The HANDE-QMC project: open-source stochastic quantum chemistry from the ground state up

James S. Spencer,[†,‡] Nick S. Blunt,[¶,§,||||] Seonghoon Choi,[¶,||||] Jiří Etrych,[¶,||||] Maria-Andreea Filip,[¶,||||] W. M. C. Foulkes,[†,||,||||] Ruth S.T. Franklin,[¶,||||] Will J. Handley,[⊥,#,@,||||] Fionn D. Malone,[†,△,||||] Verena A. Neufeld,[¶,||||] Roberto Di Remigio,[∇,††,||||] Thomas W. Rogers,[†,||||] Charles J.C. Scott,[¶,||||] James J. Shepherd,[‡‡,||||] William A. Vigor,[¶¶,||||] Joseph Weston,[†,||||] RuQing Xu,[§§,||||] and Alex J.W. Thom*[,¶,¶¶]

†*Dept. of Physics, Imperial College London, South Kensington Campus, London SW7 2AZ, United Kingdom*
‡*Dept. of Materials, Imperial College London, South Kensington Campus, London SW7 2AZ, United Kingdom*
¶*University Chemical Laboratory, Lensfield Road, Cambridge CB2 1EW, United Kingdom*
§*St John's College, St John's Street, Cambridge, CB2 1TP, United Kingdom*
||*Department of Physics, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA*
⊥*Astrophysics Group, Cavendish Laboratory, Cambridge, CB3 OHE, UK*
#*Kavli Institute for Cosmology, Madingley Road, Cambridge, CB3 0HA, UK*
@*Gonville & Caius College, Trinity Street, Cambridge, CB2 1TA, UK*
△*Quantum Simulations Group, Lawrence Livermore National Laboratory, Livermore, California 94550, USA*
∇*Hylleraas Centre for Quantum Molecular Sciences, Department of Chemistry, University of Tromsø - The Arctic University of Norway, N-9037 Tromsø, Norway*
††*Department of Chemistry, Virginia Tech, Blacksburg, Virginia 24061, United States*
‡‡*Chemistry Building, University of Iowa, Iowa, 52240, USA*
¶¶*Dept. of Chemistry, Imperial College London, South Kensington Campus, London SW7 2AZ, United Kingdom*
§§*Department of Modern Physics, University of Science and Technology, Hefei, Anhui, 230026, China*
||||*Authors listed alphabetically*

E-mail: ajwt3@cam.ac.uk

Abstract

Building on the success of Quantum Monte Carlo techniques such as diffusion Monte Carlo, alternative stochastic approaches to solve electronic structure problems have emerged over the last decade. The full configuration interaction quantum Monte Carlo (FCIQMC) method allows one to systematically approach the exact solution of such problems, for cases where very high accuracy is desired. The introduction of FCIQMC has subsequently led to the

development of coupled cluster Monte Carlo (CCMC) and density matrix quantum Monte Carlo (DMQMC), allowing stochastic sampling of the coupled cluster wave function and the exact thermal density matrix, respectively. In this article we describe the HANDE-QMC code, an open-source implementation of FCIQMC, CCMC and DMQMC, including initiator and semi-stochastic adaptations. We describe our code and demonstrate its use on three example systems; a molecule (nitric oxide), a model solid (the uniform electron gas), and a real solid (diamond). An illustrative tutorial is also included.

1 Introduction

Quantum Monte Carlo (QMC) methods, in their many forms, are among the most reliable and accurate tools available for the investigation of realistic quantum systems.[1] QMC methods have existed for decades, including notable approaches such as variational Monte Carlo (VMC),[2–6] diffusion Monte Carlo (DMC)[1,7–10] and auxiliary-field QMC (AFQMC);[11] such methods typically have low scaling with system size, efficient large-scale parallelization, and systematic improvability, often allowing benchmark quality results in challenging systems.

A separate hierarchy exists in quantum chemistry, consisting of methods such as coupled cluster (CC) theory,[12] Møller-Plesset perturbation theory (MPPT),[13] and configuration interaction (CI), with full CI (FCI)[14] providing the exact benchmark within a given single-particle basis set. The scaling with the number of basis functions can be steep for these methods: from N^4 for MP2 to exponential for FCI. Various approaches to tackle the steep scaling wall have been proposed in the literature: from adaptive selection algorithms[15–20] and many-body expansions for CI[21] to the exploitation of the locality of the one-electron basis set[22] for MP2 and CC.[23–25] Such approaches have been increasingly successful, now often allowing chemical accuracy to be achieved for systems comprising thousands of basis functions.

In 2009, Booth, Thom and Alavi introduced the full configuration interaction quantum Monte Carlo (FCIQMC) method.[26] The FCIQMC method allows essentially exact FCI results to be achieved for systems beyond the reach of traditional, exact FCI approaches; in this respect, the method occupies a similar space to the density matrix renormalization group (DMRG) algorithm[27–29] and selected CI approaches.[15–20] Employing a sparse and stochastic sampling of the FCI wave function greatly reduces the memory requirements compared to exact approaches. The introduction of FCIQMC has led to the development of several other related QMC methods, including coupled cluster Monte Carlo (CCMC),[30,31] density matrix quantum Monte Carlo (DMQMC),[32,33] model space quantum Monte Carlo (MSQMC),[34–36] clock quantum Monte Carlo,[37] driven-dissipative quantum Monte Carlo (DDQMC),[38] and several other variants, including multiple approaches for studying excited-state properties.[34,39–41]

In this article we present HANDE-QMC (Highly Accurate N-DEterminant Quantum Monte Carlo), an open-source quantum chemistry code that performs several of the above quantum Monte Carlo methods. In particular, we have developed a highly-optimized and massively-parallelized package to perform state-of-the-art FCIQMC, CCMC and DMQMC simulations.

An overview of stochastic quantum chemistry methods in HANDE-QMC is given in Section 2. Section 3 describes the HANDE-QMC package, including implementation details, our development experiences, and analysis tools. Applications of FCIQMC, CCMC and DMQMC methods are contained in Section 4. We conclude with a discussion in Section 5 with views on scientific software development and an outlook on future work. A tutorial on running HANDE is provided in the Supplementary Material.

2 Stochastic quantum chemistry

2.1 Full Configuration Interaction Quantum Monte Carlo

The FCI ansatz for the ground state wavefunction is $|\Psi_{\mathrm{CI}}\rangle = \sum_{\mathbf{i}} c_{\mathbf{i}} |D_{\mathbf{i}}\rangle$, where $\{D_{\mathbf{i}}\}$ is the set of Slater determinants. Noting that $(1 - \delta\tau\hat{H})^N |\Psi_0\rangle \propto |\Psi_{\mathrm{CI}}\rangle$ as $N \to \infty$, where Ψ_0 is some arbitrary initial vector with $\langle\Psi_0|\Psi_{\mathrm{CI}}\rangle \neq 0$ and $\delta\tau$ is sufficiently small,[42] the coefficients $\{c_{\mathbf{i}}\}$ can be found via an iterative process derived from a first-order solution to the imaginary-time Schrödinger equation:[26]

$$c_{\mathbf{i}}(\tau + \delta\tau) = c_{\mathbf{i}}(\tau) - \delta\tau \sum_{\mathbf{j}} \langle D_{\mathbf{i}}|\hat{H}|D_{\mathbf{j}}\rangle \, c_{\mathbf{j}}(\tau). \quad (1)$$

A key insight is that the action of the Hamiltonian can be applied stochastically rather than deterministically: the wavefunction is discretized by using a set of particles with weight ± 1 to represent the coefficients, and is evolved in imaginary time by stochastically creating new particles according to the Hamiltonian matrix (Section 2.4). By starting with just particles on the Hartree–Fock determinant or a small number of determinants, the sparsity of the FCI wavefunction emerges naturally. The FCIQMC algorithm hence has substantially reduced memory requirements[26] and is naturally scalable[43] in contrast to conventional Lanczos techniques. The sign problem manifests itself in the competing in-phase and out-of-phase combinations of particles with positive and negative signs on the same determinant;[42] this is alleviated by exactly canceling particles of opposite sign on the same determinant, a process termed 'annihilation'. This results in the distinctive population dynamics of an FCIQMC simulation, and a system-specific critical population is required to obtain a statistical representation of the correct FCI wavefunction.[42] Once the ground-state FCI wavefunction has been reached, the population is controlled via a diagonal energy offset[9,26] and statistics can be accumulated for the energy estimator and, if desired, other properties.

The stochastic efficiency of the algorithm (determined by the size of statistical errors for a given computer time) can be improved by several approaches: using real weights, rather than integer weights, to represent particle amplitudes;[44,45] a semi-stochastic propagation, in which the action of the Hamiltonian in a small subspace of determinants is applied *exactly*;[44,46] and more efficient sampling of the Hamiltonian by incorporating information about the magnitude of the Hamiltonian matrix elements into the selection probabilities.[47,48]

The initiator approximation[49] (often referred to as i-FCIQMC) only permits new particles to be created on previously unoccupied determinants if the spawning determinant has a weight above a given threshold — this introduces a systematic error which is reduced with increasing particle populations, but effectively reduces the severity of the sign problem. This simple modification has proven remarkably successful and permits FCI-quality calculations on Hilbert spaces orders of magnitude beyond exact FCI.

2.2 Coupled Cluster Monte Carlo

The coupled cluster wavefunction ansatz is $|\Psi_{\mathrm{CC}}\rangle = N e^{\hat{T}} |D_{\mathrm{HF}}\rangle$, where $\hat{T}$ is the cluster operator containing all excitations up to a given truncation level, N is a normalisation factor and $|D_{\mathrm{HF}}\rangle$ the Hartree–Fock determinant. For convenience, we rewrite the wavefunction ansatz as $|\Psi_{\mathrm{CC}}\rangle = t_{\mathrm{HF}} e^{\hat{T}/t_{\mathrm{HF}}} |D_{\mathrm{HF}}\rangle$, where t_{HF} is a weight on the Hartree–Fock determinant, and define $\hat{T} = \sum'_{\mathbf{i}} t_{\mathbf{i}} \hat{a}_{\mathbf{i}}$, where $'$ restricts the sum to be up to the truncation level, $\hat{a}_{\mathbf{i}}$ is an excitation operator (excitor) such that $\hat{a}_{\mathbf{i}} |D_{\mathrm{HF}}\rangle$ results in $|D_{\mathbf{i}}\rangle$ and $t_{\mathbf{i}}$ is the corresponding amplitude. Using the same first-order Euler approach as in FCIQMC gives a similar propagation equation:

$$t_{\mathbf{i}}(\tau + \delta\tau) = t_{\mathbf{i}}(\tau) - \delta\tau \sum_{\mathbf{j}} \langle D_{\mathbf{i}}|\hat{H}|D_{\mathbf{j}}\rangle \, \tilde{t}_{\mathbf{j}}(\tau). \quad (2)$$

The key difference between Eqs. (1) and (2) is $\tilde{t}_{\mathbf{j}} = \langle D_{\mathbf{j}}|\Psi_{\mathrm{CC}}\rangle$ contains contributions from clusters of excitors[30] whereas the FCI wavefunction is a simple linear combination. This is tricky to

evaluate efficiently and exactly each iteration. Instead, $\tilde{t}_{\mathbf{j}}$ is sampled and individual contributions propagated separately.[30,50,51] Bar this complication, the coupled cluster wavefunction can be stochastically evolved using the same approach as used in FCIQMC.

2.3 Density Matrix Quantum Monte Carlo

FCIQMC and CCMC are both ground-state, zero-temperature methods (although excited-state variants of FCIQMC exist[34,39–41]). The exact *thermodynamic* properties of a quantum system in thermal equilibrium can be determined from the (unnormalized) N-particle density matrix, $\hat{\rho}(\beta) = e^{-\beta\hat{H}}$, where $\beta = 1/k_B T$. A direct evaluation of $\hat{\rho}(\beta)$ requires knowledge of the full eigenspectrum of $\hat{H}$, a hopeless task for all but trivial systems. To make progress we note that the density matrix obeys the (symmetrized) Bloch equation

$$\frac{d\hat{\rho}}{d\beta} = -\frac{1}{2}\left[\hat{H}\hat{\rho} + \hat{\rho}\hat{H}\right]. \tag{3}$$

Representing $\hat{\rho}$ in the Slater determinant basis, $\rho_{\mathbf{ij}} = \langle D_{\mathbf{i}}|\hat{\rho}|D_{\mathbf{j}}\rangle$ and again using a first-order update scheme results in similar update equations to FCIQMC and CCMC:

$$\rho_{\mathbf{ij}}(\beta + \delta\beta) = \rho_{\mathbf{ij}}(\beta) - \frac{\delta\beta}{2}\sum_{\mathbf{k}}\left[\langle D_{\mathbf{i}}|\hat{H}|D_{\mathbf{k}}\rangle\,\rho_{\mathbf{kj}}(\beta)\right.$$
$$\left. + \rho_{\mathbf{ik}}(\beta)\,\langle D_{\mathbf{k}}|\hat{H}|D_{\mathbf{j}}\rangle\right]. \tag{4}$$

It follows that elements of the density matrix can be updated stochastically in a similar fashion to FCIQMC and CCMC. $\rho(\beta)$ is a single stochastic measure of the exact density matrix at inverse temperature β. Therefore, unlike FCIQMC and CCMC, multiple independent simulations must be performed in order to gather statistics at each temperature. The simplest starting point for a simulation is at $\beta = 0$, where ρ is the identity matrix. Each simulation (termed 'β-loop') consists of sampling the identity matrix and propagating to the desired value of β. Averaging over multiple β-loops gives thermal properties at all temperatures in the range $[0, \beta]$.

While this scheme is exact (except for small and controllable errors due to finite $\delta\beta$), it suffers from the issue that important states at low temperature may not be sampled in the initial ($\beta = 0$) density matrix, where all configurations are equally important.[33] To overcome this, we write $\hat{H} = \hat{H}^0 + \hat{V}$ and define the auxiliary density matrix $\hat{f}(\tau) = e^{-(\beta-\tau)\hat{H}^0}\hat{\rho}(\tau)$ with the following properties:

$$\hat{f}(0) = e^{-\beta\hat{H}^0}, \tag{5}$$

$$\hat{f}(\beta) = \hat{\rho}(\beta), \tag{6}$$

$$\frac{d\hat{f}}{d\tau} = \hat{H}^0\hat{f} - \hat{f}\hat{H}. \tag{7}$$

We see that with this form of density matrix we can begin the simulation from a mean-field solution defined by $\hat{H}_0$, which should (by construction) lead to a distribution containing the desired important states (such as the Hartree–Fock density matrix element) at low temperature. Furthermore, if $\hat{H}^0$ is a good mean field Hamiltonian then $e^{\beta\hat{H}^0}\hat{\rho}$ is a *slowly varying* function of β, and is thus easier to sample. Comparing Eqs. (3) and (7), we see that $\hat{f}$ can be stochastically sampled in a similar fashion to DMQMC, with minor modifications relative to using the unsymmetrized Bloch equation:[32] i) the choice of $\hat{H}^0$ changes the probability of killing a particle (Section 2.4); ii) the $\tau = 0$ initial configuration must be sampled according to $\hat{H}^0$ rather than the identity matrix; iii) evolving to $\tau = \beta$ gives a sample of the density matrix at inverse temperature β *only* - independent simulations must be performed to accumulate results at different temperatures. We term this method interaction-picture DMQMC (IP-DMQMC).

2.4 Commonality between FCIQMC, CCMC and DMQMC

FCIQMC, CCMC and DMQMC have more similarities than differences: the amplitudes within the wavefunction or density matrix are represented stochastically by a weight, or par-

ticle.[52] These stochastic amplitudes are sampled to produce states, which make up the wavefunction or density matrix. For FCIQMC (DMQMC), a state corresponds to a determinant (outer product of two determinants), and for CCMC corresponds to a term sampled from the cluster expansion corresponding to a single determinant. The stochastic representation of the wavefunction or density matrix is evolved by

spawning sampling the action of the Hamiltonian on each (occupied) state, which requires random selection of a state connected to the original state. The process of random selection ('excitation generation') is system-dependent, as it depends upon the connectivity of the Hamiltonian matrix; efficient sampling of the Hamiltonian has a substantial impact on the stochastic efficiency of a simulation.[44,47,48]

death killing each particle with probability proportional to its diagonal Hamiltonian matrix element.

annihilation combining particles on the same state and canceling out particles with the same absolute weight but opposite sign.

Energy estimators can be straightforwardly accumulated during the evolution process. A parallel implementation distributes states over multiple processors, each of which need only evolve its own set of states. The annihilation stage then requires an efficient process for determining to which processor a newly spawned particle should be sent.[43] For CCMC an additional communication step is required to ensure that the sampling of products of amplitudes is unbiased.[50]

Hence, FCIQMC, CCMC and DMQMC share the majority of the core algorithms in the HANDE-QMC implementations. The primary difference is the representation of the wavefunction or density matrix, and the action of the Hamiltonian in the representation. These differences reside in the outer-most loop of the algorithm and so do not hinder the re-use of components between the methods. This remains the case even for linked coupled cluster Monte Carlo, which applies the similarity-transformed Hamiltonian, $e^{-T}He^{T}$, and the interaction picture formulation of DMQMC.

It is important to note that this core paradigm also covers different approaches to propagation,[34,44,49,53] the initiator approximation,[31,49,54] excitation generators,[47,48] excited states and properties,[41,45,55] and can naturally be applied to different wavefunction Ansätze,[56] which can be added relatively straightforwardly on top of a core implementation of FCIQMC. Due to this, improvements in, say, excitation generators can be immediately used across all methods in HANDE.

3 HANDE-QMC

3.1 Implementation

HANDE-QMC is implemented in Fortran and takes advantage of the increased expressiveness provided by the Fortran 2003 and 2008 standards.[57] Parallelization over multiple processors is implemented using OpenMP (CCMC-only for intra-node shared memory communication) and MPI. Parallelization and the reusability of core procedures have been greatly aided by the use of pure procedures and minimal global state, especially for system and calculation data.

We attempt to use best-in-class libraries where possible. This allows for rapid development and a focus on the core QMC algorithms. HANDE-QMC relies upon MurmurHash2 for hashing operations,[58] dSFMT for high-quality pseudo-random number generation,[59] numerical libraries (cephes,[60] LAPACK, ScaLAPACK, TRLan[61,62]) for special functions, matrix and vector procedures and Lanczos diagonalization, and HDF5 for file I/O.[63] The input file to HANDE-QMC is a Lua script;[64] Lua is a lightweight scripting language designed for embedding in applications and can easily be used from Fortran codes via the AOTUS library.[65] Some of the advantages of using a scripting language for the input file are detailed in Section 5.

Calculation, system settings and other meta-

data are included in the output in the JSON format,[66] providing a good compromise between human- and machine-readable output.

HANDE can be compiled either into a standalone binary or into a library, allowing it to be used directly from existing quantum chemistry packages. CMake[67] is used for the build system, which allows for auto-detection of compilers, libraries and available settings in most cases. A legacy Makefile is also included for compiling HANDE in more complex environments where direct and fine-grained control over settings is useful.

Integrals for molecular and solid systems can be generated by Hartree–Fock calculations using standard quantum chemistry programs, such as Psi4,[68] HORTON,[69] PySCF,[70] Q-Chem,[71] and MOLPRO,[72] in the plain-text FCIDUMP format. HANDE can convert the FCIDUMP file into an HDF5 file, which gives a substantial space saving and can be read in substantially more quickly. For example, an all-electron FCIDUMP for coronene in a Dunning cc-pVDZ basis[73] is roughly 35GB in size and takes 1840.88 seconds to read into HANDE and initialise. When converted to HDF5 format, the resulting file is 3.6GB in size and initialising an identical calculation takes only 60.83 seconds. This is useful in maximizing resource utilization when performing large production-scale calculations on HPC facilities. The memory demands of the integrals are reduced by storing the two-electron integrals only once on each node using either the MPI-3 shared memory functionality or, for older MPI implementations, POSIX shared memory.

In common with several Monte Carlo methods, data points from consecutive iterations are not independent, as the population at a given iteration depends on the population at the previous iteration. This autocorrelation must be removed in order to obtain accurate estimates of the standard error arising from FCIQMC and CCMC simulations[74] and is most straightforwardly done via a reblocking analysis.[75] This can be performed as a post-processing step[76] but is also implemented as an on-the-fly algorithm,[77] which enables calculations to be terminated once a desired statistical error has been reached.

It is often useful to continue an existing calculation; for example to accumulate more statistics to reduce the error bar, to save equilibration time when investigating the effect of calculation parameters or small geometry changes, or for debugging when the bug is only evident deep into a calculation. To aid these use cases, calculations can be stored and resumed via the use of restart files. The state of the pseudo-random number generator is included in the restart files such that restarted calculations follow the same Markov chain as if they had been run in a single calculation assuming the same calculation setup is used. We use the HDF5 format and library for efficient I/O and compact file sizes. A key advantage of this approach is that it abstracts the data layout into a hierarchy (termed *groups* and *datasets*). This makes extending the restart file format to include additional information whilst maintaining backward compatibility with previous calculations particularly straightforward. Each calculation is labeled with a universally unique identifier (UUID),[78] stored in the restart file and included in the metadata of subsequent calculations. This is critical for tracing the provenance of data generated over multiple restarted calculations.

Extensive user-level documentation is included in the HANDE-QMC package[79] and details compilation, input options, running HANDE and calculation analysis. The documentation also includes several tutorials on FCIQMC, CCMC and DMQMC, which guide new users through generating the integrals (if required), running a QMC calculation along with enabling options for improving stochastic efficiency, and analysing the calculations. The HANDE source code is also heavily commented and contains extensive explanations on the theories and methods implemented (especially for CCMC), and data structures. Each procedure also begins with a comment block describing its action, inputs and outputs. We find this level of developer documentation to be extremely important for onboarding new developers and making HANDE accessible to modifications by other researchers.

3.2 Development methodology

The HANDE-QMC project is managed using the Git distributed version control system.[80] A public Git repository is hosted on GitHub[81] and is updated with new features, improvements and bug fixes. We also use a private Git repository for more experimental development and research; this allows for new features to be iterated upon (and potentially changed or even removed) without introducing instability into the more widely available code.[82] We regularly update the public version, from which official releases are made, with the changes made in the private repository. The code contains a comprehensive test suite[83] and when new features are implemented they are benchmarked against existing codes wherever possible. Further details of our development practices such as our development philosophy and the extensive continuous integration set up using Buildbot[84] are outlined in Ref. 85.

3.3 pyhande

Interpretation and analysis of calculation output is a critical part of computational science. While we wrote scripts for performing common analyses, such as reblocking to remove the effect of autocorrelation from estimates of the standard error, we found that users would write ad-hoc, fragile scripts for extracting other useful data, which were rarely shared and contained overlapping functionality. This additional barrier also hindered curiousity-driven exploration of results. To address this, the HANDE-QMC package includes pyhande, a Python library for working with HANDE calculation outputs. pyhande extracts metadata (including version, system and calculation parameters, calculation UUID) into a Python dictionary and the QMC output into a Pandas[86] `DataFrame`, which provides a powerful abstraction for further analysis. pyhande includes scripts and functions to automate common tasks, including reblocking analysis, plateau and shoulder[31] height estimation, stochastic inefficiency estimation[87] and reweighting to reduce the bias arising from population control.[9,88] We have found that the development of pyhande has aided reproducibility by providing a single, robust implementation for output parsing and common analyses, and has made more complex analyses more straightforward by providing rich access to raw data in a programmable environment. Indeed, many functions included in pyhande began as exploratory analysis in a Python shell or a Jupyter notebook. The HANDE-QMC documentation also details pyhande and the tutorials include several examples of using pyhande for data analysis. pyhande makes extensive use of the Python scientific stack (NumPy,[89] SciPy,[90] Pandas[86] and Matplotlib[91]).

3.4 License

HANDE-QMC is licensed under the GNU Lesser General Public License, version 2.1. The LGPLv2.1 is a weak copyleft license,[92,93] which allows the QMC implementations to be incorporated in both open- and closed-source quantum chemistry codes while encouraging developments and improvements to be contributed back or made available under the same terms.[94] pyhande is licensed under the 3-Clause BSD License,[95] in keeping with many scientific Python packages.

4 Example results

In this section we present calculations to demonstrate the core functionality included in HANDE-QMC: we consider a small molecule (nitric oxide); the uniform electron gas in the zero-temperature ground state and at finite temperatures; and a periodic solid, diamond, with k-point sampling. The supplementary material includes a tutorial on running and analyzing FCIQMC on the water molecule in cc-pVDZ basis, which is easily accessible by deterministic methods and can be easily performed on any relatively modern laptop.

4.1 Computational details

All calculations in this section were run with HANDE versions earlier than version 1.3. In-

tegrals were generated using PySCF, Psi4 and Q-Chem. Input, output and analysis scripts are available under a Creative Commons License at `https://doi.org/10.17863/CAM.31933` containing specifics on which version is used for some calculations, and which SCF program is used. Orbital visualizations were made with IboView.[96]

4.2 Molecules: Nitric oxide

Nitric oxide is an important molecule, perhaps most notably as a signalling molecule in multiple physiological processes. Here, we consider NO in a cc-pVDZ basis set,[73] correlating all 15 electrons. The FCI space size is $\sim 10^{12}$, and so is somewhat beyond the reach of exact FCI approaches. We consider initiator FCIQMC, using a walker population of 8×10^6, which is more than sufficient to achieve an accuracy of $\sim 0.1 mE_\mathrm{h}$. This is then compared to CCMC results for the CCSD, CCSDT and CCSDTQ Ansätze. An unrestricted Hartree–Fock (UHF) molecular orbital basis is used. The computational resources to perform this study are modest compared to state-of-the-art FCIQMC simulations, never using more than about 100 processing cores.

In Figure 1 and Table 1, results are presented for this system at varying internuclear distances. Remarkably good agreement between CCSDTQ-MC and the i-FCIQMC is achieved, with CCSDT-MC also performing extremely well. Statistical errors do not pose any issue in these results, as is typically the case for FCIQMC and CCMC simulations; all such error bars are naturally of order $0.1 mE_\mathrm{h}$ or less. For i-FCIQMC results the semi-stochastic adaptation was used,[44,46] choosing the deterministic space by the approach of Ref. 46. Fig. (2) demonstrates such simulations before and after enabling semi-stochastic propagation, and the benefits are clear. Indeed, i-FCIQMC results here have statistical errors of order $\sim 1\mu E_\mathrm{h}$ or smaller.

CCMC calculations were performed with real weights using the even selection algorithm.[51] For the largest calculations, CCSDTQ-MC, heatbath excitation generators were used with up to 4.5×10^6 occupied excitors, parallelizing over 96 cores. For comparison, deterministic single reference CCSDTQ calculations performed with the MRCC program package[97] required storage of 2.1×10^7 amplitudes, but did not converge beyond $R = 1.7$Å.

Table (1) also shows the percentage of correlation energy captured by the various levels of CC, compared to i-FCIQMC. CCSD and CCSDT capture $> 92\%$ and $> 98\%$ of the correlation energy, respectively, with CCSDTQ essentially exact, and the percentage decreasing with increasing bond length as expected. The CCMC approach is particularly appropriate for such high-order CC calculations, where stochastic sampling naturally takes advantage of the sparse nature of the CC amplitudes.
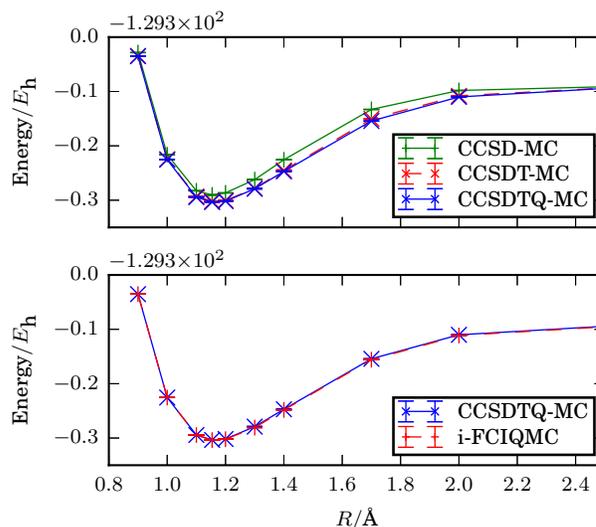

Figure 1: The binding curve of NO in a cc-pVDZ basis set, correlating all electrons. Stochastic error bars are not visible on this scale, but all are smaller than $1\,mE_\mathrm{h}$. For better resolution in the differences between methods, see Table (1).

4.3 Model Solid: Uniform electron gas

HANDE also has built-in capability to perform calculations of model systems commonly used in condensed matter physics, specifically the uniform electron gas (UEG),[98–100] the Hubbard model,[101–103] and the Heisenberg model.[32,104]

Table 1: CCMC and i-FCIQMC results for the NO molecule in a cc-pVDZ basis set, correlating all electrons, as plotted in Fig. (1). UHF orbitals were used. Numbers in parentheses show statistical error bars, not systematic initiator error, which is estimated to be $\sim 0.1\mathrm{m}E_h$ for i-FCIQMC results. i-FCIQMC results used the semi-stochastic adaptation with a deterministic space of size 2×10^4. The results of such a semi-stochastic approach are demonstrated in Fig. (2). The final three columns show the percentage of correlation energy recovered by CCSD-MC, CCSDT-MC and CCSDTQ-MC, compared to i-FCIQMC. i-FCIQMC calculations were performed with 8×10^6 walkers, and CCMC calculations used at most 7×10^6 excips.

| R/Å	(Total energy + $129E_h$)/E_h				Correlation energy recovered (%)		
	CCSD	CCSDT	CCSDTQ	i-FCIQMC	CCSD	CCSDT	CCSDTQ
0.9	-0.328507(1)	-0.3346(1)	-0.33523(4)	-0.335225(2)	97.7330(6)	99.78(5)	100.00(1)
1.0	-0.5162(2)	-0.52478(2)	-0.525448(6)	-0.525470(2)	97.06(8)	99.779(6)	99.993(2)
1.1	-0.582684(9)	-0.59317(8)	-0.59447(3)	-0.594565(3)	96.435(3)	99.58(2)	99.973(9)
1.154	-0.5904(5)	-0.6018(3)	-0.6035(2)	-0.603772(2)	96.1(2)	99.43(9)	99.92(5)
1.2	-0.58653(3)	-0.6005(4)	-0.6018(2)	-0.602136(3)	95.541(8)	99.5(1)	99.89(7)
1.3	-0.5622(2)	-0.5782(4)	-0.5790(6)	-0.580833(3)	94.67(5)	99.2(1)	99.5(2)
1.4	-0.5256(2)	-0.5451(10)	-0.5471(7)	-0.548340(3)	93.34(7)	99.1(3)	99.6(2)
1.7	-0.43299(10)	-0.4503(5)	-0.4543(1)	-0.455765(4)	92.13(3)	98.1(2)	99.48(4)
2.0	-0.39816(6)	-0.40800(9)	-0.41010(6)	-0.411350(2)	94.45(2)	98.59(4)	99.47(2)
2.5	-0.39132(5)	-0.39371(8)	-0.39434(2)	-0.3954786(4)	98.05(2)	99.17(4)	99.467(8)
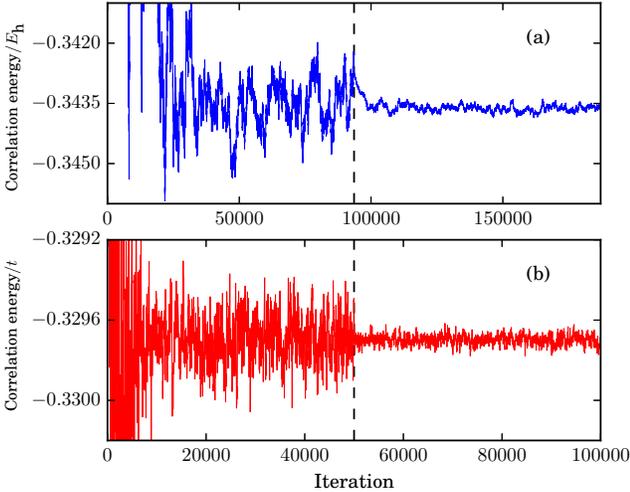
Figure 2: Example simulations in HANDE-QMC using the semi-stochastic FCIQMC approach of Umrigar and co-workers.[44] Vertical dashed lines show the iteration where the semi-stochastic adaptation is begun, and the resulting reduction in noise is clear thereafter. (a) NO in a cc-pVDZ basis set, with all electrons correlated, at an internuclear distance of 1.154Å. The deterministic space is of size 2×10^4. (b) A half-filled two-dimensional 18-site Hubbard model at $U/t = 1.3$, using a deterministic space of size 10^4.

Such model systems have formed the foundation of our understanding of simple solids and strongly correlated materials, and are a useful testing ground for new computational approaches. Studying the UEG, for example, has provided insight into the accuracy of many-body electronic structure methods and has been a critical ingredient for the development of many of the exchange-correlation kernels used in Kohn–Sham density functional theory.[105–107]

The UEG has been used recently as a means to benchmark and test performance of new methods, such as modifications to diffusion Monte Carlo (DMC), as well as low orders of coupled cluster theory[31,108–115] and FCIQMC.[116–121]

A recent CCMC study[118] employing coupled cluster levels up to CCSDTQ5 used HANDE to compute the total energy of the UEG at $r_s = [0.5, 5]\mathrm{a}_0$, the range relevant to electron densities in real solids.[100] The results suggest that CCSDTQ might be necessary at low densities beyond $r_s = 3\mathrm{a}_0$[118] in order to achieve chemical accuracy, whilst CCSDTQ5 was necessary to reproduce FCIQMC to within error bars.[116–119]

HANDE was also used in the resolution

Table 2: Ground-state exchange-correlation energies (E_{xc}) for the UEG at $r_s = 1a_0$, comparing various levels of coupled cluster theory with FCIQMC. Exchange-correlation energies were calculated using data from Ref. [118]

Method	E_{xc}/E_h
CCSD-MC	-0.551128(6)
CCSDT-MC	-0.55228(1)
CCSDTQ-MC	-0.55231(1)
CCSDTQ5-MC	-0.55232(1)
FCIQMC	-0.55233(1)

of a discrepancy between restricted path-integral Monte Carlo and configuration path-integral Monte Carlo data for the exchange-correlation energy of the UEG necessary to parametrize DFT functionals at finite temperature.[54,122–130]. The UEG at finite temperatures is parametrized by the density and the degeneracy temperature, $\Theta = T/T_F$, where T_F is the Fermi temperature[131]. When both $r_s \approx 1$ and $\Theta \approx 1$ the system is said to be in the warm dense regime, a state of matter which is to found in planetary interiors[132] and can be created experimentally in inertial confinement fusion experiments.[133]

Here, we show that use of HANDE can facilitate straightforward benchmarking of model systems at both zero and finite temperature. In Fig. 3 we compare DMQMC data for the 14-electron, spin-unpolarized UEG at finite Θ to zero temperature ($\Theta = 0$) energies found using CCMC and FCIQMC[118] for $r_s = 1a_0$. We compute the exchange-correlation internal energy

$$E_{XC}(\Theta) = E_{QMC}(\Theta) - T_0(\Theta), \qquad (8)$$

where $E_{QMC}(\Theta)$ is the QMC total energy of the UEG and T_0 is the ideal kinetic energy of the same UEG. Even at $r_s = 1a_0$, coupled cluster requires contributions from triple excitations to obtain FCI-quality energies; CCSD differs by about 1mHa. DMQMC results tend to the expected zero temperature limit given by both FCI and CC. Ground-state values from coupled cluster and FCIQMC are presented in Table. (2), to make the small differences between high-accuracy methods clearer.
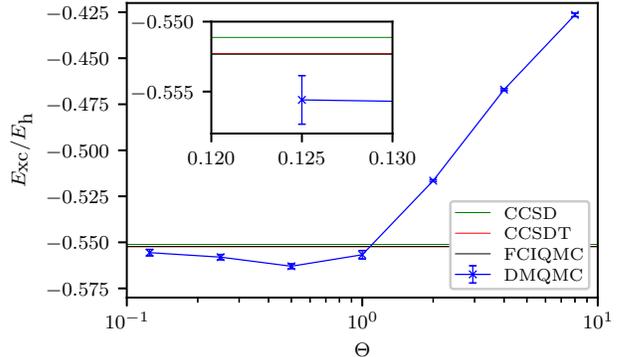

Figure 3: The exchange-correlation energy (E_{xc}) for the UEG at $r_s = 1a_0$ as a function of temperature Θ using DMQMC (Ref. 54). The horizontal lines represent basis set extrapolated CCSD, CCSDT and FCIQMC exchange-correlation energies energies (Ref. 118). Error bars on CCMC and FCIQMC results are too small to be seen on this scale. CCSDT and FCIQMC values cannot be distinguished on this scale. See Table. (2) for numerical values for CCSD to CCSDTQ5 in the ground state.

4.4 Solids: Diamond

Finally, we apply HANDE-QMC to a real periodic solid, diamond, employing $\boldsymbol{k}$ point sampling. CCMC has been applied to $1\times1\times1$ (up to CCSDTQ), $2\times1\times1$ (up to CCSDT), $2\times2\times1$ and $2\times2\times2$ (up to CCSD) $\boldsymbol{k}$ point meshes and non-initiator FCIQMC to a $1\times1\times1$ $\boldsymbol{k}$ point mesh in a GTH-DZVP[134][†] basis, and a GTH-pade pseudo-potential.[136,137] There were 2 atoms, 8 electrons in 52 spinorbitals per $\boldsymbol{k}$ point. Integral files have been generated with PySCF[70] using Gaussian density fitting.[138] Orbitals were obtained from density functional theory using the LDA Slater-Vosko-Vilk-Nusair (SVWN5) exchange-correlation functional[139] to write out complex valued integrals at different $\boldsymbol{k}$ points, and HANDE's read–in functionalities were adapted accordingly. Details of this will be the subject of a future publication on solid-state calculations. The *heat bath uniform singles*[47,48] or the *heat bath Power–Pitzer ref.* excitation generator[48] and even selection[51] or

---

[†]As used in PySCF,[70] and CP2K,[135] https://www.cp2k.org/.

multi-spawn[50] sampling were used.

Deterministic coupled cluster has been applied to diamond previously; Booth et al.[140] have investigated diamond with CCSD, CCSD(T)[141] and FCIQMC in a basis of plane waves with the projector augmented wave method;[142] McClain et al.[143] studied diamond with CCSD using GTH pseudo-potentials in DZV, DZVP, TZVP basis sets[134,136,137†]; Gruber et al.[144] used CCSD with (T) corrections in an MP2 natural orbital basis.[145]

The lattice constant was fixed to 3.567Å, as in the study by McClain et al.[143] Figure 4 shows the correlation energy as a function of number of $\boldsymbol{k}$ points comparing the CCMC and FCIQMC results to the CCSD results obtained using PySCF and the CCSD results of McClain et al.[143] The correlation energy given here is calculated with respect to the HF energy, as the correlation energy from using DFT orbitals, added to the difference of energy of reference determinant consisting of DFT orbitals and HF SCF energy. Differences in convergences are due to the use of differently optimized orbitals, and a different treatment of the exchange integral (which will feature in a future publication). In the case of CCMC, FCIQMC and CCSD-PySCF the $\boldsymbol{k}$ point mesh has been shifted to contain the Γ point, while McClain et al.[143] used Γ point centered (not shifted) meshes, which explains the larger difference between CCSD-McClain et al. and the rest of the data. An accuracy of (0.01-0.1) eV/unit ((0.00037-0.0037) E_h/unit) might be required to accurately predict, for example crystal structures,[146] so these limited $\boldsymbol{k}$-point mesh results suggest that at least CCSDT level is required for reasonable accuracy, possibly CCSDTQ. Nonetheless, we have not considered larger basis sets, additional $\boldsymbol{k}$ points, and other important aspects required for an exhaustive study.

Figure 4: Difference between the total and Hartree-Fock energy per $\boldsymbol{k}$ point for diamond using CCMC (CCSD to CCSDTQ) and (non-initiator) FCIQMC based on DFT orbitals. The CCSDTQ and the FCIQMC data point overlap to a large extent. The CCSD-PySCF data was run with Hartree-Fock orbitals. In the case of CCMC, FCIQMC and CCSD-PySCF the mesh has been shifted to contain the Γ point. CCSD-McClain et al. is data from Figure 1 in McClain et al.[143] using PySCF; we show only their data up to 12 k-points for comparison. Both studies used the DZVP basis set and GTH pseudopotentials.

5 Discussion

This article has presented the key functionality included in HANDE-QMC: efficient, extensible implementations of the full configura-
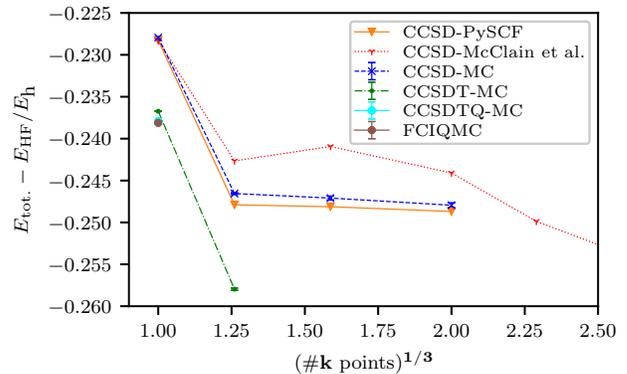
tion interaction quantum Monte Carlo, coupled cluster Monte Carlo and density matrix quantum Monte Carlo methods. Advances such as semi-stochastic propagation in FCIQMC[44,46] and efficient excitation generators[47,48] are also implemented. HANDE-QMC can be applied to model systems – the Hubbard, Heisenberg and uniform electron gas models – as well as molecules and solids.

We have found using a scripting language (Lua) in the input file[147] to be extremely beneficial – for example, in running multi-stage calculations, enabling semi-stochastic propagation after the most important states have emerged, irregular output of restart files, or for enabling additional output for debugging at a specific point in the calculation. As with (e.g.) Psi4, PySCF and HORTON, we find this approach far more flexible and powerful than a custom declarative input format used in many other scientific codes.

We are strong supporters of open-source software in scientific research and are glad that the HANDE-QMC package has been used in others research in ways we did not envisage, including in the development of Adaptive Sampling Configuration Interaction (ASCI),[53] understanding the inexact power iteration method[148] and in selecting the P subspace in the CC(P;Q) method.[149] We believe one reason for this is that the extensive user- and developer-level documentation makes learning and developing HANDE-QMC rather approachable. Indeed, five of the authors of this paper made their first contributions to HANDE-QMC as undergraduates with little prior experience in software development or computational science. In turn, HANDE-QMC has greatly benefited from existing quantum chemistry software, in particular integral generation from Hartree–Fock calculations in Psi4,[68] Q-Chem[71] and PySCF.[70] We hope in future to couple HANDE-QMC to such codes to make running stochastic quantum chemistry calculations simpler and more convenient. To this end, some degree in standardization of data formats to make it simple to pass data (e.g. wavefunctions amplitudes) between codes would be extremely helpful in connecting libraries, developing new methods[149] and reproducibility.

We close by echoing the views of the Psi4 developers:[68] 'the future of quantum chemistry software lies in a more modular approach in which small, independent teams develop reusable software components that can be incorporated directly into multiple quantum chemistry packages' and hope that this leads to an increased vibrancy in method development.

A   An introductory tutorial to HANDE-QMC

In the following we present an introductory tutorial, demonstrating how to perform basic FCIQMC and i-FCIQMC simulations with the HANDE-QMC code. More extensive tutorials, including for CCMC and DMQMC, exist in the HANDE-QMC documentation. Here we take the water molecule at its equilibrium geometry, in a cc-pVDZ basis set[73] and correlating all electrons. This is a simple example, but has a Hilbert space dimension of $\sim 5 \times 10^8$, making an exact FCI calculation non-trivial to perform.

A.1   A basic i-FCIQMC simulation

The input file for HANDE-QMC is a Lua script. The basic structure of such an input file is shown in Fig. (5).

In this the system is entirely determined by the integral file, "INTDUMP", which stores all of the necessary 1- and 2-body molecular integrals. For this tutorial, the integral file was

```
sys = read_in {
    int_file = "INTDUMP",
}

fciqmc {
    sys = sys,
    qmc = {
        tau = 0.01,
        tau_search = true,
        rng_seed = 8,
        init_pop = 500,
        mc_cycles = 5,
        nreports = 3*10^3,
        target_population = 10^4,
        excit_gen = "heat_bath",
        initiator = true,
        real_amplitudes = true,
        spawn_cutoff = 0.1,
        state_size = -1000,
        spawned_state_size = -100,
    },
}
```

Figure 5: An example input file for an i-FCIQMC simulation on a molecular system. The results of such a simulation are presented in Fig. (6).

generated through the Psi4 code.[68] Both the "INTDUMP" file, and the Psi4 script used to generate it, are available in additional material. As discussed in the main text, the integral file may be generated by multiple other quantum chemistry packages.[69–72]

In general, the system may be defined by specifying additional parameters, including the number of electrons, the spin quantum number (M_s), the point group symmetry label, and a CAS subspace, for example:

```
sys = read_in {
    int_file = "INTDUMP",
    nel = 10,
    ms = 0,
    sym = 0,
    CAS = {8, 23},
}
```

The input file then calls the `fciqmc{...}` function, which performs an FCIQMC simulation with the provided system and parameters. There are several options here; most are self-evident and are described in detail in the HANDE-QMC documentation. `tau` specifies the time step size, and `tau_search=true` updates this time step to an optimal value during the simulation. `init_pop` specifies the initial particle population, and `target_population` the value at which this population will attempt to stabilize. `excit_gen` specifies the excitation generator to be used. This option is not required, although the heat-bath algorithm of Umrigar and co-workers[47] that we have adapted for HANDE-QMC as explained in Ref.,[48] as used here, is a sensible choice in small systems. `initiator=true` ensures that the initiator adaptation, i-FCIQMC, is used. `real_amplitudes=true` ensures that non-integer particle weights are used. This leads to improved stochastic efficiency, and so is always recommended. Lastly, `state_size` and `spawned_state_size` specify the memory allocated to the particle and spawned particle arrays, respectively - a negative sign is used to specify these values in megabytes (thus 1GB and 100MB, here).

The input file is run with

```
$ mpiexec hande.x hande.lua > hande.out
```

with the MPI command varying between implementations in the usual way. The results of the input file in Fig. (5) are presented in Fig. (6).

Because of the correlated nature of the QMC data, care must be taken when estimating error bars; a large number of iterations must typically be performed, allowing data to become sufficiently uncorrelated. This task can be error-prone for new users (and old ones). HANDE-QMC includes a Python script, `reblock_hande.py`, which performs a rigorous blocking analysis of the simulation data, automatically detecting if sufficient iterations have been performed and, if so, choosing the optimal block length to provide final estimates.

This final energy estimate can be obtained by

```
$ reblock_hande.py --quiet hande.out
```

The usual estimator for the correlation energy (E_{corr}) is the Hartree–Fock projected estimator:

$$E_{\mathrm{corr}} = \frac{\langle D_0|(\hat{H} - E_{\mathrm{HF}}\,\mathbb{1})|\Psi_0\rangle}{\langle D_0|\Psi_0\rangle}, \qquad (9)$$

$$= \frac{\sum_{i\neq 0} C_i \langle D_0|\hat{H}|D_i\rangle}{C_0}, \qquad (10)$$

where $|D_0\rangle$ is the Hartree–Fock determinant and E_{HF} is the Hartree–Fock energy. C_i are the particle amplitudes, with C_0 being the Hartree–Fock amplitude. Because both the numerator and denominator are random variables, they should be averaged separately, *before* performing division. It is therefore important that data be averaged from the point where both the numerator and denominator have converged individually; in some cases the energy itself may appear converged while the numerator and denominator are still converging. This does not occur in the current water molecule case, as can be seen in Fig. (6), where the numerator and denominator are plotted in (b) and (c), respectively. Here, all relevant estimates appear converged by iteration ~ 1000.

The `reblock_hande.py` script will automatically detect when the required quantities have converged, in order to choose the iteration from which to start averaging data. However, a start-
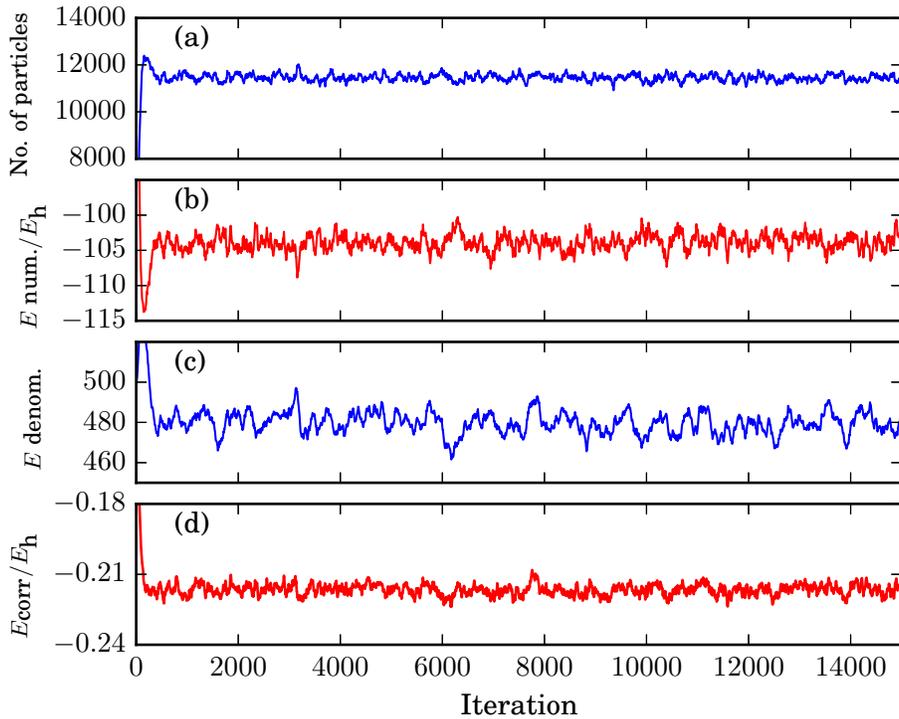
Figure 6: The results of running the input file in Fig (5). (a) shows the particle population, stabilizing slightly above the targeted value of 10^4. (b) shows the numerator of the energy estimator, $\sum_{i \neq 0} C_i \langle D_0 | \hat{H} | D_i \rangle$, as discussed in the main text. (c) shows the energy denominator, which is the number of particles on the Hartree–Fock determinant. (d) shows the correlation energy estimates themselves.

ing iteration may be manually provided using `--start`. In general it is good practice to manually plot simulation data, as in Fig. (6), to check that behavior is sensible. In this case, the `reblock_hande.py` script automatically begins averaging from iteration number 1463, which is appropriate.

A.2 Converging initiator error

After running the `reblock_hande.py` script, the correlation energy estimate can be read off simply as $E_{corr} = -0.2166(2)E_h$. This compares well to the exact FCI energy of $E_{FCI} = -0.217925E_h$, in error by $\sim 1.3 \mathrm{m} E_h$, despite using only $\sim 10^4$ particles to sample a space of dimension $\sim 5 \times 10^8$.

Nonetheless, an important feature of i-FCIQMC is the ability to converge to the exact result by varying only one parameter, the particle population. This is possible by running multiple i-FCIQMC simulation independently.

However, one can make use of the Lua input file with HANDE-QMC to perform an arbitrary number of simulations with a single input file, as shown by example in Fig. (7). Here, `targets` is a table containing particle populations from 2×10^3, and doubling until 1.28×10^5. We loop over all target populations and perform an FCIQMC simulation for each.

Running the `reblock_hande.py` script on the subsequent output file gives the results in Table (3). The final column gives the projected energy estimate of the correlation energy, and is plotted in Fig. (8), with comparison to the FCI energy. Accuracy within $1 \mathrm{m} E_h$ is reached with $N_w = 2 \times 10^4$, and an accuracy of $0.1 \mathrm{m} E_h$ by $N_w = 2 \times 10^5$.

It is simple to perform a semi-stochastic i-FCIQMC simulation. To do this, as well as passing `sys` and `qmc` parameters to the `fciqmc` function, one should also pass a `semi_stoch` table. The simplest form for this table, which is almost always appropriate, is the following:

Table 3: Output of the HANDE-QMC reblocking script, on the simulation with the input file of Fig. (7). The final column gives the estimates of the correlation energy, as determined from the projected energy estimator.

		Block from	# H psips	$\sum H_{0j}N_j$	N_0	Shift	Proj. Energy
hande.out	0	1.83000000e+03	2292(4)	-36.77(8)	172.6(5)	-0.210(3)	-0.2131(3)
	1	1.81800000e+03	4602(5)	-56.4(1)	262.5(6)	-0.213(2)	-0.2148(2)
	2	1.47300000e+03	9108(7)	-88.29(9)	408.2(5)	-0.213(1)	-0.2163(2)
	3	1.78100000e+03	19050(10)	-151.5(1)	697.0(6)	-0.217(1)	-0.2173(2)
	4	1.97200000e+03	38150(10)	-276.7(1)	1270.0(6)	-0.2188(5)	-0.21784(6)
	5	2.06500000e+03	74310(30)	-528.3(2)	2428(1)	-0.2193(6)	-0.21761(8)
	6	1.82500000e+03	152900(30)	-1081.4(4)	4964(2)	-0.2186(4)	-0.21787(5)

```
sys = read_in {
    int_file = "INTDUMP",
}

targets = {2*10^3, 4*10^3, 8*10^3,
    1.6*10^4, 3.2*10^4, 6.4*10^4,
    1.28*10^5}

for i,target in ipairs(targets) do
    fciqmc {
        sys = sys,
        qmc = {
            tau = 0.01,
            rng_seed = 8,
            init_pop = target/20,
            mc_cycles = 5,
            nreports = 3*10^3,
            tau_search = true,
            target_population = target,
            excit_gen = "heat_bath",
            initiator = true,
            real_amplitudes = true,
            spawn_cutoff = 0.1,
            state_size = -1000,
            spawned_state_size = -100,
        },
    }
end
```

Figure 7: An example input file showing how to use Lua features to perform multiple simulations in a single input file, with particle populations from 2000 to 128,000.
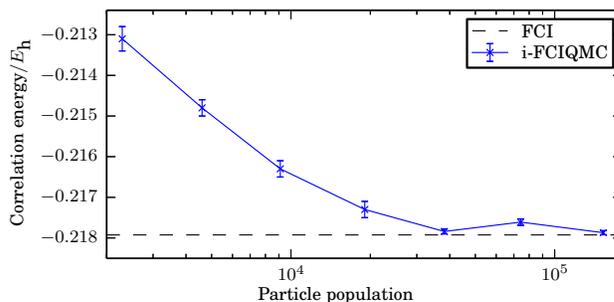

Figure 8: Initiator convergence for the water molecule in a cc-pVDZ basis set, with all electrons correlated. Results were obtained by running the input file of Fig. (7).

```
semi_stoch = {
    size = 10^4,
    start_iteration = 2*10^3,
    space = "high",
},
```

The "high" option generates a deterministic space by choosing the most highly-weighted determinants in the FCIQMC wave function at the given iteration (which in general should be an iteration where the wave function is largely converged), 2×10^3 in this case. The total size of the deterministic space is given by the size parameter, 10^4 in this case.

B Parallelization

In this appendix, we describe two techniques that can optimize the FCIQMC parallelization, *load balancing* and *non-blocking communication*. Parallelization of CCMC has been ex-

plained in Ref. 50 but does not yet make use of *non-blocking communication.*

By and large, HANDE's FCIQMC implementation follows the standard parallel implementation of the FCIQMC algorithm, a more complete description of which can be found in Ref. 43. In short, each processor stores a sorted main list of instantaneously occupied determinants containing the determinant's bit string representation, the walker's weight as well as any simulation dependent flags. For each iteration every walker is given the chance to spawn to another connected determinant, with newly spawned walkers being added to a second spawned walker array. After evolution a collective `MPI_AlltoAllv` is set up to communicate the spawned walker array to the appropriate processors. The annihilation step is then carried out by merging the subsequently sorted spawned walker array with the main list.

During the simulation every walker needs to know which processor a connected determinant resides on but naturally can not store this mapping. In order to achieve a relatively uniform distribution of determinants at a low computational cost, each walker is assigned to a processors p as

$$p(|D_\mathbf{i}\rangle) = \text{hash}(|D_\mathbf{i}\rangle) \bmod N_\text{p}, \qquad (11)$$

where N_p is the number of processors and hash is a hash function[58].

B.1   Load Balancing

The workload of the algorithm is primarily determined by the number of walkers on a given processor, but the above hashing procedure distributes work to processors on a determinant basis. For the hashing procedure to be effective we require that the average population for a random set of determinants to be roughly uniform. Generally hashing succeeds in this regard and one finds a fairly even distribution of both walkers and determinants. When scaling a problem of a fixed size to more processors, i.e. strong scaling, one observes that the distribution loses some of its uniformity with certain processors becoming significantly under

and over populated which negatively affects the parallelism[43]. This is to be expected as in the limit $N_\text{p} \to N_\text{Dets}$ there would be quite a pronounced load imbalance unless each determinant's coefficient was of a similar magnitude (which can often be the case for strongly correlated systems). Naturally this limit is never reached, but the observed imbalance is largely a consequence of this increased refinement.

In HANDE we optionally use dynamic load balancing to achieve better parallel performance. In practice, we define an array p_map as

$$p_\text{map}(i) = i \bmod N_\text{p}, \qquad (12)$$

so that its entries cyclically contain the processor IDs, $0, \ldots, N_\text{p} - 1$. Determinants are then initially mapped to processors as

$$p(|D_\mathbf{i}\rangle) = p_\text{map}\Big(\text{hash}(|D_\mathbf{i}\rangle) \bmod N_\text{p} \times M\Big), \tag{13}$$

where M is the bin size. Eq. (13) reduces to Eq. (11) when $M = 1$.

The walker population in each of these M bins on each processor can be determined and communicated to all other processors. In this way, every processor knows the total distribution of walkers across all processors. In redistributing the $N_\text{p} \times M$ bins we adopt a simple heuristic approach by only selecting bins belonging to processors whose populations are either above or below a certain user defined threshold. By redistributing bins in order of increasing population we can, in principle, isolate highly populated determinants while also allowing for a finer distribution.

This procedure translates to a simple modification of p_map so that its entries now contain the processor IDs which give the determined optimal distribution of bin.

Finally, the walkers which reside in the chosen bins have to be moved to their new processor, which can simply be achieved using a communication procedure similar to that used for the annihilation stage. Some care needs to be taken that all determinants are on their correct processors at a given iteration so that annihilation takes place correctly.

Once the population of walkers has stabilised

the distribution across processors should be roughly constant, although small fluctuations will persist. With this in mind redistribution should only occur after this stabilisation has occurred and also should not need to occur too frequently. This ensures that the computational cost associated with performing load balancing is fairly minor in a large calculation. Additionally as M is increased the optimal distribution of walkers should be approached, although with an increase in computational effort.

B.2 Non-blocking communication

HANDE also makes use of non-blocking asynchronous communication to alleviate latency issues when scaling to large processor counts.[150] Using asynchronous communications is nontrivial in HANDE due to the annihilation stage of FCIQMC-like algorithms. We use the following algorithm: Consider the evolution of walkers from τ to $\tau + \Delta\tau$, then for each processor the following steps are carried out:

1. Initialise the non-blocking receive of walkers spawned onto the current processor from time τ.

2. Evolve the main list to time $\tau + \Delta\tau$.

3. Complete the receive of walkers.

4. Evolve the received walkers to $\tau + \Delta\tau$.

5. Annihilate walkers spawned from the evolution of the two lists as well as the evolved received list with the main list on this processor.

6. Send remaining spawned walkers to their new processors.

While this requires more work per iteration, it should result in improved efficiency if the time take to complete this work is less than the latency time. This also ensures faster processors can continue doing work, i.e. evolving the main list, while waiting for other processors to finish evolving their main lists. For communications to be truly overlapping the slowest processor would need to complete the steps above before the fastest processor reaches step (3), otherwise there will be latency as the received list cannot be evolved before all walkers spawned onto a given processor are received.

It should be pointed out that walkers spawned onto a processor at time τ are only annihilated with the main list after evolution to $\tau + \Delta\tau$, which differs from the normal algorithm. While annihilation is vital to attaining converged results[26,42] the times at which it takes place is somewhat arbitrary, once walkers are annihilated at the same point in simulation time. Communication between processors is also required when collecting statistics, however the usual collectives required for this can simply be replaced by the corresponding non-blocking procedures. This does require that information is printed out in a staggered fashion but this is of minor concern.

Notes and References

(1) Foulkes, W. M. C.; Mitas, L.; Needs, R. J. et al. Quantum Monte Carlo simulations of solids. *Rev. Mod. Phys.* **2001**, *73*, 33–83.

(2) McMillan, W. L. Ground State of Liquid ^{4}He. *Phys. Rev.* **1965**, *138*, A442–A451.

(3) Umrigar, C. J.; Wilson, K. G.; Wilkins, J. W. Optimized Trial Wave Functions for Quantum Monte Carlo Calculations. *Phys. Rev. Lett.* **1988**, *60*, 1719.

(4) Umrigar, C. J.; Toulouse, J.; Filippi, C. et al. Alleviation of the Fermion-Sign Problem by Optimization of Many-Body Wave Functions. *Phys. Rev. Lett.* **2007**, *98*, 110201.

(5) Neuscamman, E.; Umrigar, C. J.; Chan, G. K.-L. Optimizing large parameter sets in variational quantum Monte Carlo. *Phys. Rev. B* **2012**, *85*, 045103.

(6) Neuscamman, E. Variation after re-

sponse in quantum Monte Carlo. *J. Chem. Phys.* **2016**, *145*, 081103.

(7) Grimm, R. C.; Storer, R. G. Monte-Carlo solution of Schrödinger's equation. *J. Comput. Phys.* **1971**, *7*, 134.

(8) Anderson, J. B. A random-walk simulation of the Schrodinger equation: H_3^+. *J. Chem. Phys.* **1975**, *63*, 1499.

(9) Umrigar, C. J.; Nightingale, M. P.; Runge, K. J. A diffusion Monte Carlo algorithm with very small time-step errors. *J. Chem. Phys.* **1993**, *99*, 2865.

(10) Kim, J.; Baczewski, A. D.; Beaudet, T. D. et al. QMCPACK: an open source ab initio quantum Monte Carlo package for the electronic structure of atoms, molecules and solids. *J. Phys. Condsens. Matter* **2018**, *30*, 195901.

(11) Zhang, S.; Krakauer, H. Quantum Monte Carlo Method using Phase-Free Random Walks with Slater Determinants. *Phys. Rev. Lett.* **2003**, *90*, 136401.

(12) Čížek, J. On the Correlation Problem in Atomic and Molecular Systems. Calculation of Wavefunction Components in UrsellType Expansion Using QuantumField Theoretical Methods. *J. Chem. Phys.* **1966**, *45*, 4256–4266.

(13) Møller, C.; Plesset, M. S. Note on an Approximation Treatment for Many-Electron Systems. *Phys. Rev.* **1934**, *46*, 618–622.

(14) Knowles, P. J.; Handy, N. C. A New Determinant-based Full Configuration Interaction Method. *Chem. Phys. Letters* **1984**, *111*, 315–321.

(15) Huron, B.; Malrieu, J. P.; Rancurel, P. Iterative perturbation calculations of ground and excited state energies from multiconfigurational zeroth-order wavefunctions. *J. Chem. Phys.* **1973**, *58*, 5745–5759.

(16) Giner, E.; Scemama, A.; Caffarel, M. Using perturbatively selected configuration interaction in quantum Monte Carlo calculations. *Can. J. Chem.* **2013**, *91*, 879.

(17) Schriber, J. B.; Evangelista, F. A. Communication: An adaptive configuration interaction approach for strongly correlated electrons with tunable accuracy. *J. Chem. Phys.* **2016**, *144*, 161106.

(18) Tubman, N. M.; Lee, J.; Takeshita, T. Y. et al. A deterministic alternative to the full configuration interaction quantum Monte Carlo method. *J. Chem. Phys.* **2016**, *145*, 044112.

(19) Holmes, A. A.; Tubman, N. M.; Umrigar, C. J. Heat-Bath Configuration Interaction: An Efficient Selected Configuration Interaction Algorithm Inspired by Heat-Bath Sampling. *J. Chem. Theory Comput.* **2016**, *12*, 3674–3680.

(20) Garniron, Y.; Scemama, A.; Loos, P.-F. et al. Hybrid stochastic-deterministic calculation of the second-order perturbative contribution of multireference perturbation theory. *J. Chem. Phys.* **2017**, *147*, 034101.

(21) Eriksen, J. J.; Lipparini, F.; Gauss, J. Virtual Orbital Many-Body Expansions: A Possible Route towards the Full Configuration Interaction Limit. *J. Phys. Chem. Lett.* **2017**, 4633–4639.

(22) Saebo, S.; Pulay, P. Local Treatment of Electron Correlation. *Annu. Rev. Phys. Chem.* **1993**, *44*, 213–236.

(23) Hampel, C.; Werner, H.-J. Local treatment of electron correlation in coupled cluster theory. *J. Chem. Phys.* **1996**, *104*, 6286–6297.

(24) Riplinger, C.; Neese, F. An efficient and near linear scaling pair natural orbital based local coupled cluster method. *J. Chem. Phys.* **2013**, *138*, 034106.

(25) Ziółkowski, M.; Jansík, B.; Kjaergaard, T. et al. Linear scaling coupled cluster method with correlation energy based error control. *J. Chem. Phys.* **2010**, *133*, 014107.

(26) Booth, G. H.; Thom, A. J. W.; Alavi, A. Fermion Monte Carlo without fixed nodes: A game of life, death, and annihilation in Slater determinant space. *J. Chem. Phys.* **2009**, *131*, 054106–1–10.

(27) White, S. R. Density matrix formulation for quantum renormalization groups. *Phys. Rev. Lett.* **1992**, *69*, 2863.

(28) Chan, G. K.-L. An algorithm for large scale density matrix renormalization group calculations. *J. Chem. Phys.* **2004**, *120*, 3172.

(29) Olivares-Amaya, R.; Hu, W.; Nakatani, N. et al. The ab-initio density matrix renormalization group in practice. *J. Chem. Phys.* **2015**, *142*, 034102.

(30) Thom, A. J. W. Stochastic Coupled Cluster Theory. *Phys. Rev. Lett.* **2010**, *105*, 263004–1–4.

(31) Spencer, J. S.; Thom, A. J. W. Developments in stochastic coupled cluster theory: The initiator approximation and application to the uniform electron gas. *J. Chem. Phys.* **2016**, *144*, 084108.

(32) Blunt, N. S.; Rogers, T. W.; Spencer, J. S. et al. Density-matrix quantum Monte Carlo method. *Phys. Rev. B* **2014**, *89*, 245124.

(33) Malone, F. D.; Blunt, N. S.; Shepherd, J. J. et al. Interaction picture density matrix quantum Monte Carlo. *J. Chem. Phys.* **2015**, *143*, 044116.

(34) Ten-no, S. Stochastic determination of effective Hamiltonian for the full configuration interaction solution of quasi-degenerate electronic states. *J. Chem. Phys.* **2013**, *138*, 164126.

(35) Ohtsuka, Y.; Ten-no, S. A study of potential energy curves from the model space quantum Monte Carlo method. *J. Chem. Phys.* **2015**, *143*, 214107.

(36) Ten-no, S. Multi-state effective Hamiltonian and size-consistency corrections in stochastic configuration interactions. *J. Chem. Phys.* **2017**, *147*, 244107.

(37) McClean, J. R.; Aspuru-Guzik, A. Clock quantum Monte Carlo technique: An imaginary-time method for real-time quantum dynamics. *Phys. Rev. A* **2015**, *91*.

(38) Nagy, A.; Savona, V. Driven-dissipative quantum Monte Carlo method for open quantum systems. *Phys. Rev. A* **2018**, *97*, 052129.

(39) Booth, G. H.; Chan, G. K.-L. Communication: Excited states, dynamic correlation functions and spectral properties from full configuration interaction quantum Monte Carlo. *J. Chem. Phys.* **2012**, *137*, 191102.

(40) Humeniuk, A.; Mitrić, R. Excited states from quantum Monte Carlo in the basis of Slater determinant. *J. Chem. Phys.* **2014**, *141*, 194104.

(41) Blunt, N. S.; Smart, S. D.; Booth, G. H. et al. An excited-state approach within full configuration interaction quantum Monte Carlo. *J. Chem. Phys.* **2015**, *143*, 134117.

(42) Spencer, J. S.; Blunt, N. S.; Foulkes, W. M. The sign problem and population dynamics in the full configuration interaction quantum Monte Carlo method. *J. Chem. Phys.* **2012**, *136*, 054110–1–10.

(43) Booth, G. H.; Smart, S. D.; Alavi, A. Linear-scaling and parallelisable algorithms for stochastic quantum chemistry. *Mol. Phys.* **2014**, *112*, 1855–1869.

(44) Petruzielo, F. R.; Holmes, A. A.; Changlani, H. J. et al. Semistochastic Projector Monte Carlo Method. *Phys. Rev. Lett.* **2012**, *109*, 230201.

(45) Overy, C.; Booth, G. H.; Blunt, N. S. et al. Unbiased reduced density matrices and electronic properties from full configuration interaction quantum Monte Carlo. *J. Chem. Phys.* **2014**, *141*, 244117.

(46) Blunt, N. S.; Smart, S. D.; Kersten, J. A. F. et al. Semi-stochastic full configuration interaction quantum Monte Carlo: Developments and application. *J. Chem. Phys.* **2015**, *142*, 184107.

(47) Holmes, A. A.; Changlani, H. J.; Umrigar, C. J. Efficient Heat-Bath Sampling in Fock Space. *J. Chem. Theory Comput.* **2016**, *12*, 1561–1571.

(48) Neufeld, V. A.; Thom, A. J. W. Exciting Determinants in Quantum Monte Carlo: Loading the Dice with Fast, Low-Memory Weights. *J. Chem. Theory Comput.* **2019**, *15*, 127–140.

(49) Cleland, D.; Booth, G. H.; Alavi, A. Communications: Survival of the fittest: Accelerating convergence in full configuration-interaction quantum Monte Carlo. *J. Chem. Phys.* **2010**, *132*, 41103.

(50) Spencer, J. S.; Neufeld, V. A.; Vigor, W. A. et al. Large scale parallelization in stochastic coupled cluster. *J. Chem. Phys.* **2018**, *149*, 204103.

(51) Scott, C. J. C.; Thom, A. J. W. Stochastic coupled cluster theory: Efficient sampling of the coupled cluster expansion. *J. Chem. Phys.* **2017**, *147*, 124105.

(52) The original algorithm used integer weights. It was subsequently shown that floating-point weights greatly reduce the stochastic noise. HANDE-QMC uses fixed-precision for the weights such

(53) Tubman, N. M.; Lee, J.; Takeshita, T. Y. et al. A deterministic alternative to the full configuration interaction quantum Monte Carlo method. *J. Chem. Phys.* **2016**, *145*, 044112.

(54) Malone, F. D.; Blunt, N.; Brown, E. W. et al. Accurate Exchange-Correlation Energies for the Warm Dense Electron Gas. *Phys. Rev. Lett.* **2016**, *117*, 115701.

(55) Blunt, N. S.; Booth, G. H.; Alavi, A. Density matrices in full configuration interaction quantum Monte Carlo: Excited states, transition dipole moments, and parallel distribution. *J. Chem. Phys.* **2017**, *146*, 244105.

(56) Shepherd, J. J.; Henderson, T. M.; Scuseria, G. E. Using full configuration interaction quantum Monte Carlo in a seniority zero space to investigate the correlation energy equivalence of pair coupled cluster doubles and doubly occupied configuration interaction. *J. Chem. Phys.* **2016**, *144*, 094112.

(57) The use of Fortran 2003 and 2008 imposes a need for a recent Fortran compiler. Indeed, we have found bugs in both open-source and proprietary compilers and worked around them where possible.

(58) Appleby, A. SMHasher. `https://github.com/aappleby/smhasher`.

(59) Saito, M.; Matsumoto, M. dSFMT. `https://github.com/MersenneTwister-Lab/dSFMT`.

(60) Moshier, S. R. cephes. `http://www.netlib.org/cephes/`.

(61) Wu, K.; Simon, H. TRLan. `https://codeforge.lbl.gov/projects/trlan`.

(62) Yamazaki, I.; Bai, Z.; Simon, H. et al. Adaptive Projection Subspace Dimension for the Thick-Restart Lanc-

zos Method. *ACM Trans. Math. Softw.* **2010**, *37*, 27.

(63) The HDF Group, Hierarchical Data Format, version 5. 1997-2018; http://www.hdfgroup.org/HDF5/.

(64) Ierusalimschy, R. *Programming in Lua*, 4th ed.; Lua.org, 2016.

(65) AOTUS: Advanced Options and Tables in Universal Scripting. `https://geb.sts.nt.uni-siegen.de/doxy/aotus/index.html`.

(66) JSON. `https://www.json.org/`, Accessed: 2018-10-31.

(67) CMake. `https://cmake.org/`, Accessed: 2018-10-31.

(68) Parrish, R. M.; Burns, L. A.; Smith, D. G. A. et al. Psi4 1.1: An Open-Source Electronic Structure Program Emphasizing Automation, Advanced Libraries, and Interoperability. *J. Chem. Theory Comput.* **2017**, *13*, 3185–3197.

(69) Toon Verstraelen, Pawel Tecmer, Farnaz Heidar-Zadeh, Cristina E. Gonzlez-Espinoza, Matthew Chan, Taewon D. Kim, Katharina Boguslawski, Stijn Fias, Steven Vandenbrande, Diego Berrocal, and Paul W. Ayers HORTON 2.1.0, http://theochem.github.com/horton/, 2017.

(70) Sun, Q.; Berkelbach, T. C.; Blunt, N. S. et al. PySCF: the Python-based simulations of chemistry framework. *Wiley Interdisciplinary Reviews: Computational Molecular Science 8*, e1340.

(71) Shao, Y.; Gan, Z.; Epifanovsky, E. et al. Advances in molecular quantum chemistry contained in the Q-Chem 4 program package. *Molecular Physics* **2015**, *113*, 184–215.

(72) Werner, H.-J.; Knowles, P. J.; Knizia, G. et al. *MOLPRO, version 2015.1, a package of ab initio programs*; 2015.

(73) Dunning, T. H., Jr. Gaussian basis sets for use in correlated molecular calculations. I. The atoms boron through neon and hydrogen. *J. Chem. Phys.* **1989**, *90*, 1007–1023.

(74) DMQMC averages over independent calculations and so does not suffer from a correlation issue.

(75) Flyvbjerg, H.; Petersen, H. G. Error estimates on averages of correlated data. *J. Chem. Phys.* **1989**, *91*, 461–466.

(76) `https://github.com/jsspencer/pyblock`.

(77) Kent, D. R.; Muller, R. P.; Anderson, A. G. et al. Efficient algorithm for on-the-fly error analysis of local or distributed serially correlated data. *Journal of Computational Chemistry* **2007**, *28*, 2309–2316.

(78) Leach, P.; Mealling, M.; Salz, R. *A universally unique identifier (UUID) URN namespace*; 2005.

(79) `https://hande.readthedocs.io`.

(80) Git is a very powerful distributed version control system and has become the *de facto* standard in software development. Its decentralized model has undoubtedly contributed to the huge growth of open-source software. The official documentation is available online `https://git-scm.com/`, but, as with any powerful tool, it is not an easy task to become familiar with it. A large number of tutorials are available online. We can recommend `http://gitimmersion.com/` and `https://coderefinery.github.io/git-intro/`.

(81) `https://github.com/hande-qmc/hande`.

(82) We note that access to the private repository is liberally granted.

(83) J. S. Spencer, testcode, https://github.com/jsspencer/testcode 2017.

(84) Buildbot. `https://buildbot.net/`, Accessed: 2018-11-1.

(85) Spencer, J. S.; Blunt, N. S.; Vigor, W. A. et al. Open-Source Development Experiences in Scientific Software: The HANDE Quantum Monte Carlo Project. *J. Open Res. Softw.* **2015**, *3*, 1–6.

(86) McKinney, W. *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*, 2nd ed.; O'Reilly Media, Incorporated, 2017.

(87) Vigor, W. A.; Spencer, J. S.; Bearpark, M. J. et al. Understanding and improving the efficiency of full configuration interaction quantum Monte Carlo. *J. Chem. Phys.* **2016**, *144*, 094110.

(88) Vigor, W. A.; Spencer, J. S.; Bearpark, M. J. et al. Minimising biases in full configuration interaction quantum Monte Carlo. *J. Chem. Phys.* **2015**, *142*, 104101.

(89) Oliphant, T. E. *Guide to NumPy*, 2nd ed.; CreateSpace Independent Publishing Platform: USA, 2015.

(90) Jones, E.; Oliphant, T.; Peterson, P. et al. SciPy: Open source scientific tools for Python. 2001; `http://www.scipy.org/`.

(91) Hunter, J. D. Matplotlib: A 2D graphics environment. *Computing In Science & Engineering* **2007**, *9*, 90–95.

(92) Rosen, L. *Open Source Licensing: Software Freedom and Intellectual Property Law*; Prentice Hall PTR: Upper Saddle River, NJ, USA, 2004.

(93) St. Laurent, A. M. *Understanding Open Source and Free Software Licensing*; O'Reilly Media, 2004.

(94) The full legal text of the license is available from the Free Software Foundation: `https://www.gnu.org/licenses/old-licenses/lgpl-2.1.en.html`.

(95) Legal text available from the Open Source Initiative: `https://opensource.org/licenses/BSD-3-Clause`.

(96) Knizia, G. IboView. `http://www.iboview.org`.

(97) MRCC, a quantum chemical program suite written by M. Kllay, Z. Rolik, J. Csontos, P. Nagy, G. Samu, D. Mester, J. Cska, B. Szab, I. Ladjnszki, L. Szegedy, B. Ladczki, K. Petrov, M. Farkas, P. D. Mezei, and B. Hgely. See also Z. Rolik, L. Szegedy, I. Ladjnszki, B. Ladczki, and M. Kllay, J. Chem. Phys. 139, 094105 (2013), as well as: `www.mrcc.hu`.

(98) Loos, P.-F.; Gill, P. M. W. The uniform electron gas. *WIREs Comput. Mol. Sci.* **2016**, *6*, 410–429.

(99) Giuliani, G.; Vignale, G. *Quantum Theory Electron Liq.*; Cambridge University Press: Cambridge, 2005; pp 1–68.

(100) Martin, R. M. *Electron. Struct.*; Cambridge University Press: Cambridge, 2004; pp 100–118.

(101) Gutzwiller, M. C. Effect of Correlation on the Ferromagnetism of Transition Metals. *Phys. Rev. Lett.* **1963**, *10*, 159–162.

(102) Hubbard, J. Electron Correlations in Narrow Energy Bands. *Proc. R. Soc. London A Math. Phys. Eng. Sci.* **1963**, *276*, 238–257.

(103) Kanamori, J. Electron Correlation and Ferromagnetism of Transition Metals. *Prog. Theor. Phys.* **1963**, *30*, 275–289.

(104) Altland, A.; Simons, B. *Condensed Matter Field Theory*; Cambridge University Press, 2010.

(105) Ceperley, D. M.; Alder, B. J. Ground State of the Electron Gas by a Stochastic Method. *Phys. Rev. Lett.* **1980**, *45*, 566–569.

(106) Perdew, J. P.; Zunger, A. Self-interaction correction to density-functional approximations for many-electron systems. *Phys. Rev. B* **1981**, *23*, 5048–5079.

(107) Giuliani, G.; Vignale, G. *Quantum Theory Electron Liq.*; Cambridge University Press: Cambridge, 2005; pp 327–404.

(108) Freeman, D. L. Coupled-cluster expansion applied to the electron gas: Inclusion of ring and exchange effects. *Phys. Rev. B* **1977**, *15*, 5512–5521.

(109) Bishop, R. F.; Lührmann, K. H. Electron correlations: I. Ground-state results in the high-density regime. *Phys. Rev. B* **1978**, *17*, 3757–3780.

(110) Bishop, R. F.; Lührmann, K. H. Electron correlations. II. Ground-state results at low and metallic densities. *Phys. Rev. B* **1982**, *26*, 5523–5557.

(111) Shepherd, J. J.; Grüneis, A.; Booth, G. H. et al. Convergence of many-body wavefunction expansions using a plane wave basis: from the homogeneous electron gas to the solid state. **2012**,

(112) Shepherd, J. J.; Grüneis, A. Many-body quantum chemistry for the electron gas: Convergent perturbative theories. *Phys. Rev. Lett.* **2013**, *110*, 226401.

(113) Roggero, A.; Mukherjee, A.; Pederiva, F. Quantum Monte Carlo with coupled-cluster wave functions. *Phys. Rev. B* **2013**, *88*, 115138.

(114) McClain, J.; Lischner, J.; Watson, T. et al. Spectral functions of the uniform electron gas via coupled-cluster theory and comparison to the G W and related approximations. *Phys. Rev. B* **2016**, *93*, 235139.

(115) Shepherd, J. J. Communication: Convergence of many-body wave-function expansions using a plane-wave basis in the thermodynamic limit. *J. Chem. Phys.* **2016**, *145*, 031104.

(116) Shepherd, J. J.; Booth, G.; Grüneis, A. et al. Full configuration interaction perspective on the homogeneous electron gas. *Phys. Rev. B Condens. Matter* **2012**, *85*.

(117) Shepherd, J. J.; Booth, G. H.; Alavi, A. Investigation of the full configuration interaction quantum Monte Carlo method using homogeneous electron gas models. *J. Chem. Phys.* **2012**, *136*, 244101.

(118) Neufeld, V. A.; Thom, A. J. W. A study of the dense uniform electron gas with high orders of coupled cluster. *J. Chem. Phys.* **2017**, *147*, 194105.

(119) Luo, H.; Alavi, A. Combining the Transcorrelated Method with Full Configuration Interaction Quantum Monte Carlo: Application to the Homogeneous Electron Gas. *J. Chem. Theory Comput.* **2018**, *14*, 1403–1411.

(120) Ruggeri, M.; Ríos, P. L.; Alavi, A. Correlation energies of the high-density spin-polarized electron gas to meV accuracy. *Phys. Rev. B* **2018**, *98*, 161105.

(121) Blunt, N. S. Communication: An efficient and accurate perturbative correction to initiator full configuration interaction quantum Monte Carlo. *J. Chem. Phys.* **2018**, *148*, 221101.

(122) Brown, E. W.; Clark, B. K.; DuBois, J. L. et al. Path-Integral Monte Carlo Simulation of the Warm Dense Homogeneous Electron Gas. *Phys. Rev. Lett.* **2013**, *110*, 146405.

(123) Schoof, T.; Groth, S.; Vorberger, J. et al. *Ab Initio* Thermodynamic Results for the Degenerate Electron Gas at Finite Temperature. *Phys. Rev. Lett.* **2015**, *115*, 130402.

(124) Groth, S.; Schoof, T.; Dornheim, T. et al. Ab initio quantum Monte Carlo simulations of the uniform electron gas without fixed nodes. *Phys. Rev. B* **2016**, *93*, 085102.

(125) Dornheim, T.; Groth, S.; Schoof, T. et al. Ab initio quantum Monte Carlo simulations of the uniform electron gas without fixed nodes: The unpolarized case. *Phys. Rev. B* **2016**, *93*, 205134.

(126) Dornheim, T.; Groth, S.; Malone, F. D. et al. Ab initio quantum Monte Carlo simulation of the warm dense electron gas. *Phys. Plasmas* **2017**, *24*, 056303.

(127) Brown, E. W.; DuBois, J. L.; Holzmann, M. et al. Exchange-correlation Energy for the Three-Dimensional Homogeneous Electron Gas at Arbitrary Temperature. *Phys. Rev. B* **2013**, *88*, 081102.

(128) Karasiev, V. V.; Sjostrom, T.; Dufty, J. et al. Accurate Homogeneous Electron Gas Exchange-Correlation Free Energy for Local Spin-Density Calculations. *Phys. Rev. Lett.* **2014**, *112*, 076403.

(129) Dornheim, T.; Groth, S.; Sjostrom, T. et al. Ab Initio Quantum Monte Carlo Simulation of the Warm Dense Electron Gas in the Thermodynamic Limit. *Phys. Rev. Lett.* **2016**, *117*, 156403.

(130) Groth, S.; Dornheim, T.; Sjostrom, T. et al. Ab initio Exchange-Correlation Free Energy of the Uniform Electron Gas at Warm Dense Matter Conditions. *Phys. Rev. Lett.* **2017**, *119*, 135001.

(131) Dornheim, T.; Groth, S.; Bonitz, M. The uniform electron gas at warm dense matter conditions. *Physics Reports* **2018**, *744*, 1.

(132) Fortney, J. J.; Glenzer, S. H.; Koenig, M. et al. Frontiers of the physics of dense plasmas and planetary interiors: Experiments, theory, and applications. *Phys. Plasmas* **2009**, *16*, 041003.

(133) Hu, S. X.; Militzer, B.; Goncharov, V. N. et al. First-principles equation-of-state table of deuterium for inertial confinement fusion applications. *Phys. Rev. B* **2011**, *84*, 224109.

(134) VandeVondele, J.; Krack, M.; Mohamed, F. et al. Quickstep: Fast and accurate density functional calculations using a mixed Gaussian and plane waves approach. *Comput. Phys. Commun.* **2005**, *167*, 103–128.

(135) Hutter, J.; Iannuzzi, M.; Schiffmann, F. et al. CP2K: atomistic simulations of condensed matter systems. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **2014**, *4*, 15–25.

(136) Goedecker, S.; Teter, M.; Hutter, J. Separable dual-space Gaussian pseudopotentials. *Phys. Rev. B* **1996**, *54*, 1703–1710.

(137) Hartwigsen, C.; Goedecker, S.; Hutter, J. Relativistic separable dual-space Gaussian pseudopotentials from H to Rn. *Phys. Rev. B* **1998**, *58*, 3641–3662.

(138) Sun, Q.; Berkelbach, T. C.; McClain, J. D. et al. Gaussian and plane-wave mixed density fitting for periodic systems. *J. Chem. Phys.* **2017**, *147*, 164119.

(139) Vosko, S. H.; Wilk, L.; Nusair, M. Accurate spin-dependent electron liquid correlation energies for local spin density calculations: a critical analysis. *Can. J. Phys.* **1980**, *58*, 1200–1211.

(140) Booth, G. H.; Grüneis, A.; Kresse, G. et al. Towards an exact description of electronic wavefunctions in real solids. *Nature* **2013**, *493*, 365–370.

(141) Raghavachari, K.; Trucks, G. W.; Pople, J. A. et al. A fifth-order perturbation comparison of electron correlation theories. *Chem. Phys. Lett.* **1989**, *157*, 479–483.

(142) Blöchl, P. E. Projector augmented-wave method. *Phys. Rev. B* **1994**, *50*, 17953–17979.

(143) McClain, J.; Sun, Q.; Chan, G. K.-L. et al. Gaussian-Based Coupled-Cluster Theory for the Ground-State and Band Structure of Solids. *J. Chem. Theory Comput.* **2017**, *13*, 1209–1218.

(144) Gruber, T.; Liao, K.; Tsatsoulis, T. et al. Applying the Coupled-Cluster Ansatz to Solids and Surfaces in the Thermodynamic Limit. *Phys. Rev. X* **2018**, *8*, 021043.

(145) Grüneis, A.; Booth, G. H.; Marsman, M. et al. Natural Orbitals for Wave Function Based Correlated Calculations Using a Plane Wave Basis Set. *J. Chem. Theory Comput.* **2011**, *7*, 2780–2785.

(146) Wagner, L. K.; Ceperley, D. M. Discovering correlated fermions using quantum Monte Carlo. *Reports Prog. Phys.* **2016**, *79*, 094501.

(147) We note we probably would have chosen, like Psi4, to use Python via the excellent pybind11 library had HANDE-QMC been written in C++, in part due to Python already being used extensively in scientific research.

(148) Lu, J.; Wang, Z. The full configuration interaction quantum Monte Carlo method in the lens of inexact power iteration. *arXiv:1711.09153 [physics]* **2017**, arXiv: 1711.09153.

(149) Deustua, J. E.; Shen, J.; Piecuch, P. Converging High-Level Coupled-Cluster Energetics by Monte Carlo Sampling and Moment Expansions. *Phys. Rev. Lett.* **2017**, *119*, 223003.

(150) Gillan, M. J.; Towler, M. D.; Alfe, D. Petascale computing opens new vistas for quantum Monte Carlo. *Psi-k Newsletter* **2011**, *103*, 32.

Graphical TOC Entry