Efficient implementation of the continuous-time interaction-expansion quantum Monte Carlo method

Hiroshi Shinaoka, Yusuke Nomura, Emanuel Gull

Abstract

We describe an open-source implementation of the continuous-time interaction-expansion quantum Monte Carlo method for cluster-type impurity models with onsite Coulomb interactions and complex Weiss functions. The code is based on the ALPS libraries.

Keywords: Quantum impurity problems, continuous-time impurity solver, interaction expansion, complex Green’s functions, dynamical mean-field theory

1. Introduction

Quantum impurity problems describe small interacting sets of orbitals coupled to wide non-interacting leads. Originally developed in the context of magnetic impurity atoms embedded in a non-magnetic host [1], they have since found applications to quantum dots and molecular conductors [2], atoms adsorbed to surfaces [3], and appear as auxiliary objects in quantum embedding theories such as the dynamical mean field theory [4], its extensions [5-9] and the self-energy embedding theory [10, 11].

Most of these applications require the calculation of impurity model energies, Green’s functions and self-energies in a non-perturbative regime. Analytic methods are ill suited to this task, and one needs to resort to numerical methods such as the numerical renormalization group [12], exact diagonalization [13], configuration interaction [14], density matrix renormalization group theory [15-17], or quantum Monte Carlo [18-23].

Many embedding methods, especially when formulated as cluster theories [7] for simplified low-energy effective models, generate impurity models that have general off-diagonal and potentially complex-valued hybridization functions but interactions of the density-density type. Models with off-diagonal complex hybridization functions are substantially more difficult to solve than those with diagonal hybridizations, and require the use of specialized impurity solvers.

In this paper, we describe an open source implementation of such a solver. The method is an implementation of the algorithm developed by Rubtsov et al. [20] and implements the stochastic sampling...
of a weak coupling perturbation series to all orders.
The algorithm also implements the submatrix update scheme of Refs. [24, 25, 29]. In the absence of
a sign problem, it scales cubically as a function of system size, inverse temperature, and interaction strength. In general, results at low temperature are hampered by an exponential scaling do to a fermionic sign problem [27].

The remainder of this paper is organized as follows. In section 2 we introduce the model and algorithm. In section 3 we show the usage of the code. In section 4 we illustrate the usage of the code at a few examples. Section 5 contains our conclusions.

2. Model and algorithm

The current version of the ALPS/CT-INT impurity solver supports single-orbital multi-site (N-site) impurity models with onsite Hubbard interactions defined by the action

\[ S_{\text{imp}} = S_0 + S_{\text{int}}, \]

where

\[ S_0 = - \int_0^\beta d\tau d\tau' \sum_{i,j=0}^{N_\alpha - 1} \sum_{\sigma = 0}^{N_\sigma - 1} \left[ g_{\sigma\beta}(\tau - \tau') \right]_{ij} \]

\[ \hat{c}_{i\sigma}(\tau)\hat{c}_{j\sigma}(\tau'), \]

where \( \sigma, i, j \) are indices and the double integration goes over \( \tau \) and \( \tau' \). Here, \( \hat{c}_{i\sigma} \) (\( \hat{c}_{i\sigma}^\dagger \)) is a Grassmann variable representing the creation (annihilation) of an impurity electron specified by indices \( i \) and \( \sigma \). The solver assumes the Weiss function \( g_{\sigma\beta}(\tau - \tau') \) to be diagonal in \( \sigma \) but it can be off-diagonal in \( i \) and \( j \). The Weiss function is a (complex-valued) matrix with respect to the index \( i \) and \( j \) for each \( \sigma \).

In a typical single-orbital multi-site impurity model with two spins \( |\uparrow\rangle \) and \( |\downarrow\rangle \) but no hopping between different spins, \( \sigma \) enumerates spins \( (N_\sigma = 2) \) and \( i \) and \( j \) the \( N \) impurity sites \( (N = N) \). The interaction part is defined as

\[ S_{\text{int}} = \int_0^\beta d\tau \sum_{s = \pm 1} \sum_{i = 0}^{N_\sigma - 1} \frac{U}{2} \left[ \hat{n}_{i0}(\tau) - \alpha_0(s) \right] \times [\hat{n}_{i1}(\tau) - \alpha_1(s)], \]

where \( \delta = 1/2 + 0^+ \) and \( \hat{n}_{i\sigma} = \hat{c}_{i\sigma}^\dagger \hat{c}_{i\sigma} \). The onsite Coulomb repulsion \( U \) is assumed to be site-independent. The parameters \( \alpha_0(s) \) and \( \alpha_1(s) \) are introduced [20, 28] to avoid a trivial sign problem.

On the other hand, in a single-orbital multi-site model with spin-orbit coupling, the presence of hopping terms that mix different spin flavors implies that \( i \) and \( j \) enumerate spin-sites \((N_\sigma = 2N) \) but \( N = 1 \). In such cases, \( i \) and \( j \) enumerate spin-sites (the spin index runs first). Accordingly, the interaction part is given by

\[ S_{\text{int}} = \int_0^\beta d\tau \sum_{s = \pm 1} \sum_{i = 0}^{N_\sigma - 1} \frac{U}{2} \left[ \hat{n}_{i21,0}(\tau) - \alpha_0(s) \right] \times [\hat{n}_{i21+1,0}(\tau) - \alpha_1(s)], \]

where \( \alpha_0(s) \) and \( \alpha_1(s) \) are the same as those defined in Eq. (3).

The ALPS/CT-INT solver implements the continuous-time interaction-expansion QMC method [20]. A series expansion of the partition function is sampled in terms of \( U \) using an efficient sampling method, the so-called submatrix update [24, 25, 29]. For details of the submatrix updates in CT-INT refer to Ref. [29].

3. Usage

3.1. Requirements and installation

The CT-INT code is built on an updated version of the core libraries of ALPS (Applications and Libraries for Physics Simulations libraries) [ALPSCore libraries] [30], the Boost libraries, and Eigen3. Eigen3 is a C++ template header-only library for linear algebra. They must be preinstalled. One needs a MPI C++ compiler with support for C++11 language features and CMake to build the solver. It will install two executables “ctint_real” and “ctint_complex”. Only the difference between these two is that ctint_real assumes the Weiss function to be real and runs faster in such cases. The formats of input and output files are the same.

The latest version of the code is available from a public Git repository at https://github.com/ALPSCore/CT-INT. One can also find a more detailed description of usage in Wiki documentation pages at https://github.com/ALPSCore/CT-INT/wiki.
3.2. Input data

The essential input data of the solver are

- The complex-valued Weiss function defined on a grid in the interval $[0, \beta]$.
- The onsite Coulomb interaction $U$.

We can also specify the number of thermalization steps, measurement steps, the interval of measurement of the Green’s function. All input parameters except for the Weiss function are read from a single input file. The Weiss function must be stored in a separated text file in a given format. The format of input files are described in the Wiki documentation pages.

3.3. Execution

Once you prepare two input files for runtime parameters and the Weiss function, you can run the solver as follows.

```bash
$ mpirun -np 120 ctint_real params.ini
```

In this example, we specify the values of runtime parameters in params.ini. The simulation results are written into a HDF5 file named params.out.h5. Some examples are given in the following section.

3.4. Output data

The Green’s function is defined as

$$ G_{\sigma,ij}(\tau) \equiv \langle T_\tau c_{\sigma i}^\dagger(\tau)c_{j\sigma}(0) \rangle, \quad (5) $$

where $T_\tau$ is a time-ordering operator. The Green’s function is measured as

$$ S_{\sigma,ij}(i\omega_n) \equiv -\sum_k \Sigma_{\sigma,ik}(i\omega_n)G_{\sigma,kj}(i\omega_n). \quad (6) $$

In practice, the expansion coefficients in the Legendre representation ($S_{\sigma,ij}$) are measured [31]. The Matsubara-frequency data $S(i\omega_n)$ are reconstructed as

$$ S_{\sigma,ij}(i\omega_n) = \sum_l T_{nl}(S_{\sigma,ij})_l, \quad (7) $$

where $l$ is the index of Legendre polynomials. The matrix elements $T_{nl}$ are introduced in Ref. [31].

Then, the Green’s function can be reconstructed via the Dyson equation. Equal-time quantities such as $\langle n_{\sigma i}n_{\sigma j} \rangle$ and $\langle n_{\sigma i} \rangle$ are also measured.

The results of the measurement are stored in a HDF5 [32] file. The format of the output file is described in detail in Wiki documentation pages.

4. Examples

4.1. Three-site impurity model

We consider a three-site model with onsite Coulomb repulsion. The local Hamiltonian is given by

$$ H = -\sum_{i \neq j} \sum_{\sigma} c_{\sigma i}^\dagger c_{j\sigma} - \mu \sum_{i=1}^{3} n_{i\sigma} + U \sum_{i=1}^{3} n_{i\uparrow}n_{i\downarrow}, \quad (8) $$

where $c_{\sigma i}^\dagger$ and $c_{\sigma i}$ are creation/annihilation operators of an electron at site $i$ with spin $\sigma$. The electron density operator is defined as $n_{i\sigma} \equiv c_{i\sigma}^\dagger c_{i\sigma}$. For the bath, we use a semieliptical density of states with bandwidth equal to $4t$, and set $t = 1$. This model is the same as that used for investigating a sign problem in a previous study [27].

We solve the model for $U = 4$ and $\beta = 10$ at $\mu = U/2$. The input file looks like this:

```bash
total_steps = 15000000
thermalization_steps = 15000
measurement_period = 10
model.beta = 10.0
model.spins = 2
model.U = 4.0
model.sites = 3
model.G0_tau_file = G0_TAU.txt
G1.n_matsubara = 1000
G1.n_legendre = 50
```

The data were obtained by running the solver with 120 MPI processes for 45 minutes. The Green’s function was measured every 10 Monte Carlo steps. The program writes the following messages to the standard output at the end of the simulation:

- average matrix size was: 42.0968 42.0968
- average sign was: 0.999426

#### Timing analysis ####

For measurement_period = 10 steps, each part took

- Monte Carlo update: 194.746 ms
- Recompute inverse matrix: 0.483439 ms
- Global update: 0.000690577 ms
- Measurement: 1.73871 ms

The average matrix sizes are around 42 for each spin. The computational time spent for the measurement is negligible compared to that for the Monte Carlo updates. This indicates that you could reduce the value of measurement_period to measure the Green’s function more often. The average sign is close to 1.

The computed results of $\Sigma_{\sigma,ij}(i\omega_n)$ are shown in Fig. 1. The self-energy was obtained by solving the Dyson equation. The data for up and down spins agree within error bars.
4.2. Self-consistent calculations of the 2D Hubbard model within the dynamical cluster approximation

The ALPS/CT-INT can be used together with the DCA for solving the 2D Hubbard model on the square lattice. The Hubbard model on the 2D square lattice is defined as

$$H = -t \sum_{\langle i,j \rangle} \sum_{\sigma} c_{i\sigma}^\dagger c_{j\sigma} - t' \sum_{\langle\langle i,j \rangle\rangle} \sum_{\sigma} c_{i\sigma}^\dagger c_{j\sigma} + U \sum_i n_{i\uparrow} n_{i\downarrow},$$  \hspace{1cm} (9)

where $\langle i,j \rangle$ and $\langle\langle i,j \rangle\rangle$ indicate pairs of nearest neighbor and next nearest neighbor sites, respectively. We take nearest neighbor hopping amplitude $t$ as energy unit, i.e., $t = 1$. $t'$ and $U$ are the next nearest neighbor hopping and Hubbard interaction, respectively.

Here we employ 8 site DCA, in which the Brillouin zone is partitioned into 8 patches (Fig. 2). Each patch is labeled by the central momentum $\mathbf{K}$. Due to symmetry, the number of inequivalent patches is reduced to 4: The inequivalent patches are $(0, 0)$, $(\pi/2, \pi/2)$, $(\pi, 0)$, and $(\pi, \pi)$. In each patch, the self-energy is approximated by that of central momentum i.e., $\Sigma(\mathbf{k}, i\omega_n) = \Sigma_{\mathbf{K}}(i\omega_n)$. We also assume a nonmagnetic solution, dropping the spin index hereafter. Within this approximation, the Green’s function for each patch is given by

$$G_{\mathbf{K}}(i\omega_n) = \int d\mathbf{k} \frac{1}{i\omega_n + \mu - E_{\mathbf{k}} - \Sigma_{\mathbf{K}}(i\omega_n)}$$  \hspace{1cm} (10)
where the integral is done at each momentum patch. $\epsilon_k$ is the energy dispersion of noninteracting system given by $\epsilon_k = -2t(\cos(k_x) + \cos(k_y)) - 4t'\cos(k_x)\cos(k_y)$. Further details of the self-consistency can be found in Ref. [7].

We perform self-consistent calculations combining the ALPS/CT-INT solver and an external program for solving self-consistent equations. Figure 3 illustrates the self-consistent cycle of DCA. An input file for ALPS/CT-INT, “ctint.ini” looks like this:

```
model.beta = 5.0
model.spins = 2
G1.n_matsubara = 1000
total_steps = 600000
G1.n_legendre = 60
model.U = 6.0
measurement_period = 10
thermalization_steps = 500
model.sites = 8
model.G0_tau_file = G0_TAU.txt
```

Figure 4 shows the 8 site DCA results of (a) Green’s function and (b) the imaginary part of self-energy for $t' = -0.15$, $U = 6$, $\mu = U/2 - 1.0$, and $\beta = 5$.

5. Summary

We have presented an open-source C++ implementation of the continuous-time interaction expansion Monte Carlo method for impurity models with certain types of density-density Coulomb interactions and general hybridization functions. More general forms of interaction will be supported in a future.
version. We have discussed the technical details of
the implementation. We presented some examples
of Monte Carlo simulation results for a three-site
model as well as results of 8-site DCA calculations
for the two-dimensional Hubbard model. They can
serve as a benchmark or reference.

Acknowledgments

We gratefully acknowledge support by the wider
ALPS community [34, 35]. HS was supported by
JSPS KAKENHI Grant No. 16H01064 (J-Physics),
18H04301 (J-Physics), 16K17735. YN was supported
by Grant-in-Aids for Scientific Research (JSPS
KAKENHI) Grant No. 17K14336 and was supported
by JSPS KAKENHI Grant No. 16H01064 (J-
Physics), 18H04301 (J-Physics), 16K17735. YN
was supported by JSPS KAKENHI Grant No. 17K14336 and
16H06345. HS and YS were supported by JSPS
KAKENHI Grant No. 18H01158. EG was sup-
ported by NSF DMR 1606348. Part of the calcula-
tions were performed on the ISSP supercomputing
system.

References

References

[1] P. W. Anderson, Localized magnetic states in
metals, Phys. Rev. 124 (1) (1961) 41–53. doi:10.1103/
PhysRev.124.41.

URL http://dx.doi.org/10.1103/PhysRev.124.41

[2] R. Hanson, L. P. Kouwenhoven, J. R. Petta, S. Tarucha,
L. M. K. Vandersypen, Spins in few-electron quantum
dots, Reviews of Modern Physics 79 (4) (2007) 1217.
doi:10.1103/RevModPhys.79.1217.

URL http://link.aps.org/abstract/RMP/v79/p1217

[3] R. Brako, D. M. Neum's, Slowly varying time-dependent
local perturbations in metals: a new approach, Journal
of Physics C: Solid State Physics 14 (21) (1981) 3065–
3078. doi:10.1088/0022-3719/14/21/023.

URL http://stacks.iop.org/0022-3719/14/3065

[4] A. Georges, G. Kotliar, W. Krauth, M. J. Rozenberg,
Dynamical mean-field theory of strongly correlated
fermion systems and the limit of infinite dimensions,
Reviews of Modern Physics 68 (1) (1996) 13–125.
doi:10.1103/RevModPhys.68.13.

URL http://adsabs.harvard.edu/cgi-bin/nph-data_query?bibcode=1996RvMP...68...13G&link_type=
EJOURNAL

[5] M. H. Hettler, M. Jarrell, H. R. Krishnamurthy, M.
Mukherjee, The Dynamical Cluster Approximation
(1999). doi:10.1103/PhysRevB.58.R7475.

URL http://link.aps.org/doi/10.1103/PhysRevB.58.R7475

[6] A. I. Lichtenstein, M. I. Katsnelson, Antiferro-
romagnetism and d-wave superconductivity in
cuprates: A cluster dynamical mean-field theory,
Physical Review B 62 (14) (2000) R9283–R9286.
doi:10.1103/PhysRevB.62.R9283.

URL http://link.aps.org/doi/10.1103/PhysRevB.62.R9283

[7] T. Maier, M. Jarrell, T. Pruschke, M. H. Hettler,
Quantum cluster theories, Reviews of Modern Physics
77 (3) (2005) 1027–1080. doi:10.1103/RevModPhys.77.1027.

URL http://link.aps.org/doi/10.1103/RevModPhys.77.1027

[8] K. Held, A. A. Katanin, A. A. Katanin, A. Toschi,
Dynamical Vertex Approximation An Introduction,
Progress of Theoretical ... arXiv:0807.1860v1.

URL http://ipvs.oxfordjournals.org/content/176/117.abstract

[9] A. N. Rubtsov, M. I. Katsnelson, A. I. Lichtenstein,
Dual fermion approach to nonlocal correlations in
the Hubbard model, Physical Review B 77 (3) (2008)
033101–4. doi:10.1103/PhysRevB.77.033101.

URL http://link.aps.org/doi/10.1103/PhysRevB.77.033101

[10] A. A. Katanen, E. Gull, D. Zgid, Systematically
improvable multiscale solver for correlated electron sys-
tems, Phys. Rev. B 91 (2015) 121111. doi:10.1103/
PhysRevB.91.121111.

URL https://link.aps.org/doi/10.1103/PhysRevB.91.121111

[11] D. Zgid, E. Gull, Finite temperature quantum embed-
ding theories for correlated systems, New Journal of
Physics 19 (2) (2017) 023047.

URL http://stacks.iop.org/1367-2639/19/i=2/a=023047

[12] R. Builla, T. A. Costi, T. Pruschke, Numerical
renormalization group method for quantum impurity
systems, Reviews of Modern Physics 80 (2) (2008) 395.
doi:DOI:10.1103/RevModPhys.80.395.

URL http://journals.aps.org/rmp/abstract/10.1103/
RevModPhys.80.395

[13] M. Caffarel, W. Krauth, Exact diagonalization
approach to correlated fermions in infinite di-
dimensions: Mott transition and superconductivity,
Physical Review Letters 72 (10) (1994) 1545–1548.
doi:10.1103/PhysRevLett.72.1545.

URL http://link.aps.org/doi/10.1103/
PhysRevLett.72.1545

[14] D. Zgid, E. Gull, G. K.-L. Chan, Truncated configura-
tion interaction expansions as solvers for correlated
quantum impurity models and dynamical mean-field
theory, Physical Review B 86 (16) (2012) 165128.
doi:10.1103/PhysRevB.86.165128.

URL http://journals.aps.org/prb/pdf/10.1103/
PhysRevB.86.165128

[15] D. J. García, K. Hallberg, M. J. Rozenberg, Dynamical
mean field theory with the density matrix renormal-
ization group, Phys. Rev. Lett. 93 (24) (2004) 246403.
doi:10.1103/PhysRevLett.93.246403.

URL http://dx.doi.org/10.1103/PhysRevLett.93.
246403

[16] S. Nishimoto, F. Gebhard, E. Jeckelmann, Dynamical
mean-field theory calculation with the dynamical
density-matrix renormalization group, Physica B:
Condensed Matter 378-380 (2006) 283 – 285, proceedings
of the International Conference on Strongly Correlated
Electron Systems - SCES 2005. doi:10.1016/j.physb.
2006.01.104.

URL http://dx.doi.org/10.1016/j.physb.2006.01.104

[17] F. A. Wolf, I. P. McCulloch, O. Parcollet,
U. Schollwöck, Chebyshev matrix product state impurity
solver for dynamical mean-field theory, Phys. Rev.
