langle T_\tau c_{\sigma,i}(\tau) c_{\sigma,j}^\dagger(0) \rangle, \quad (1)$$

with orbital index $i(j)$, spin index σ and imaginary time τ . The corresponding Matsubara Green's function can be obtained via Fourier transform

$$G_{\sigma,ij}(\omega_n) = \int_0^\beta G_{\sigma,ij}(\tau) e^{i\omega_n \tau}, \quad (2)$$

with fermionic Matsubara frequency $\omega_n = (2n+1)\frac{\pi}{\beta}$ for inverse temperature β .

The connection between Matsubara $(G_{\sigma,ij}(\omega_n))$ and retarded $(G_{\sigma,ij}^R(\omega))$ Green's functions is known as an analytical continuation. If the analytical expression for the Matsubara Green's function is known everywhere in the complex plane, e.g. through the Lehmann representation, the retarded Green's function can be obtained by the substitution $i\omega_n \rightarrow \omega + i\delta$ (where δ is a positive infinitesimal number). In practice, the function is only known on the Matsubara axis, and obtaining the real frequency information requires solution of an inhomogeneous Fredholm integral equation of the first kind with reference to the retarded Green's function and known Matsubara Green's function on the left hand side:

$$G_{\sigma,ij}(\omega_n) = -\frac{1}{\pi} \int_{-\infty}^{\infty} \frac{G_{\sigma,ij}^R(\omega)}{i\omega_n - \omega} d\omega. \quad (3)$$

The numerical solution of this equation is ill-posed [25] and unstable in the presence of numerical noise.

In this work we present a C++ software package that solves a Nevanlinna-Pick problem [58] using Nevanlinna and Carathéodory functions for the scalar- and matrix-valued analytical continuation problem as introduced in [53,55].

2.1. Schur algorithm

The main mathematical background for Nevanlinna and Carathéodory analytical continuation methods is based on the Schur algorithm. The Schur algorithm solves the Nevanlinna-Pick interpolation problem for contractive functions that are analytic and bounded by 1 on the unit disk [58]. For a given Schur function $S(z)$ we have a set of M pairs (z_i, F_i) of node points z_i and Schur parameters $F_i = S_i(z_i)$, that forms the following recurrence relation [63–66]

$$S_{i+1}(z) = \frac{|z_i|(z_i - z)}{z_i(1 - z_i^* z)} \left[\mathbb{1} - F_i F_i^\dagger \right]^{-\frac{1}{2}} \left[S_i(z) - F_i \right] \left[\mathbb{1} - F_i^\dagger S_i(z) \right]^{-1} \left[\mathbb{1} - F_i^\dagger F_i \right]^{\frac{1}{2}}, \quad (4)$$

here $X^{\frac{1}{2}}$, in case of a matrix-valued functions, standing for the Hermitian square root of X , and algebraic square root for scalar-valued functions.

An iterative algorithm first constructs all Schur functions $S_i(z_j)$ at all node points z_j , then solves inverse problem of Eq. (4) for $z \in \mathcal{D} \setminus \{z_i\}$: $S_M(z) \rightarrow S_{M-1}(z) \rightarrow \dots \rightarrow S_0(z) = S(z)$, where $S(z)$ is the desired Schur interpolant and $S_M(z)$ is an arbitrary Schur function [64].

2.2. Nevanlinna continuation

Nevanlinna functions [67,68] belong to a class of complex functions which are analytic in the open upper half-plane \mathcal{C}^+ and have non-negative imaginary part. Using the invertible Möbius transform Nevanlinna functions $N(z)$ can be mapped onto a class of contractive functions $\theta(z) = \frac{N(z)-i}{N(z)+i}$ that can be solved by Nevanlinna-Pick interpolation algorithm [58,69].

In case of the scalar-valued functions, inverse recurrent problem of Eq. (4) can be simplified and written as:

$$\theta_0(z) = \frac{a(z)\theta_M(z) + b(z)}{c(z)\theta_M(z) + d(z)}, \quad (5)$$

where $\theta_M(z)$ is an arbitrary contractive function, $a(z), b(z), c(z)$ and $d(z)$ defined as [69]

$$\begin{pmatrix} a(z) & b(z) \\ c(z) & d(z) \end{pmatrix} = \prod_{i=0}^{M-1} \begin{pmatrix} \frac{z-Y_j}{z-Y_j^*} & \theta_j(z_j) \\ \theta_j^*(z_j) \frac{z-Y_j}{z-Y_j^*} & 1 \end{pmatrix}, \quad (6)$$

here $z \in C^+$ and Y_j are values of node points in C^+ where the target Nevanlinna function has known values.

As it is shown in Ref. [53], a negative of the diagonal element of the fermionic Green's function, Eq. (2), is a Nevanlinna function. With that we can construct Nevanlinna-Pick interpolation problem for a Nevanlinna function $N(z) = -G(z)$, using positive Matsubara frequencies as node points. After the Nevanlinna-Pick interpolation is performed, the Green's function on the real frequency can be obtained through the inverse Möbius transform:

$$G(z = \omega + i\eta) = -i \frac{1 + \theta(z)}{1 - \theta(z)}. \quad (7)$$

2.3. Carathéodory continuation

A matrix-valued function $F(w)$ defined on open subset B belongs to the class of Carathéodory functions \mathbb{C} if, for any $w \in B$, the Hermitian matrix $\frac{1}{2}(F(z) + F^\dagger(z))$ is positive semi-definite. Caratheodory functions $C(z)$ defined on a unit disk $\mathcal{D} = z : |z| < 1$ can be mapped to corresponding Schur function using a Cayley transform:

$$S(z) = [\mathbb{1} - C(z)] [\mathbb{1} + C(z)]^{-1}, \quad (8)$$

and the Nevanlinna-Pick interpolation can be constructed [64].

It is easy to show that the product of a matrix-valued Green's function, Eq. (2), and imaginary unit i belongs to the class of Carathéodory functions [55]. To solve Nevanlinna-Pick interpolation problem for matrix-valued Green's function, we first need to transform Green's function $G(w)$ into a Carathéodory function $C(w)$ and map it onto $\bar{C}(w)$ defined on a unit disk $\mathcal{D} = z : |z| < 1$ using Möbius transform. Then, using Cayley transform, we map this Carathéodory function to a corresponding Schur function $S(z)$ and perform Schur algorithm (Eq. (4)). After the interpolation on the entire unit disk is obtained we subsequently apply inverse Cayley and Möbius transform to obtain Carathéodory function in the upper half of the complex plane

$$C(w) = \left[\mathbb{1} + \bar{C}\left(\frac{w-i}{w+i}\right) \right]^{-1} \left[\mathbb{1} - \bar{C}\left(\frac{w-i}{w+i}\right) \right]. \quad (9)$$

The real frequency matrix-valued Green's function is then obtained as:

$$G(w = \omega + i\eta) = -iC(w). \quad (10)$$

2.4. Existence criteria

Even though, mathematically both scalar- and matrix-valued Green's functions belong to Nevanlinna and Carathéodory function up to a corresponding prefactors (-1 and i), in actual calculations, due to round off errors, noise, and systematic approximation error the restrictions imposed on Green's functions can be violated. However, there are straightforwardly verifiable criteria for the existence of Nevanlinna and Carathéodory interpolants directly based on input data, which is a generalization of the Pick criterion. An interpolant exists if and only if the corresponding Pick's matrix is positive semi-definite [57,67]. The Pick's matrix for Nevanlinna function is

$$P_{ij}^N = \begin{bmatrix} 1 - \theta(Y_i)\theta^*(Y_j) \\ 1 - \frac{Y_i - i}{Y_i + i} \left(\frac{Y_j - i}{Y_j + i} \right)^* \end{bmatrix}_{ij}, \quad (11)$$

where $\theta(Y_i)$ is the corresponding value of contractive function at the Matsubara frequency Y_i . Similarly, the Pick matrix for Caratheodory function is [66,65]

$$P^C = \begin{bmatrix} 1 - \bar{C}^*(z_j)\bar{C}(z_i) \\ 1 - z_j^* z_i \end{bmatrix}_{mn \times mn}, \quad (12)$$

More detailed discussion of the underlying mathematics and the numerical procedure is presented in Refs. [53,55].

3. Usage

3.1. Installation

The current version has the following dependencies: TRIQS library version 3.2, Boost-Multiprecision library version 1.7.4 or higher, Eigen3 library version 3.4.0 or higher and optionally MPFR library for fast multiprecision evaluation. The installation instructions of TRIQS are provided on the website <https://triqs.github.io>.

Installing Nevanlinna is similar to that of other TRIQS-based applications. Assuming that TRIQS 3.2 has been installed at `/path/to/TRIQS/install` Nevanlinna is simply installed by issuing the following commands at the shell prompt:

```
$ git clone https://github.com/TRIQS/Nevanlinna
$ cd Nevanlinna
```

```
$ mkdir build && cd build
$ cmake -DTRIQS_PATH=/path/to/TRIQS/install/dir \
         -DCMAKE_INSTALL_PREFIX=/path/to/Nevanlinna/install/dir ../
$ make
$ make test
$ make install
```

This will compile the source code, run automatic tests, and install Nevanlinna in the `/path/to/Nevanlinna/install/dir` directory. Further installation instructions, such as installation without TRIQS support, are given in the online documentation [70].

3.2. Basic usage

Running Nevanlinna to analytically continue input data requires writing a simple Python script. This usage method is standard for TRIQS applications. Prior to executing the script the user should load the TRIQS as well as the TRIQS/Nevanlinna installation into their current environment:

```
source /path/to/TRIQS/install/dir/share/triqs/triqsvars.sh
source /path/to/Nevanlinna/install/dir/share/nevanlinna/nevanlinnavars.sh
```

Details of the script will vary depending on the physical observable to be continued, and its representation. Nonetheless, a typical script will have the following basic parts.

- Import TRIQS and Nevanlinna Python modules.

```
# TRIQS Green's functions used to store input and output data.
from triqs.gf import *
# Use TRIQS HDF5 library for filesystem storage
from h5 import HDFArchive
# Import Nevanlinna solver
from triqs_Nevanlinna import Solver
```

- Load the observable to be continued from an HDF5 archive.

```
# Open an HDF5 file and read Matsubara Green's function.
ar = HDFArchive('input.h5', 'r')
G_im = ar['input/gf']
# or create a Gf input Green's function
G_im = ...
```

Only the values stored in the `G_im.data` numpy array, the TRIQS Green's function container data accessor, will be used by Nevanlinna.

- Initialize the continuation solver

```
# Open an HDF5 file and read Matsubara Green's function.
solver = Solver(kernel=<Kernel Type>, precision=<precision>)
```

There are two possible options for kernel type: “NEVANLINNA” and “CARATHEODORY”. By default, the Nevanlinna kernel will be used. One can also provide the desired precision for multi-precision operations if the Nevanlinna code has been built with MPFR library support. By default, `precision = 100` decimal digits is used.

- Solve the factorization problem for the desired kernel:

```
solver.solve(G_im)
```

In current version, `G_im` should be matrix-valued TRIQS Green's function. If the Nevanlinna kernel is chosen the analytic continuation will only be performed for the diagonal matrix elements and a warning message will be shown. To continue a full matrix valued Green's function including off-diagonal elements the Carathéodory kernel should be chosen.

- Setup the real-frequency grid and evaluate continuation.

```
# Setup equidistant grid and choose broadening parameter
m = MeshReFreq(-5.0, 5.0, 101)
eta = 0.1
# Evaluate continuation for chosen grid and broadening eta
G_re = solver.evaluate(m, eta)
```

This function call is normally the most expensive part of the script. The result of the continuation will be stored in the `G_re` container as a TRIQS Green's function object.

3.2.1. Parallelization

Nevanlinna is implemented with support of OpenMP and MPI parallelization. To choose number of OpenMP threads the environment variable `NEVANLINNA_NUM_THREADS` is used. By default, the number of OpenMP threads are set to 1. When Nevanlinna is run within MPI environment it will be automatically detected and run in parallel. Fig. 1 shows parallel speedup and efficiency of MPI and OpenMP parallelization. In case of Carathéodory continuation, OpenMP scheme shows overall better parallel performance due to larger computational needs for each loop iteration.

3.3. Advanced usage

3.3.1. Finding smooth spectral functions

As described in Ref. [53], the continued fraction form of the Nevanlinna factorization provides an interpolation of the input data irrespective of the choice of θ_{M+1} , the last element in the factorization:

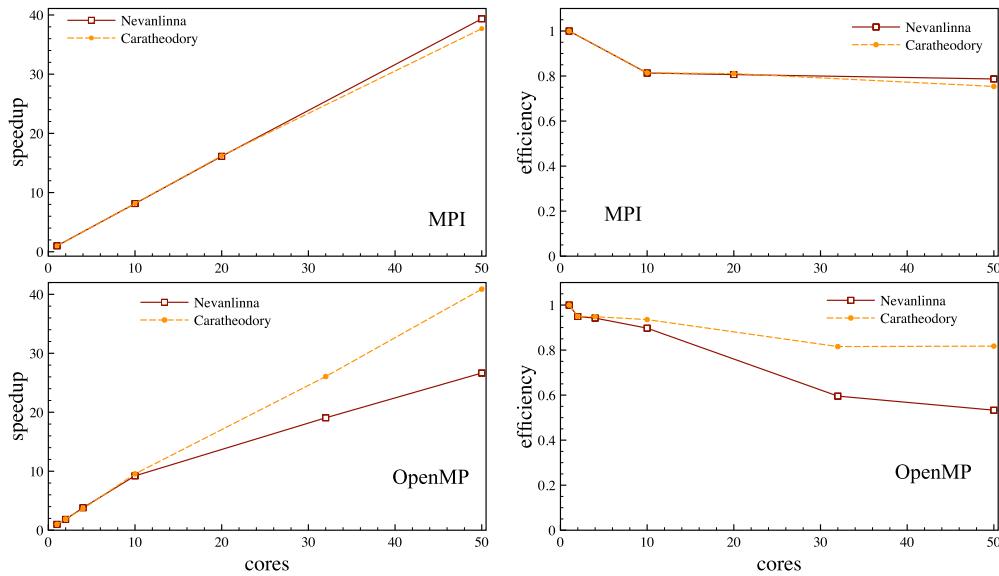


Fig. 1. Parallel speedup (left) and efficiency (right) using MPI (top panel) and OpenMP (bottom panel) parallelization for Nevanlinna and Carathéodory continuation of 10 orbital Matsubara Green's function.

$$G(z) = -h^{-1} [\theta(z)] \quad (13)$$

$$\theta(z) = \frac{a(z)\theta_{M+1}(z) + b(z)}{c(z)\theta_{M+1}(z) + d(z)}, \quad (14)$$

where $a(z)$, $b(z)$, $c(z)$ and $d(z)$ are Nevanlinna factorization coefficients, and h^{-1} is the inverse Möbius transform.

For systems with discrete spectra, both Nevanlinna and Carathéodory continuation work well irrespective of the choice of θ_{M+1} [53]. θ_{M+1} is therefore usually set to zero. However, for systems with continuous spectral functions the continuation will likely return a highly oscillating function if $\theta_{M+1} = 1$. This is due to the fact that many spectral functions interpolate the data provided as input; all of these spectral functions can be expressed by adjusting θ_{M+1} . The remaining freedom in the continuation problem can therefore be used to ‘smoothen’ the resulting spectral function. This involves an expansion of the free Nevanlinna function into appropriate basis functions (we choose basis functions of a Hardy space), followed by the minimization of a function norm [53].

This code implements a Hardy function optimization for Nevanlinna continuation. To perform the optimization, the user has to define a target function for the optimization problem that takes a retarded Green's function as an input. The optimization is performed using SciPy's conjugate gradient method [71,72]. In the example below we show an implementation for smoothness optimization by minimizing second derivative, as described in Ref. [53].

```
w = np.array([x for x in m.values()])
def SmoothnessOptimization (Gw):
    lmbda = 1e-4
    diff = 0.0
    # loop over all orbitals
    for i in range(Gw.data.shape[1]):
        # extract spectral function
        Aw = -Gw.data[:,i,i].imag/np.pi
        # compute estimate for second derivative
        Aw2 = (Aw[:-2] - 2*Aw[1:-1] + Aw[2:])/ (Gw.mesh.delta**2)
        # evaluate normalization criteria weight
        diff += np.abs(1 - scipy.integrate.simpson(Aw, w))**2
        # evaluate smoothness criteria weight
        diff += lmbda * scipy.integrate.simpson(Aw2**2, w[1:-1])
    return diff
# perform Hardy optimization
G_w_opt = solver.optimize(m, eta, SmoothnessOptimization, gtol=1e-3, maxiter=1000)
```

3.3.2. Continuation of raw data

In some cases a user may not have the option to provide TRIQS Green's function objects to perform analytical continuation. For that reason we provide direct access to continuation kernels classes and user simply can pass numpy arrays as an input. In the example below we show how to initialize kernel and use it on the raw data read from HDF5 file.

```
# import numpy and h5py modules to prepare raw data for continuation
import numpy as np
import h5py
# import continuation kernels
from triqs_Nevanlinna import kernels
# Initialize Nevanlinna kernel object
kernel = kernels.NevanlinnaKernel()
# Read raw data from HDF5 file
with h5py.File("data.h5", "r") as input_data:
    data = input_data["g/iw/data"]()
```

```

mesh = input_data["g/iw/mesh"] []
# Make Matsubara frequencies purely imaginary
mesh *= 1.j
# Build Nevanlinna factorization
kernel.init(mesh, data)
# Define real frequency grid for continuation
eta = 0.01
w = np.linspace(-4, 4, 1000)
w = w + 1.j*eta
# Compute the diagonal part of the retarded Green's function
Gw = kernel.evaluate(w)[:, :, :]

```

Here the data is a three-dimensional numpy array of size (niw, N, N) , where niw is the number of Matsubara frequencies and N is the number of orbitals. Currently, we have two kernels available `NevanlinnaKernel` for scalar valued continuation and `CaratheodoryKernel` for matrix-valued continuation.

3.3.3. C++ interface

The C++ part of Nevanlinna is designed as a library and uses CMake as a build system. Adding it as a dependency into another CMake project is straightforward. First, one needs to include and link Nevanlinna in the CMake project file

```

find_package(triqs_Nevanlinna)
target_link_libraries(TARGET_NAME triqs_Nevanlinna::triqs_Nevanlinna_c)

```

Here `TARGET_NAME` should be replaced according to the CMake target name in the project. Second, the user needs to provide the path to the installed Nevanlinna package by passing the following argument to CMake:

```
-Dtriqs_Nevanlinna_DIR=<Nevanlinna_install_dir>/lib/cmake/triqs_Nevanlinna
```

The example bellow shows a simple use case scenario reading imaginary time data from a HDF5 file into TRIQS nda arrays. The file with input data and a source code of the example can also be found in the `examples` directory of the github repository.

```

#include <triqs_Nevanlinna/kernels.hpp>
#include <mpi/mpi.hpp>
#include <nda/nda.hpp>
#include <nda/h5.hpp>
using namespace std::literals;
int main(int argc, char **argv) {
    // Initialize MPI environment
    mpi::environment env(argc, argv);
    // Create kernel object
    triqs_Nevanlinna::Nevanlinna_kernel kernel;
    // Define real-frequency grid
    size_t N_w = 5000;
    double w_min = -10., w_max = 10.;
    double eta = 0.1;
    auto del = (w_max - w_min) / (N_w - 1);
    auto grid = nda::basic_array{w_min + nda::arange(N_w) * del + eta * 1i};
    // Define imaginary time grid and data arrays
    auto mesh = nda::array<std::complex<double>, 1>{};
    auto G_iw = nda::array<std::complex<double>, 3>{};
    // Read imaginary time data from "input.h5" file
    auto input = h5::file(DATA_PATH + "/input.h5", 'r');
    h5::read(input, "data", G_iw);
    h5::read(input, "mesh", mesh);
    // Build Nevanlinna factorization
    kernel.init(mesh, G_iw);
    // Perform analytical continuation onto real frequency axis
    auto G_w = kernel.evaluate(grid);
}

```

3.4. Analyzes and troubleshooting

Both Nevanlinna and Carathéodory continuations interpolate, rather than fit, Matsubara data and are therefore susceptible to the quality of the input data. The existence of a causal continuation can be guaranteed only if the input data corresponds to a positive semi-definite matrix-valued function anywhere in the upper-half of the complex plane. In numerical double precision, this is almost never the case due to numerical round-off. Nevertheless, a simple check on the quality of the input data is given by the eigenvalues of the Pick matrix of the input data [57]. For precise causal data, negative eigenvalues of the Pick matrix will lie within the numerical diagonalization threshold. To obtain eigenvalues of the Pick matrix in this code, a user may read the `get_Pick_eigenvalues` property from the solver object. Fig. 2 shows typical behavior of Pick's matrix eigenvalues obtained for data without numerical noise (left panel) and for data with numerical noise (right panel). Since the largest negative eigenvalue $|v| \gg 0$, no causal interpolation of the input data exists, and therefore a successful analytical continuation cannot be guaranteed.

4. Examples

We demonstrate the application for scalar-valued functions using the Nevanlinna formalism.

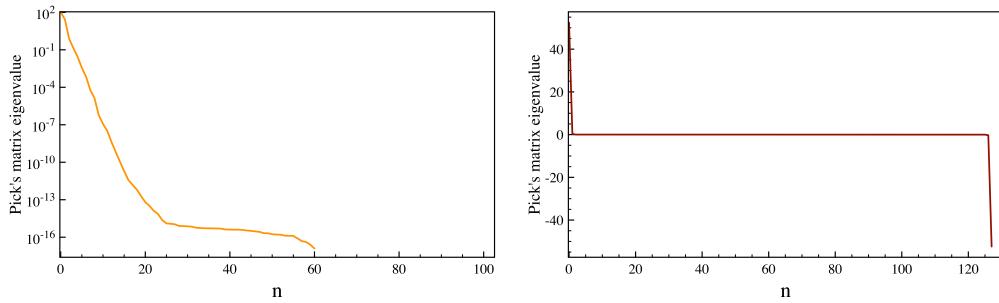


Fig. 2. Eigenvalues of the Pick matrix for data obtain from solution of a multi-orbital impurity problem with Exact Diagonalization (left) and Continous-Time QMC (right).

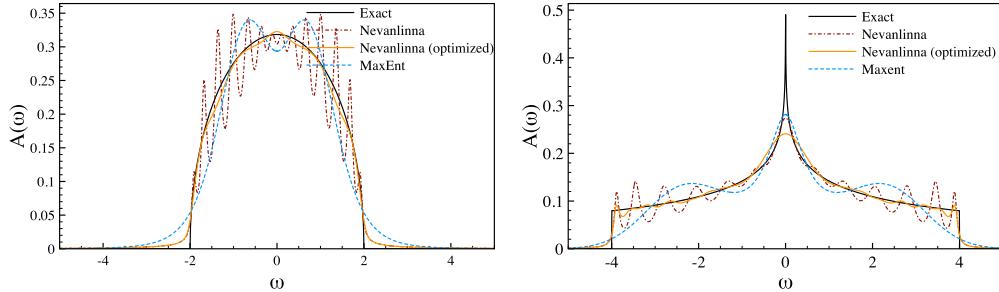


Fig. 3. Continuation results for analytically solvable models. Left panel: Semi-circular density of states for non-interacting Bethe lattice. Right panel: non-interacting square lattice density of states. Black lines: exact solution. Dashed red: Nevanlinna continuation. Orange: Nevanlinna continuation with Hardy function optimization. Dashed blue: Maxent analytical continuation.

4.1. Scalar-valued function continuation

We consider four exactly solvable examples: the non-interacting Bethe lattice, the non-interacting square lattice, the Hubbard atom, and the single impurity Anderson model. For the Hardy functions optimization we choose the smoothness criteria as described in the Ref. [53].

The first model we consider is non-interacting Bethe lattice, with

$$G(z) = \frac{z - \sqrt{4t^2 - z^2}}{2t^2}, \quad (15)$$

with general complex frequency z and hopping t . For Matsubara frequencies $z = i\omega_n$ and for real frequencies $z = \omega + i\delta$, where δ is Lorentzian broadening parameter. We choose $\delta = 0.1$ for all examples. The left panel in Fig. 3 shows the resulting unoptimized spectral function (dashed red line) and its comparison to the exact semicircular density of states (black line). As expected the unoptimized spectral function consists of a set of sharp peaks. After performing the Hardy functions optimization (see Sec. 3.3 for details) we are able to recover smoothness of the spectral function away from the boundaries.

Similarly to Bethe lattice, we show results for a non-interacting square lattice local density of states in the right panel of Fig. 3. Again, the Hardy function optimization allows to substantially suppress sharp features in the continuation. It is worth to mention that in both these examples second derivative does not exist at the spectral boundaries and possibly different type of optimization could be necessary.

Next we consider the single Hubbard atom with the Coulomb interaction $U = 2$ at half-filling. Its spectral function has discrete levels at $\omega = \pm \frac{U}{2}$. The Green's function for general complex frequency of the Hubbard atom is

$$G(z) = \frac{1}{2} \left[\frac{1}{z + \frac{U}{2}} + \frac{1}{z - \frac{U}{2}} \right] \quad (16)$$

The left panel of Fig. 4 shows Nevanlinna continuation (dashed orange) and exact (black) results. As expected Nevanlinna continuation perfectly resolved sharp peaks for that model.

The last example shows results for the discrete single impurity Anderson model that is described by the Hamiltonian

$$H = -\mu(n_{\uparrow,0} + n_{\downarrow,0}) - h(n_{\uparrow,0} - n_{\downarrow,0}) + Un_{\uparrow,0}n_{\downarrow,0} + \sum_{i=1}^N \sum_{\sigma} E_i n_{\sigma,i} + \sum_{i=1}^N \sum_{\sigma} V_i (c_{\sigma,i}^{\dagger} c_{\sigma,0} + c_{\sigma,0}^{\dagger} c_{\sigma,i}), \quad (17)$$

where μ is the chemical potential, U is the local interaction, E_i is the bath electrons energy and V_i is the coupling between impurity and bath electrons. This model does not have a simple closed-form solution and we use exact diagonalization to obtain both the Matsubara and real frequency Green's function using the Lehmann representation for a small number of bath sites $N = 2$. We choose $\mu = 1.0$, $U = 2.0$, $E = [-0.5, 0.7]$ and $V = [0.4, 0.5]$. The right panel of Fig. 4 shows the results from exact diagonalization (black) and Nevanlinna continuation (dashed orange). Similarly to the Hubbard atom, Nevanlinna continuation perfectly resolves all the poles in the real frequency Green's function.

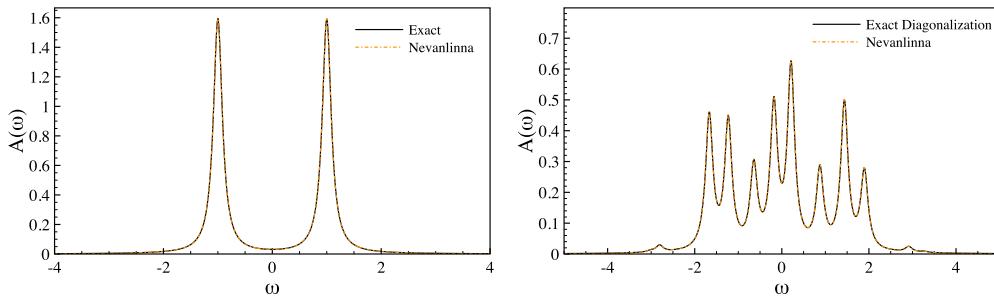


Fig. 4. Results for analytical continuation of exactly solvable models. Left panel: half-filled Hubbard atom at $U = 2.0$. Right panel: single orbital Anderson impurity model for $\mu = 1.0$, $U = 2.0$, with bath parameters $E = [-0.5, 0.7]$ and $V = [0.4, 0.5]$. Black lines: exact diagonalization, dashed orange lines: Nevanlinna analytical continuation.

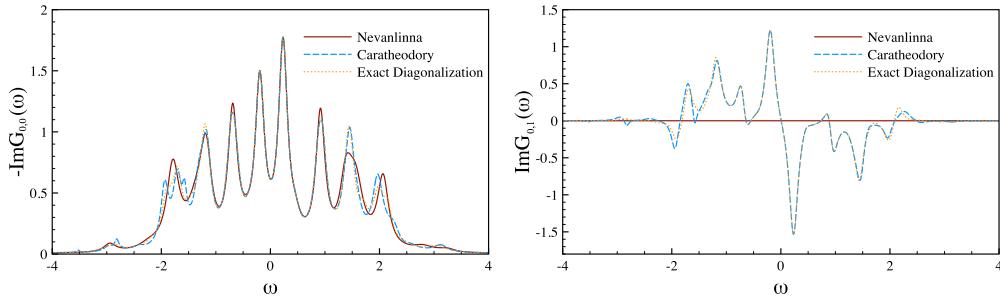


Fig. 5. Analytical continuation of the exact solution of two-orbital Anderson model, left panel: diagonal part of the Green's function, right panel: inter-orbital part of the Green's function. Dark red: Nevanlinna analytical continuation. Dashed blue: Carathéodory matrix-valued analytical continuation. Dotted: Exact diagonalization reference solution.

4.2. Matrix-valued function continuation

For the matrix valued continuation we consider the discrete two orbital Anderson impurity model with inter-orbital hopping that is described by the following Hamiltonian

$$\begin{aligned}
 H = & -\mu \sum_{i=0,1} (n_{\uparrow,i} + n_{\downarrow,i}) - h(n_{\uparrow,0} - n_{\downarrow,0}) + U n_{\uparrow,0} n_{\downarrow,0} \\
 & + \sum_{\sigma} t_{ij} (c_{\sigma,0}^{\dagger} c_{\sigma,1} + c_{\sigma,1}^{\dagger} c_{\sigma,0}) + V \frac{1}{2} \sum_{\sigma\sigma'} (n_{\sigma,0} n_{\sigma',1}) \\
 & + \sum_{k=2}^N \sum_{\sigma} E_k n_{\sigma,k} + \sum_{i=0,1} \sum_{k=2}^N \sum_{\sigma} V_k (c_{\sigma,k}^{\dagger} c_{\sigma,i} + c_{\sigma,i}^{\dagger} c_{\sigma,k}),
 \end{aligned} \tag{18}$$

with inter-orbital hopping t and inter-orbital interaction V , for two bath sites $N = 2$.

Fig. 5 shows the results for analytical continuation results of the two-orbital Anderson impurity model (Eq. (18)) for $U = 2.0$, $V = 0.1$, $\mu = 1.0$, $t = 0.3$, and bath parameters $E_k = [-0.5, 0.7]$ and $V_k = [0.4, 0.5]$, at inverse temperature $\beta = 40$ and broadening parameter $\eta = 0.1$. Dark red lines show Nevanlinna continuation and dashed blue lines correspond to Carathéodory continuation. As the Nevanlinna continuation kernel only allows to continue the diagonal part (left panel), the off-diagonal part (right panel) is only relevant when using the Carathéodory continuation.

5. Summary

We have presented the open-source TRIQS/Nevanlinna package that implements the analytic continuation method for noise free data providing two kernels for scalar-valued and matrix-valued function. We demonstrated the effectiveness of the TRIQS/Nevanlinna implementation for obtaining spectral functions for both model systems and realistic materials. We also provide Hardy function optimization for scalar-valued functions to optimize the continuation of continuous spectra. The current implementation allows user to pass a target function to perform desired optimization.

Future releases will include additional kernels for noisy data and continuation optimization for matrix-valued functions. The presented application is based upon the TRIQS software library and is part of its ecosystem, ensuring support for new compilers and libraries, and flexibility when used in other applications.

CRediT authorship contribution statement

Sergei Iskakov: Writing – original draft, Software, Data curation, Conceptualization. **Alexander Hampel:** Writing – review & editing, Validation, Software. **Nils Wentzell:** Writing – review & editing, Supervision, Project administration. **Emanuel Gull:** Writing – review & editing, Supervision, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data is provided with the shared code.

Acknowledgements

S.I. was supported by the Simons Foundation. E.G. was supported by the National Science Foundation under Grant No. 2001465. The Flatiron Institute is a division of the Simons Foundation.

References

- [1] O. Parcollet, M. Ferrero, T. Ayral, H. Hafermann, I. Krivenko, L. Messio, P. Seth, Triqs: a toolbox for research on interacting quantum systems, *Comput. Phys. Commun.* 196 (2015) 398–415, <https://doi.org/10.1016/j.cpc.2015.04.023>.
- [2] G. Mahan, Many-Particle Physics, Physics of Solids and Liquids, Springer, 2000, <https://books.google.com/books?id=xzSgZ4-yyMEC>.
- [3] D.M. Ceperley, Path integrals in the theory of condensed helium, *Rev. Mod. Phys.* 67 (1995) 279–355, <https://doi.org/10.1103/RevModPhys.67.279>.
- [4] R. Blankenbecler, D.J. Scalapino, R.L. Sugar, Monte Carlo calculations of coupled boson-fermion systems. I, *Phys. Rev. D* 24 (1981) 2278–2286, <https://doi.org/10.1103/PhysRevD.24.2278>.
- [5] D.J. Scalapino, R.L. Sugar, Monte Carlo calculations of coupled boson-fermion systems. II, *Phys. Rev. B* 24 (1981) 4295–4308, <https://doi.org/10.1103/PhysRevB.24.4295>.
- [6] L. Hedin, New method for calculating the one-particle Green's function with application to the electron-gas problem, *Phys. Rev.* 139 (1965) A796–A823, <https://doi.org/10.1103/PhysRev.139.A796>.
- [7] L.J. Holleboom, J.G. Snijders, A comparison between the Mo/ller-Plesset and Green's function perturbative approaches to the calculation of the correlation energy in the many-electron problem, *J. Chem. Phys.* 93 (8) (1990) 5826–5837, <https://doi.org/10.1063/1.459578>, https://pubs.aip.org/aip/jcp/article-pdf/93/8/5826/11207185/5826_1_online.pdf.
- [8] N.E. Dahlen, R. van Leeuwen, U. von Barth, Variational energy functionals of the Green function and of the density tested on molecules, *Phys. Rev. A* 73 (2006) 012511, <https://doi.org/10.1103/PhysRevA.73.012511>.
- [9] A. Stan, N.E. Dahlen, R. van Leeuwen, Levels of self-consistency in the GW approximation, *J. Chem. Phys.* 130 (11) (2009) 114105, <https://doi.org/10.1063/1.3089567>, arXiv:https://pubs.aip.org/aip/jcp/article-pdf/doi/10.1063/1.3089567/114105_1_online.pdf.
- [10] J.J. Phillips, D. Zgid, Communication: the description of strong correlation within self-consistent Green's function second-order perturbation theory, *J. Chem. Phys.* 140 (24) (2014) 241101, <https://doi.org/10.1063/1.4884951>, https://pubs.aip.org/aip/jcp/article-pdf/doi/10.1063/1.4884951/14082240/241101_1_online.pdf.
- [11] A.A. Rusakov, D. Zgid, Self-consistent second-order Green's function perturbation theory for periodic systems, *J. Chem. Phys.* 144 (5) (2016) 054106, <https://doi.org/10.1063/1.4940900>.
- [12] C.-N. Yeh, S. Iskakov, D. Zgid, E. Gull, Fully self-consistent finite-temperature *gw* in Gaussian Bloch orbitals for solids, *Phys. Rev. B* 106 (2022) 235104, <https://doi.org/10.1103/PhysRevB.106.235104>.
- [13] W. Metzner, D. Vollhardt, Correlated lattice fermions in $d = \infty$ dimensions, *Phys. Rev. Lett.* 62 (1989) 324–327, <https://doi.org/10.1103/PhysRevLett.62.324>.
- [14] A. Georges, G. Kotliar, Hubbard model in infinite dimensions, *Phys. Rev. B* 45 (1992) 6479–6483, <https://doi.org/10.1103/PhysRevB.45.6479>.
- [15] A. Georges, G. Kotliar, W. Krauth, M.J. Rozenberg, Dynamical mean-field theory of strongly correlated fermion systems and the limit of infinite dimensions, *Rev. Mod. Phys.* 68 (1996) 13–125, <https://doi.org/10.1103/RevModPhys.68.13>.
- [16] G. Kotliar, S.Y. Savrasov, K. Haule, V.S. Oudovenko, O. Parcollet, C.A. Marianetti, Electronic structure calculations with dynamical mean-field theory, *Rev. Mod. Phys.* 78 (2006) 865–951, <https://doi.org/10.1103/RevModPhys.78.865>.
- [17] D. Zgid, E. Gull, Finite temperature quantum embedding theories for correlated systems, *New J. Phys.* 19 (2) (2017) 023047, <https://doi.org/10.1088/1367-2630/aa5d34>.
- [18] A.A. Rusakov, S. Iskakov, L.N. Tran, D. Zgid, Self-energy embedding theory (seet) for periodic systems, *J. Chem. Theory Comput.* 15 (1) (2019) 229–240, <https://doi.org/10.1021/acs.jctc.8b00927>, pMID: 30540474.
- [19] N.V. Prokof'ev, B.V. Svistunov, Polaron problem by diagrammatic quantum Monte Carlo, *Phys. Rev. Lett.* 81 (1998) 2514–2517, <https://doi.org/10.1103/PhysRevLett.81.2514>.
- [20] N. Prokof'ev, B. Svistunov, Bold diagrammatic Monte Carlo technique: when the sign problem is welcome, *Phys. Rev. Lett.* 99 (2007) 250201, <https://doi.org/10.1103/PhysRevLett.99.250201>.
- [21] A.N. Rubtsov, V.V. Savkin, A.I. Lichtenstein, Continuous-time quantum Monte Carlo method for fermions, *Phys. Rev. B* 72 (2005) 035122, <https://doi.org/10.1103/PhysRevB.72.035122>.
- [22] P. Werner, A. Comanac, L. de' Medici, M. Troyer, A.J. Millis, Continuous-time solver for quantum impurity models, *Phys. Rev. Lett.* 97 (2006) 076405, <https://doi.org/10.1103/PhysRevLett.97.076405>.
- [23] E. Gull, P. Werner, O. Parcollet, M. Troyer, Continuous-time auxiliary-field Monte Carlo for quantum impurity models, *Europhys. Lett.* 82 (5) (2008) 57003, <https://doi.org/10.1209/0295-5075/82/57003>.
- [24] E. Gull, A.J. Millis, A.I. Lichtenstein, A.N. Rubtsov, M. Troyer, P. Werner, Continuous-time Monte Carlo methods for quantum impurity models, *Rev. Mod. Phys.* 83 (2011) 349–404, <https://doi.org/10.1103/RevModPhys.83.349>.
- [25] M. Jarrell, J. Gubernatis, Bayesian inference and the analytic continuation of imaginary-time quantum Monte Carlo data, *Phys. Rep.* 269 (3) (1996) 133–195, [https://doi.org/10.1016/0370-1573\(95\)00074-7](https://doi.org/10.1016/0370-1573(95)00074-7).
- [26] D. Bergeron, A.-M.S. Tremblay, Algorithms for optimized maximum entropy and diagnostic tools for analytic continuation, *Phys. Rev. E* 94 (2016) 023303, <https://doi.org/10.1103/PhysRevE.94.023303>.
- [27] R. Levy, J. LeBlanc, E. Gull, Implementation of the maximum entropy method for analytic continuation, *Comput. Phys. Commun.* 215 (2017) 149–155, <https://doi.org/10.1016/j.cpc.2017.01.018>.
- [28] L. Huang, Aflow: an open source toolkit for analytical continuation of quantum Monte Carlo data, *SSRN Electron. J.* (2022).
- [29] J. Kaufmann, K. Held, Anacon python package for analytic continuation, *Comput. Phys. Commun.* 282 (2023) 108519, <https://doi.org/10.1016/j.cpc.2022.108519>.
- [30] Y. Burnier, A. Rothkopf, Bayesian approach to spectral function reconstruction for Euclidean quantum field theories, *Phys. Rev. Lett.* 111 (2013) 182003, <https://doi.org/10.1103/PhysRevLett.111.182003>.
- [31] A.W. Sandvik, Stochastic method for analytic continuation of quantum Monte Carlo data, *Phys. Rev. B* 57 (1998) 10287–10290, <https://doi.org/10.1103/PhysRevB.57.10287>.
- [32] A.S. Mishchenko, N.V. Prokof'ev, A. Sakamoto, B.V. Svistunov, Diagrammatic quantum Monte Carlo study of the Fröhlich polaron, *Phys. Rev. B* 62 (2000) 6317–6336, <https://doi.org/10.1103/PhysRevB.62.6317>.
- [33] S. Fuchs, T. Pruschke, M. Jarrell, Analytic continuation of quantum Monte Carlo data by stochastic analytical inference, *Phys. Rev. E* 81 (2010) 056701, <https://doi.org/10.1103/PhysRevE.81.056701>.
- [34] A.W. Sandvik, Constrained sampling method for analytic continuation, *Phys. Rev. E* 94 (2016) 063308, <https://doi.org/10.1103/PhysRevE.94.063308>.
- [35] I. Krivenko, M. Harland, Triqs/som: implementation of the stochastic optimization method for analytic continuation, *Comput. Phys. Commun.* 239 (2019) 166–183, <https://doi.org/10.1016/j.cpc.2019.01.021>.
- [36] H. Shao, A.W. Sandvik, Progress on stochastic analytic continuation of quantum Monte Carlo data, *Phys. Rep.* 1003 (2023) 1–88, <https://doi.org/10.1016/j.physrep.2022.11.002>, Progress on stochastic analytic continuation of quantum Monte Carlo data.

[37] O. Goulko, A.S. Mishchenko, L. Pollet, N. Prokof'ev, B. Svistunov, Numerical analytic continuation: answers to well-posed questions, *Phys. Rev. B* 95 (2017) 014102, <https://doi.org/10.1103/PhysRevB.95.014102>.

[38] J. Otsuki, M. Ohzeki, H. Shinaoka, K. Yoshimi, Sparse modeling approach to analytical continuation of imaginary-time quantum Monte Carlo data, *Phys. Rev. E* 95 (2017) 061302, <https://doi.org/10.1103/PhysRevE.95.061302>.

[39] K. Yoshimi, J. Otsuki, Y. Motoyama, M. Ohzeki, H. Shinaoka, Spm: sparse modeling tool for analytic continuation of imaginary-time Green's function, *Comput. Phys. Commun.* 244 (2019) 319–323, <https://doi.org/10.1016/j.cpc.2019.07.001>.

[40] J. Otsuki, M. Ohzeki, H. Shinaoka, K. Yoshimi, Sparse modeling in quantum many-body problems, *J. Phys. Soc. Jpn.* 89 (1) (2020) 012001, <https://doi.org/10.7566/JPSJ.89.012001>.

[41] L. Ying, Analytic continuation from limited noisy matsubara data, *J. Comput. Phys.* 469 (2022) 111549, <https://doi.org/10.1016/j.jcp.2022.111549>.

[42] H. Yoon, J.-H. Sim, M.J. Han, Analytic continuation via domain knowledge free machine learning, *Phys. Rev. B* 98 (2018) 245101, <https://doi.org/10.1103/PhysRevB.98.245101>.

[43] R. Fournier, L. Wang, O.V. Yazyev, Q. Wu, Artificial neural network approach to the analytic continuation problem, *Phys. Rev. Lett.* 124 (2020) 056401, <https://doi.org/10.1103/PhysRevLett.124.056401>.

[44] Z. Huang, E. Gull, L. Lin, Robust analytic continuation of Green's functions via projection, pole estimation, and semidefinite relaxation, *Phys. Rev. B* 107 (2023) 075151, <https://doi.org/10.1103/PhysRevB.107.075151>.

[45] H.J. Vidberg, J.W. Serene, Solving the eliashberg equations by means of N-point Padé approximants, *J. Low Temp. Phys.* 29 (3) (1977) 179–192, <https://doi.org/10.1007/BF00655090>.

[46] K.S.D. Beach, R.J. Gooding, F. Marsiglio, Reliable Padé analytical continuation method based on a high-accuracy symbolic computation algorithm, *Phys. Rev. B* 61 (2000) 5147–5157, <https://doi.org/10.1103/PhysRevB.61.5147>.

[47] G.J. Kraberger, R. Triebel, M. Zingl, M. Aichhorn, Maximum entropy formalism for the analytic continuation of matrix-valued Green's functions, *Phys. Rev. B* 96 (2017) 155128, <https://doi.org/10.1103/PhysRevB.96.155128>.

[48] A. Rothkopf, Bayesian inference of nonpositive spectral functions in quantum field theory, *Phys. Rev. D* 95 (2017) 056016, <https://doi.org/10.1103/PhysRevD.95.056016>.

[49] E. Gull, A.J. Millis, Pairing glue in the two-dimensional Hubbard model, *Phys. Rev. B* 90 (2014) 041110, <https://doi.org/10.1103/PhysRevB.90.041110>.

[50] A. Reymbaut, D. Bergeron, A.-M.S. Tremblay, Maximum entropy analytic continuation for spectral functions with nonpositive spectral weight, *Phys. Rev. B* 92 (2015) 060509, <https://doi.org/10.1103/PhysRevB.92.060509>.

[51] X. Dong, E. Gull, A.J. Millis, Quantifying the role of antiferromagnetic fluctuations in the superconductivity of the doped Hubbard model, *Nat. Phys.* 18 (11) (2022) 1293–1296, <https://doi.org/10.1038/s41567-022-01710-z>.

[52] C. Yue, P. Werner, Maximum entropy analytic continuation of anomalous self-energies, [arXiv:2303.16888](https://arxiv.org/abs/2303.16888), 2023.

[53] J. Fei, C.-N. Yeh, E. Gull, Nevanlinna analytical continuation, *Phys. Rev. Lett.* 126 (2021) 056402, <https://doi.org/10.1103/PhysRevLett.126.056402>.

[54] R. Nevanlinna, Zur Theorie der Meromorphen Funktionen, *Acta Math.* 46 (1–2) (1925) 1–99, <https://doi.org/10.1007/BF02543858>.

[55] J. Fei, C.-N. Yeh, D. Zgid, E. Gull, Analytical continuation of matrix-valued functions: Carathéodory formalism, *Phys. Rev. B* 104 (2021) 165111, <https://doi.org/10.1103/PhysRevB.104.165111>.

[56] C. Carathéodory, Über den Variabilitätsbereich der Koeffizienten von Potenzreihen, die Gegebene Werte Nicht Annehmen, *Math. Ann.* 64 (1) (1907) 95–115, <https://doi.org/10.1007/BF01449883>.

[57] G. Pick, Über die Beschränkungen Analytischer Funktionen, Welche Durch Vorgegebene Funktionswerte Bewirkt Werdenber die Beschränkungen Analytischer Funktionen, Welche Durch Vorgegebene Funktionswerte Bewirkt Werden, *Math. Ann.* 78 (1917) 270–275, <https://doi.org/10.1007/BF01457103>.

[58] J. Schur, Über Potenzreihen, die im Innern des Einheits-Kreises Beschränkt Sind, *J. Reine Angew. Math.* 1918 (1918) 122–145, <https://doi.org/10.1515/crll.1918.148.122>.

[59] C.-N. Yeh, A. Shee, Q. Sun, E. Gull, D. Zgid, Relativistic self-consistent *gw*: exact two-component formalism with one-electron approximation for solids, *Phys. Rev. B* 106 (2022) 085121, <https://doi.org/10.1103/PhysRevB.106.085121>.

[60] T. Bergamaschi, W.I. Jay, P.R. Oare, Hadronic structure, conformal maps, and analytic continuation, [arXiv:2305.16190](https://arxiv.org/abs/2305.16190), 2023.

[61] K. Nogaki, H. Shinaoka, Bosonic nevanlinna analytic continuation, *J. Phys. Soc. Jpn.* 92 (3) (2023) 035001, <https://doi.org/10.7566/JPSJ.92.035001>.

[62] K. Nogaki, J. Fei, E. Gull, H. Shinaoka, Nevanlinna.jl: a Julia implementation of Nevanlinna analytic continuation, [arXiv:2302.10476](https://arxiv.org/abs/2302.10476), 2023.

[63] P. Delsarte, Y. Genin, Y. Kamp, The Nevanlinna-pick problem for matrix-valued functions, *SIAM J. Appl. Math.* 36 (1) (1979) 47–61, <http://www.jstor.org/stable/2100767>.

[64] P. Delsarte, Y. Genin, Y. Kamp, Generalized Schur representation of matrix-valued functions, *SIAM J. Algebraic Discrete Methods* 2 (2) (1981) 94–107, <https://doi.org/10.1137/0602013>.

[65] G. Chen, Çetin Kaya Koç, Computing matrix-valued Nevanlinna-pick interpolation, *Linear Algebra Appl.* 203–204 (1994) 253–263, [https://doi.org/10.1016/0024-3795\(94\)90205-4](https://doi.org/10.1016/0024-3795(94)90205-4).

[66] C. Yazici, H.K. Sevindir, A correction for computing matrix-valued Nevanlinna-Pick interpolation problem, *AIP Conf. Proc.* 1558 (1) (2013) 2474–2477, <https://doi.org/10.1063/1.4826042>.

[67] R.H. Nevanlinna, R. Nevanlinna, Über Beschränkte Analytische Funktionen, *Suomalaisen tiedeakatemian kustantama*, 1929.

[68] N. Alkheizer, The Classical Moment Problem and Some Related Questions in Analysis, *Classics in Applied Mathematics*, Society for Industrial and Applied Mathematics, 2020, <https://books.google.com/books?id=DhDiEAAQBAJ>.

[69] V.M. Adamyan, J. Alcober, I.M. Tkachenko, Reconstruction of distributions by their moments and local constraints, *Appl. Math. Res. Express* 2003 (2) (2003) 33–70, <https://doi.org/10.1155/S1687120003212028>.

[70] S. Iskakov, A. Hampel, N. Wentzell, E. Gull, <https://trqs.github.io/Nevanlinna/latest/>, 2023, [link]. <https://trqs.github.io/Nevanlinna/latest/>.

[71] E. Jones, T. Oliphant, P. Peterson, et al., SciPy: open source scientific tools for python, <http://www.scipy.org/>, 2001.

[72] *Conjugate Gradient Methods*, Springer, New York, New York, NY, 2006, pp. 101–134.