StochasticGW-GPU: rapid quasi-particle energies for molecules beyond 10000 atoms

Phillip S. Thomas[§], Minh Nguyen, Dimitri Bazile, Tucker Allen, Barry Y. Li, Wenfei Li, Daniel Neuhauser, Mauro Del Ben, and Jack Deslippe[*]

E-mail: pthomas@lbl.gov

Abstract

`StochasticGW` is a code for computing accurate Quasi-Particle (QP) energies of molecules and material systems in the GW approximation. `StochasticGW` utilizes the stochastic Resolution of the Identity (sROI) technique to enable a massively-parallel implementation with computational costs that scale semi-linearly with system size, allowing the method to access systems with tens of thousands of electrons. We introduce a new implementation, $\texttt{StochasticGW}-\texttt{GPU}$, for which the main bottleneck steps have been ported to GPUs and which gives substantial performance improvements over previous versions of the code. We showcase the new code by computing band gaps of hydrogenated silicon clusters ($\mathrm{Si_xH_y}$) containing up to 10001 atoms and 35144 electrons, and we obtain individual QP energies with a statistical precision of better than ±0.03 eV with times-to-solution on the order of minutes.

Introduction

In recent years, predicting electronic properties of materials from first-principles has become a key step in the materials design process, greatly reducing laboratory time and costs by directing synthetic efforts towards the most promising material candidates for a given application.

Properties of interest, including band gaps, ionization potentials, and optical spectra, can be computed via electronic structure methods implemented in commercially-available and open source software. For excited states, post-Hartree-Fock methods, including multi-reference configuration interaction[1,2] and equation-of-motion coupled-cluster methods[3,4], while being gold standards for accuracy, are only applicable to small molecules since the computational cost of these methods grows steeply with the number of electrons. Due to their more favorable scaling, Density Functional Theory (DFT)-based methods[5] have become the industry standards for predicting ground state energies of large molecules and materials[6]; however, their accuracy is poor when used to predict Quasi-Particle (QP) energies corresponding to excited states[7–9]. Excited-state methods that can be applied on top of a DFT starting point, such as Time-Dependent (TD-)DFT[10], the GW approximation[9,11,12], including its extensions using perturbation theory[13] and the Bethe-Salpeter Equation (BSE) approach[14], provide superior accuracy compared to DFT, but they are expensive to apply, limiting excited state calculations to systems containing ∼10,000 electrons[15–20].

The GW method has emerged as a robust and routinely-used tool for computing QP energies of material systems[11,21,22], and GW implementations are now found in many quantum chemistry/materials software packages[23–36]. Here, one approximates the self-energy operator, Σ, which embodies the many-body electron exchange and correlation effects, as the product of the single-particle Green's function, G, and the screened Coulomb interaction, W, i.e. $\Sigma = iGW$. In common practice, one initiates a GW calculation by first solving the Kohn-Sham equation using a DFT method of choice to generate the starting orbitals and energies,

$$\left[-\frac{1}{2}\nabla^2 + V_{ion} + V_H + V_{XC}^{KS}\right]\phi_k^{KS} \quad = \quad \varepsilon_k^{KS}\phi_k^{KS} \tag{1}$$

where V_{ion}, V_H, and V_{XC} are the ionic, Hartree, and exchange correlation potential terms, respectively, and ϕ_k^{KS} and ε_k^{KS} are the k-th orbital and energy eigenpair. To obtain QP wavefunctions and energies, one starts by setting $\phi_k^{QP} = \phi_k^{KS}$ and $\varepsilon_k^{QP} = \varepsilon_k^{KS}$ and then solves the analogous Dyson equation,

$$\left[-\frac{1}{2}\nabla^2 + V_{ion} + V_H + \Sigma\left(\varepsilon_k^{QP}\right)\right]\phi_k^{QP} \;=\; \varepsilon_k^{QP}\phi_k^{QP} \qquad (2)$$

for ϕ_k^{QP} and ε_k^{QP} to self-consistency[37–39]. For many practical applications, it is sufficient to solve the equation in a single pass, possibly from a pre-optimized starting point[40]. This is referred to as the G_0W_0 approximation, and this is what we use throughout the manuscript with the zero subscript omitted for clarity.

Evaluating the self-energy operator is costly and can be tackled by one of two strategies, broadly defined as “deterministic” and “stochastic”. In deterministic GW, the overall cost is dominated by computing the inverse dielectric ϵ^{-1} and Σ operator matrix elements, requiring one to evaluate many integrals and summations over valence-conduction orbital pairs; this formally scales as $\mathcal{O}\left(N_e^4\right)$ for an N_e-electron molecule or periodic system. Considerable efforts have been directed towards improving this scaling: one can achieve $\mathcal{O}\left(N_e^3 \log N_e\right)$ complexity by employing, for example, the space-time formulation and using the Fast Fourier Transform (FFT) to transform to-and-from the real space[41–44]. Interpolative density fitting[45] also achieves cubic complexity, potentially with smaller prefactors than the real space-time methods. The stochastic pseudobands approach[46] can be used to reduce the size of the valence space needed to converge matrix elements of Σ even further, decreasing the overall scaling to $\mathcal{O}\left(N_e^{2.4}\right)$.

The developments described above have spurred increasing interest in performing large-scale GW calculations[16,19,20], and several massively-parallel deterministic GW implementations have been benchmarked. Zhang *et al* recently demonstrated a portable GPU implementation in the BerkeleyGW code capable of scaling efficiently to entire exascale architectures, achieving excellent time to solution (on the order of minutes) for the computation of quasiparticle (QP) energies in semiconductor systems containing up to 17574 atoms in the simulation cell[20]. Yu and Govoni computed states of an interface model of Si and Si_3N_4 with up to 2376 atoms and 10368 electrons on 10368 V100 GPUs in ∼578 minutes (summed total of `wstat` and `wfreq` steps) using the GPU-enabled `WEST` code[18]. Wu *et al* reported calculations on 13824-atom, 13824-electron LiH supercells on 449280 SW26010Pro cores in 285 s using a massively-parallel version of `PWDFT`[19]. Very recently, Vetsch *et al* performed non-equilibrium Green's function calculations on hydrogen-passivated silicon nanoribbons with up to 42240 atoms on 37600 MI250X GPUs in 42 s per iteration using a novel self-consistent GW algorithm with domain decomposition, implemented in their QuaTrEx code[47].

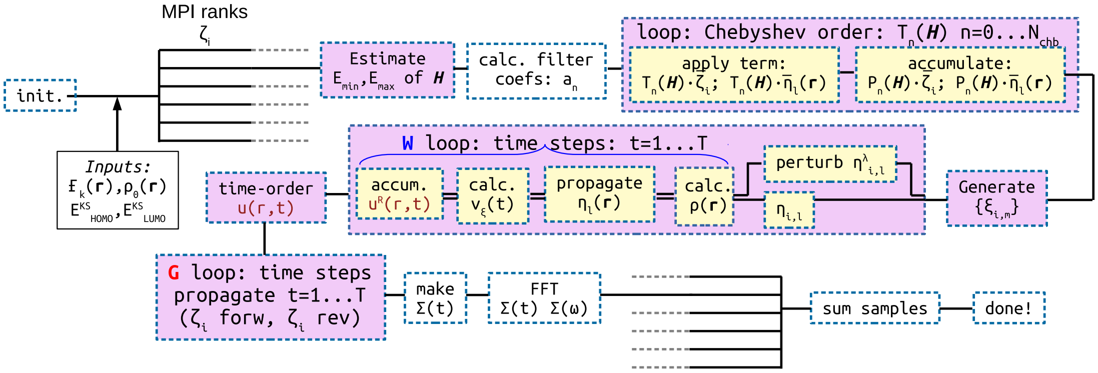
For systems containing thousands of atoms or more, one can evaluate the self-energy operator using a stochastic GW formulation at greatly reduced cost, as detailed by our previous works[48–51]. Here, we briefly summarize the main features of the method. First, we evaluate the self-energy operator in the time domain to exploit the direct product computation of $\mathit{\Sigma}(t)$ from Green's function G and screened Coulomb potential W; we Fourier transform the resulting $\Sigma(t)$ to $\Sigma(\omega)$ only in the final stage of the calculation. Second, we invoke the stochastic Resolution of Identity (sRoI)[52] and define random orbital functions to use as bases for evaluating the Green's function G and effective polarization W. We then compute the expectation values of these operators using real time propagation and accumulate statistical averages over products of random samples. This is the main ingredient of stochastic GW and it has the advantage of allowing one to decouple the spatial- and time- dependence in the six-dimensional integrals needed to evaluate $\Sigma(t)$[48]. As a result, instead of requiring the full space of occupied and unoccupied orbitals and energies $\{\phi_k, \varepsilon_k\}$ (which typically number in the tens of thousands or for a thousand-atom molecule), we, in effect, evaluate G and W using compact sets of stochastic linear combinations of the occupied or unoccupied orbital space. An additional benefit of sROI is that computations in the stochastic bases can be done independently, enabling the critical path of the calculation to be made embarrassingly parallelizable. Third, we incorporate sparse stochastic compression[50] in our stochastic Time-Dependent Hartree propagation algorithm[48,53] for evaluating W. This enables computing components of W over a collection of randomly-chosen short segments without needing a full spatial grid, reducing storage costs. Finally, instead of projecting each stochastic sample onto the full set of occupied orbitals $\left\{\phi_k^{KS}\right\}_{occ}$ from the preliminary DFT calculation (incurring substantial I/O and computational costs), we filter these samples to generate occupied stochastic orbitals. We construct a filter from a Chebyshev polynomial expansion of the Kohn-Sham Hamiltonian[54]. While the filtering approach has the disadvantage of requiring many terms to produce a sharp cutoff at the Fermi energy, we recently found that the expansion length of the filter can be greatly decreased[51] by relaxing the expansion to have zero weights inside the band gap (where no states are present); this, in turn, reduces the number of matrix-vector products needed to prepare the occupied stochastic orbitals.

The above framework enables a near-linear $\mathcal{O}\left(N_e \log N_e\right)$ scaling stochastic GW algorithm, with costs dominated by performing FFTs on the spatial grid. While development of stochastic algorithms for computing electronic properties has lagged behind that of deterministic ones[20], for QP *energies* stochastic GW is well-suited for handling large systems at a much reduced computational cost compared to deterministic GW. Some large-scale stochastic calculations have been performed: Vlcek *et al*[50] computed HOMO-LUMO gaps of Γ-point Diamond and Silicon supercells containing up to 2744 atoms and 10976 electrons in under 2000 core hours on an HPC cluster containing 144 nodes with 1728 Intel Xeon E5-2680v3@2.5 GHz processors. More recently, Brooks *et al*[17] used `StochasticGW` to compute twist-induced localized Moire states of bi-layer phosphorene sheets containing up to 2708 atoms and 13540 electrons. Both of these calculations were performed with a CPU-only version of the code without gapped-filtering. In this paper, we assimilate the ideas described above in a new GPU-accelerated version of `StochasticGW` which we showcase by computing QP energies of clusters containing upwards of 10001 atoms and 35144 electrons on $\sim$ 1000 GPUs with times-to-solution of minutes.

Implementation Details

Algorithmic Overview

Our stochastic GW implementation is similar to that described previously[50] and with the inclusion of the gapped filtering[51] technique. Here, we only summarize the key components of the algorithm; see the earlier works for a more detailed explanation of the methodology. A block diagram, shown in Figure 1, depicts the major portions of the code.



MPI ranks
ζ_i
init.
Inputs:
$\phi_k(\mathbf{r}),\rho_0(\mathbf{r})$
$E^{KS}_{HOMO},E^{KS}_{LUMO}$
Estimate
E_{min},E_{max} of H
calc. filter
coefs: a_n
loop: Chebyshev order: $T_n(H)$ n=0...N_{chb}
apply term:
$T_n(H)\cdot\bar{\zeta}_i$; $T_n(H)\cdot\bar{\eta}_l(\mathbf{r})$
accumulate:
$P_n(H)\cdot\bar{\zeta}_i$; $P_n(H)\cdot\bar{\eta}_l(\mathbf{r})$
W loop: time steps: t=1...T
accum.
$u^R(r,t)$
calc.
$v_\xi(t)$
propagate
$\eta_l(\mathbf{r})$
calc.
$\rho(\mathbf{r})$
perturb $\eta^\lambda_{i,l}$
$\eta_{i,l}$
Generate
$\{\xi_{i,m}\}$
time-order
$u(r,t)$
G loop: time steps
propagate t=1...T
(ζ_i forw, ζ_i rev)
make
$\Sigma(t)$
FFT
$\Sigma(t)$ $\Sigma(\omega)$
sum samples
done!

Figure 1: Block diagram of the main steps of the StochasticGW algorithm. Each MPI rank performs the same operations on a different set of data (see text for details). Steps enclosed in shaded boxes have been ported to GPUs.

The `StochasticGW` code requires as inputs: 1) coordinates of the atoms, 2) a pseudopotential for each atomic type, 3) the ground state density $\rho_0(\mathbf{r})$, 4) estimates of the energies of the highest occupied and lowest unoccupied molecular orbitals (E^{KS}_{HOMO} and E^{KS}_{LUMO}, respectively), and 5) a spatial orbital ϕ_k for which the quasiparticle energy ε^{QP}_k is desired. Items 1) - 3) are needed to construct the Kohn-Sham Hamiltonian internally in `StochasticGW`; item 4) defines the cutoff region of the gapped filter. We obtain items 3) - 5) from a preliminary DFT calculation.

We begin by constructing the spectral filter to apply the Heaviside operator. The Heaviside operator (Θ) expanded in Chebyshev polynomials T_n in the Hamiltonian $\hat{H}$, which can be written as:

$$\Theta\left(\mu-\hat{H}\right)\simeq\sum_{n=0}^{N_{chb}}a_nT_n\left(\hat{H}\right) \tag{3}$$

where μ is the chemical potential, a_n are expansion coefficients, and N_{chb} is the maximum degree of Chebyshev polynomial needed to build the filter. One requires the upper and lower spectral bounds of the Kohn-Sham Hamiltonian $\hat{H}^{KS}$ to shift-and-scale the eigenvalue spectrum into the interval $[-1,1]$; there are various schemes to obtain these bounds but we find that one of the simplest, a shifted power iteration, works well for this purpose. In the gapped-filtering method, rather than applying the cutoff of filter Θ at a specific value of μ we instead apply it over the band gap containing μ, i.e. over $E_{HOMO}^{KS}\leq\mu\leq E_{LUMO}^{KS}$, so we must also map the energies E_{HOMO}^{KS} and E_{LUMO}^{KS} into $[-1,1]$. With these we compute the N_{chb} filter coefficients a_n as described previously[51].

Next, we generate N_ζ random "white-noise" start orbitals, $\left|\bar{\zeta}_i\left(\mathbf{r},t=0\right)\right\rangle$, for the stochastic realization of G; we dub these the Monte Carlo (MC) samples. For each, we also generate N_η additional white-noise orbitals, $\left|\bar{\eta}_{i,\ell}\left(\mathbf{r},t=0\right)\right\rangle$, needed to calculate the action of the time-dependent effective interaction operator $W\left(t\right)$ on a vector related to each $\left|\bar{\zeta}_i\right\rangle$. We apply the Heaviside filter to both sets of $\left\{\bar{\zeta}_i\right\}$ and $\left\{\bar{\eta}_{i,\ell}\right\}$ orbitals in order to project them onto random linear combinations of the occupied orbital subspace $\left\{\phi_k^{KS}\right\}_{occ}$. We denote these "filtered" orbitals as $\left\{\zeta_i\right\}$ and $\left\{\eta_{i,\ell}\right\}$ (without the overhead bar).

Subsequently, we evaluate the diagonal time-dependent self-energy matrix element, $\left\langle\phi_k\left|\Sigma\left(t\right)\right|\phi_k\right\rangle$, in two phases. In the first, we use linear-response time-dependent Hartree[48] to compute the action of the retarded polarization interaction W^R on the occupied states. Algorithmically, we create a perturbed copy of each $\eta_{i,\ell}$ (denoted $\eta_{i,\ell}^{\lambda}$), and propagate both perturbed and unperturbed copies in time under the action of the time-dependent Hamiltonian. We accumulate the causal response function, $u^R\left(\mathbf{r},t\right)$, as the difference of the perturbed and unperturbed time-dependent potentials, and we time-order[55] the accumulated $u^R\left(\mathbf{r},t\right)$ to produce the effective polarization potential $u\left(\mathbf{r},t\right)$. Note that instead of accumulating $u^R\left(\mathbf{r},t\right)$ in the basis of $\left\{\eta_{i,\ell}\right\}$ (which requires keeping a copy of $u^R\left(\mathbf{r},t\right)$ on the full spatial grid for each $\eta_{i,\ell}$), we reduce the storage requirement by projecting $u^R(\mathbf{r},t)$ onto a set of N_ξ randomly chosen functions $|\xi_{i,m}\rangle$ on short, fragmented segments of the spatial grid and accumulate $u^R(\mathbf{r},t)$ in the $\{\xi_{i,m}\}$ basis.

In the second phase, we evaluate the action of the Green's function iG on each $\bar{\zeta}_i$. Numerically, we compute $\zeta_i(t)$ by propagating the filtered ζ_i (representing a random linear combination of occupied states) backwards in time while simultaneously propagating its orthogonal complement (representing a random linear combination of unoccupied states) forwards in time. This represents the stochastic realization of Green's function as a time correlation function,

$$iG(\mathbf{r},\mathbf{r}',t) = \frac{1}{N_\zeta}\sum_\zeta \zeta(\mathbf{r},t)\,\bar{\zeta}(\mathbf{r}'). \tag{4}$$

Having obtained $u(\mathbf{r},t)$ and $\zeta(\mathbf{r},t)$ from the first and second phases, respectively, the diagonal self-energy matrix element for orbital ϕ_k becomes

$$\begin{aligned}\langle\phi_k|\Sigma(t)|\phi_k\rangle &= \int\int \phi_k(\mathbf{r})\,iG(\mathbf{r},\mathbf{r}',t)\,W(\mathbf{r},\mathbf{r}',t)\,\phi_k(\mathbf{r}')\,d\mathbf{r}d\mathbf{r}' \\ &= \frac{1}{N_\zeta}\sum_\zeta\int\int \phi_k(\mathbf{r})\,\zeta(\mathbf{r},t)\,W(\mathbf{r},\mathbf{r}',t)\,\bar{\zeta}(\mathbf{r}')\,d\mathbf{r}d\mathbf{r}' \\ &= \frac{1}{N_\zeta}\sum_\zeta\int \phi_k(\mathbf{r})\,\zeta(\mathbf{r},t)\,u(\mathbf{r},t)\,d\mathbf{r}.\end{aligned} \tag{5}$$

We then compute the the frequency-resolved self-energy matrix element $\langle\phi_k|\Sigma(\omega)|\phi_k\rangle$ from the time-dependent form via discrete Fourier transform, and we obtain quasi-particle energy, ε_k^{QP}, by solving

$$\varepsilon_k^{QP} = \varepsilon_k^{KS} + \left\langle\phi_k\middle|X+\Sigma\left(\omega=\varepsilon_k^{QP}\right)-V_{XC}\middle|\phi_k\right\rangle \tag{6}$$

where X is the sROI realization of the Fock exchange operator in the basis of $\{\eta_{i,\ell}\}$ and all other quantities have been previously defined.

Scaling of the method

A key aim of our stochastic GW formulation is to achieve computational scaling that grows slowly, ideally linearly, with respect to system size. The most numerically intensive portions of the algorithm apply matrix-vector products repeatedly to the set of $\{\eta_{i,\ell}\}$ during the filtering and propagation phases. Here, one applies either the Kohn-Sham Hamiltonian, $\hat{H}^{KS}$, or the evolution operator, $e^{-i\hat{H}(t)\Delta t}$, to a set of vectors with each having length $N_g = N_x N_y N_z$. We apply matrix-vector products in a Fourier grid representation whereby FFT pairs switch between position and momentum representations where the potential energy and kinetic energy operators are diagonal, respectively. Applying the individual kinetic and potential energy operators scales as $\mathcal{O}\left(N_g\right)$, but the cost of each Hamiltonian/evolution operation is dominated by the $\mathcal{O}\left(N_g \log_2 N_g\right)$ FFT cost. Thus, to filter the $N_\zeta N_\eta$ starting orbitals, one applies a length N_{chb} filter at a cost of $\mathcal{O}\left(N_\zeta N_\eta N_{chb} N_g \log_2 N_g\right)$ operations. Likewise, propagating the full set of $\{\eta_{i,\ell}\}$ for N_τ time steps has a cost scaling as $\mathcal{O}\left(N_\zeta N_\eta N_\tau N_g \log_2 N_g\right)$. Accumulating $u^R\left(\mathbf{r},t\right)$ costs $\mathcal{O}\left(N_\zeta N_\xi N_\tau N_g f_g\right)$, where f_g is the fractional length of each of the fragmented stochastic functions $\{\xi_{i,m}\}$ relative to the full spatial grid length, N_g.

For tackling quasi-particle energies of large molecules it is important to consider the dependence of each parameter on system size. The number of MC samples, N_ζ, and the number of occupied stochastic orbitals, N_η determine the statistical accuracy of the QP energies and do not increase with system size (N_η actually decreases with increasing system size due to self-averaging). The number of time steps, N_τ, determines the energy resolution of $\Sigma\left(\omega\right)$ and is also independent of system size. The number of grid points, N_g, while cubic in dimension ($N_g = N_x N_y N_z$), grows linearly overall with system size due to spatial packing of atoms in 3-dimensional space. For accumulating $u^R\left(\mathbf{r},t\right)$, the statistical error does increase with the ratio $\frac{N_g}{N_\xi}$. This means that one must simultaneously increase the number of stochastic segments N_ξ as the grid size N_g increases to prevent growth of errors. However, in the sparse stochastic basis, the cost increase from requiring larger N_ξ can be offset by decreasing the fractional length f_g of each segment $\{\xi_{i,m}\}$ (i.e. by using more ξ vectors but making them "sparser")[50]. Finally, the number of Chebyshev coefficients, N_{chb}, needed to fit the gapped filter, depends on the width of the band gap relative to the spectral width of the Kohn-Sham Hamiltonian. The value of N_{chb} needed to accurately fit the filter does increase with system size due to larger spectral width of $\hat{H}^{KS}$, but this can be mitigated by applying a kinetic energy cutoff. In summary, as long as care is taken to manage the growth of the N_ξ and N_{chb} parameters accordingly, one can achieve near-linear scaling with system size in stochastic GW calculations.

GPU implementation

The original `StochasticGW` code (through v.2.0) is written in Fortran 90 and parallelized using Message Passing Interface (MPI). A key feature of `StochasticGW` is that the N_ζ Monte Carlo samples can be processed independently of one another, resulting in embarrassing parallelism over large portions of the algorithm. Additionally, the code contains an option to extend the MPI-level parallelism over the N_η occupied stochastic functions at the cost of an additional call to mpi_allreduce() at each time step (needed to compute the time-dependent density $\rho\left(\mathbf{r},t\right)$). In the original implementation, operations over grid points are performed in serial. For systems containing thousands of atoms or more, the grids are large enough that these operations become significant serial bottlenecks, motivating us to develop a GPU port to handle them in parallel.

In the GPU implementation, we retain the idea of processing each of the N_ζ MC samples with a separate MPI rank, but the N_η occupied stochastic functions per sample reside on the same MPI rank so that MPI calls are not needed at each time step to evaluate the time-dependent density $\rho\left(\mathbf{r},t\right)$. The GPU code utilizes kernels written using OpenACC directives and calls to specialized libraries (cuRAND and cuFFT) when needed. To maximize efficency, attention must be given to minimize the amount of data transferred between the host CPU and each GPU and to organize the computational workload to expose as much parallelism to the GPU as possible. To this end, we performed several optimizations.

First, we structured the arrays containing the stochastic orbitals with multi-indices so that each kernel can process the orbitals in single-instruction-multiple-data (SIMD) fashion. Many of the operations in the filtering and propagation cycles, such as applying the kinetic and potential energy operators, are simple element-wise array multiplications which are highly vectorizable on GPU hardware. In each case which follows, we construct the multi-index arrays and then offload them once onto a single GPU, retrieving the result only after the full set of filtering or propagation iterations. For the filtering cycle, this means that on each MPI rank we pack the $\bar{\zeta}_i$ associated with rank i along with its set of $\{\bar{\eta}_{i,\ell}\}\,; \ell = 1 \ldots N_\eta$ orbitals into an array of size $(N_g \times N_\eta + 1)$. For the propagation cycle involving the set of $\{\eta_{i,\ell}\}$, the perturbed and unperturbed copies can be processed in parallel, so we pack both copies into an array of size $(N_g \times N_\eta \times 2)$. The MC sample ζ_i cannot be propagated in parallel with the $\{\eta_{i,\ell}\}$ here since the former depends on the effective polarization potential $u\left(\mathbf{r}, t\right)$ resulting from the $\{\eta_{i,\ell}\}$ propagation cycle. However, since reverse-time propagation of ζ_i and forward-time propagation of its orthogonal complement are operationally identical (other than a difference in sign), we can pack these functions into an array of size $(N_g \times 2)$ and achieve a parallel performance boost for propagation of ζ_i as well.

Not all operations in the filtering and propagation steps are trivial to vectorize. Normalizations appear periodically in each of the filtering, propagation, and spectral estimation stages; each requires summing over values defined over N_g grid points. For instance, for normalizations performed in the $\{\eta_{i,\ell}\}$ propagation cycle, at most only $2N_\eta$ operations can be performed in parallel instead of $2N_\eta N_g$. For the largest systems in this work, $N_\eta \sim 8$ while $N_g \sim 1.6 \times 10^8$, meaning that the benefits of having many parallel threads are largely lost in each normalization kernel call. To solve this, we divided the N_g grid points into short segments of length L. This allows us to parallelize sums over grid points over $\frac{N_g}{L}$ threads at the cost of having to perform an atomic add by each thread after the sum over each segment has been accumulated. The optimal value of L is hardware dependent; on NVIDIA A100 GPUs we achieved the best performance with $L \sim 256$. In this manner, we achieve an overall parallelism of up to $2N_\eta \frac{N_g}{L}$ threads in normalization calls.

Second, the main computations needed to accumulate $u(\mathbf{r}, t)$ have also been ported to the GPU. We generate the $\{\xi_{i,m}\}$ basis via calls to the cuRAND library, and we compute the overlaps $\left\langle \xi \mid u^R(t) \right\rangle$ on-the-fly during the $\{\eta_{i,\ell}\}$ propagation phase in a segmented fashion similar to the one described above for normalization. Here, we multiply two arrays of sizes $(N_\xi \times N_g f_g)$ and $(N_g f_g \times 2)$ parallelized over $2N_\xi \frac{N_g f_g}{L}$ threads, where each ξ function is processed in segments of length $L = 32$, and the factor of 2 again arises from performing the unperturbed-η and perturbed-η propagations in parallel. Finally, we perform the time-ordering operation to convert the resulting $u^R(\mathbf{r}, t)$ to $u(\mathbf{r}, t)$ by calling the cuFFT library before and after an OpenACC kernel used for performing the complex conjugation step.

Utilities

The newest (3.0) version of StochasticGW is freely available[56] on GitHub and includes several utilities to aid researchers in preparing inputs for the code:

The dft2sgw utility reads and preprocesses results from a preliminary DFT calculation. This utility requires a DFT output file and a set of .cube files as input; dft2sgw prepares an input file, sgwinp.txt, containing atomic coordinates, HOMO and LUMO energies (for gapped filtering), the spatial charge density, and a requested set of orbitals for the system of interest. dft2sgw also has a functionality, enabled via the FFTW[57] library, to up- or down-sample the orbital/density spatial grid from the preliminary DFT calculation in case a different grid for the stochastic GW step is desired. The utility currently supports Quantum ESPRESSO[25,58], the Real-Space Multigrid (RMG)-[59,60] DFT code, and CP2K[35] (but note that StochasticGW does not yet include pseudopotential support for CP2K).

StochasticGW also includes two utilities, plotfilter.py and plotorbital.py, which use the `MatPlotLib`[61] python package to generate plots related to stochastic GW calculations:

The `plotfilter.py` utility plots the filter and depicts the log of the magnitudes of the filter coefficients and is useful for checking the quality of the Chebyshev expansion of the filter.

The `plotorbital.py` utility visualizes the atomic coordinates, spatial orbitals, and charge density contained in the file sgwinp.txt; this feature allows one to quickly select orbitals of interest for a subsequent `StochasticGW` calculation.

Numerical Experiments

We now test our implementation of `StochasticGW` by computing QP energies of a series of non-periodic nanoclusters, $Si_{293}H_{172}$, $Si_{705}H_{300}$, $Si_{5031}H_{1172}$, $Si_{7745}H_{1572}$, $Si_{8381}H_{1620}$. We constructed each cluster from a uniformly expanded silicon superlattice of size $15 \times 15 \times 15$ using the experimental unit cell parameter for silicon ($a = b = c = 10.26$ Bohr) corresponding to the diamond cubic structure with an eight-atom unit cell[62]. We then shifted the coordinate origin to the geometric center of the superlattice and applied a spherical truncation, retaining only Si atoms within 20-70 Bohr of the origin. The truncated cluster is passivated with hydrogen atoms to saturate dangling bonds, and the resulting structure is relaxed to its equilibrium geometry using the MMFF94 force field[63] as implemented in Open Babel software[64]. Figure 2 depicts the largest cluster, $Si_{8381}H_{1620}$.

We performed the initial periodic DFT calculations to generate the orbitals and charge densities using the `RMG`[59,60] DFT code. The DFT Hamiltonian uses the GGA PBE exchange-correlation functional with Troullier-Martins[65] norm-conserving pseudo-potentials. For each system, we performed the calculation on the Gamma k-point in a primitive cubic cell with isotropic sampling. Each cluster is separated from its periodic image by a vacuum layer of 11-17 bohr. The initial DFT step provides the energy estimates E^{KS}_{HOMO} and E^{KS}_{LUMO} used to define the gapped filter for the GW step. Cell and grid parameters, along with HOMO

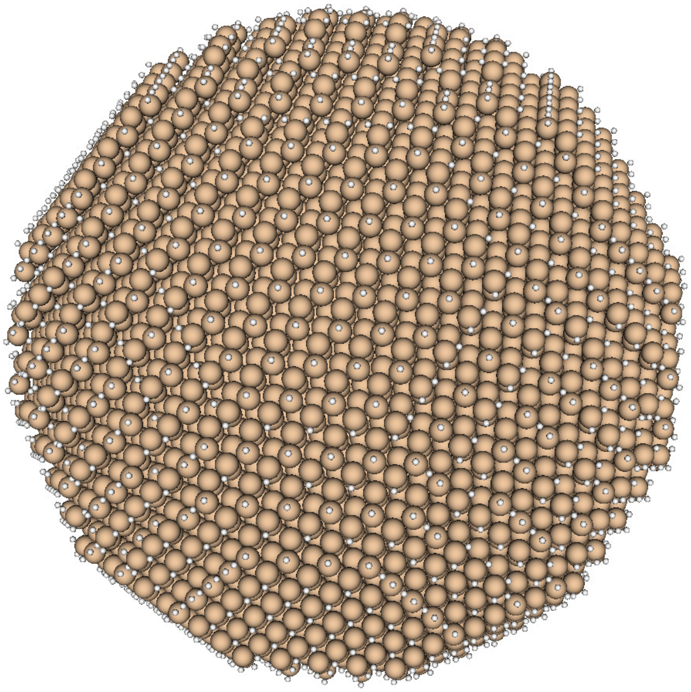


Figure 2: $Si_{8381}H_{1620}$ cluster. Silicon and hydrogen atoms shown as brown and white spheres, respectively.

and LUMO energies, are listed in Table 1.

Table 1: Details of preliminary DFT calculations on each cluster, including numbers of electrons (N_e), points in the spatial grid (N_g), grid spacing (Δ_g, bohr), along with resulting HOMO and LUMO energies and band gaps (eV).

System	N_e	N_g	Δ_g	E^{KS}_{HOMO}	E^{KS}_{LUMO}	KS Band Gap
$Si_{293}H_{172}$	1344	128^3	0.4429	-3.259	-1.526	1.733
$Si_{705}H_{300}$	3120	128^3	0.5167	-1.928	-0.457	1.471
$Si_{5031}H_{1172}$	21296	256^3	0.5000	-1.705	-0.776	0.928
$Si_{7745}H_{1572}$	32552	256^3	0.5400	-0.986	-0.121	0.865
$Si_{8381}H_{1620}$	35144	256^3	0.5600	-1.095	-0.245	0.851

We then used `StochasticGW` to compute ε_{QP} for the HOMO and LUMO orbitals of each system. The Kohn-Sham Hamiltonian in `StochasticGW` uses the same pseudopotentials and grids as the previous DFT step; here, we employ the PBE functional[66] as implemented in the LibXC[67] library and apply an energy cutoff of 28 Hartrees to the kinetic energy operator. In the filtering step, for the largest system we studied, $Si_{8381}H_{1620}$, the energy difference $E^{KS}_{LUMO} - E^{KS}_{HOMO}$ is $\sim$ 0.11% the full spectral range of $\hat{H}^{KS}$. Even though the cutoff is spread over the full band gap, it is still sharp enough to require 8192 Chebyshev terms to reduce the Gibbs oscillations to negligible levels outside of the band gap (Figure 3). While this filter length is an order of magntude larger than that used in our recent calculation on the napthalene molecule ($N_{chb} = 450$)[51], it is still less than lengths required in earlier calculations performed on much smaller systems without gapped filtering ($N_{chb} \sim 18000$)[49].

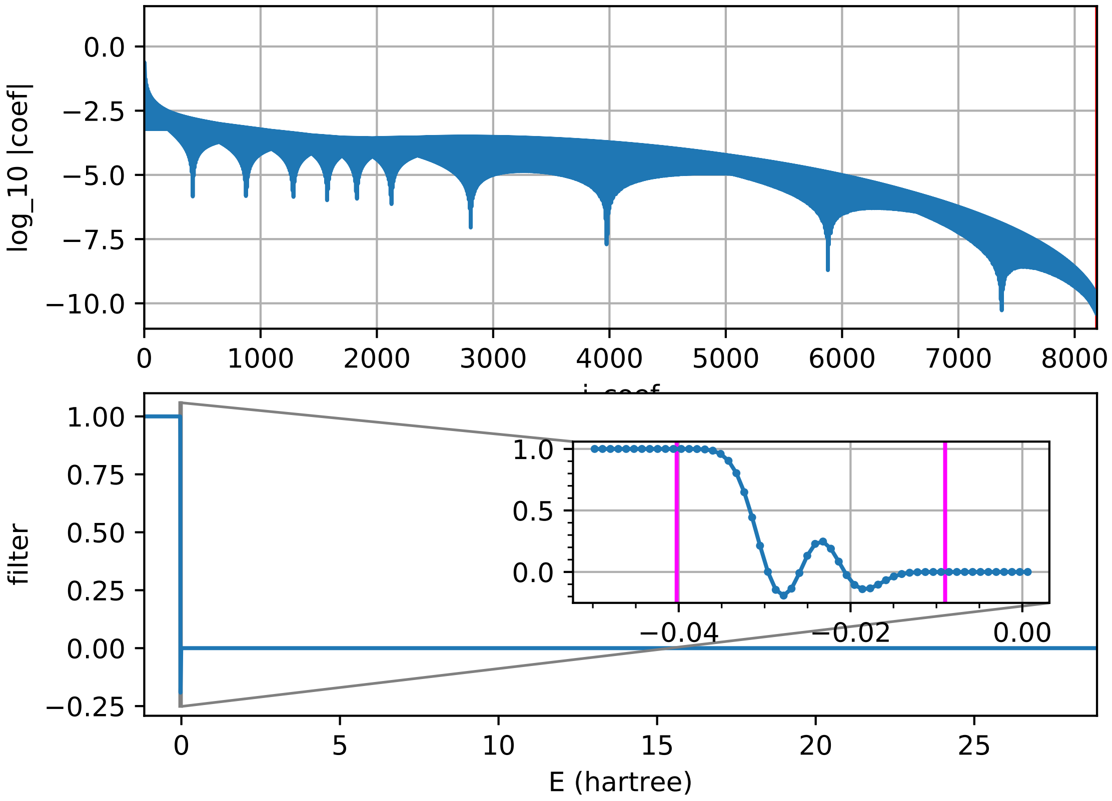


Figure 3: (Top) plot of the log of the absolute magnitudes of Chebyshev coefficients used to construct the gapped filter for the $Si_{8381}H_{1620}$ cluster. (Bottom) Reconstructed filter, where the inset shows an expansion of the region of the band gap. The purple vertical lines in the inset indicate the positions of E^{KS}_{HOMO} and E^{KS}_{LUMO}.

For each calculation we averaged 1024 Monte Carlo samples which is sufficient to achieve a statistical error better than 0.03 eV for all QP energies. The number of time steps, N_τ, are controlled internally by the energy-broadening parameter, γ, which we apply when Fourier transforming the self-energy element from the time domain to the frequency domain,

$$\langle \phi_k \left| \Sigma\left(\omega\right) \right| \phi_k \rangle = \int \langle \phi_k \left| \Sigma\left(t\right) \right| \phi_k \rangle e^{-\frac{\gamma^2 t^2}{2}} e^{i\omega t} dt. \tag{7}$$

We use a time step size of $\Delta t = 0.05 E_h^{-1}\hbar$ in the split-operator propagation of the orbitals. The number of time steps to obtain a desired energy resolution is $N_\tau \approx \frac{3}{\gamma \cdot \Delta t}$; for all calculations in this work, we set $\gamma = 0.06 E_h \hbar^{-1}$ which yields a propagation length of $N_\tau = 1000$ time steps over 50 atomic time units. Numbers of occupied stochastic orbitals (N_η) and segmented stochastic functions (N_ξ) along with the fractional grid lengths (f_g) for the latter are chosen at values similar to those in previous works[17,49,50]. Input parameters are summarized in Table 2. All calculations were run on 256 GPU nodes of NERSC-Perlmutter; each node contains one AMD EPYC 7763 processor running at 2.5 GHz and 4 NVIDIA A100 GPUs.

Table 2: Parameters used in stochastic GW calculations.

Description	Parameter	Value
Number of Monte Carlo samples	N_ζ	1024
Number of occupied stochastic orbitals	N_η	8
Number of segmented stochastic functions	N_ξ	10000
Grid fraction of each segmented function	f_g	0.003
Number of Chebyshev polynomials in filter	N_{chb}	8192
Damping parameter (Hartrees)	γ	0.06
Kinetic energy cutoff (Hartrees)	E^k_{cut}	28.0

Figure 4 plots the QP energies of the HOMO and LUMO and their difference for each system; these values are also listed in Table 3. The statistical errors in the MC energies are shown as error bars in the HOMO and LUMO traces and are small compared to the magnitudes of the energies. Moreover, comparing the bandgaps across the five clusters, the bandgaps show convergent behavior towards ~ 1.36 eV, suggesting that the largest clusters are approaching the bulk limit for our choice of density functional and pseudopotential.

Table 3 also lists wall times for all calculations. The two smaller clusters have comparable times of 800 ± 40 s and the larger three clusters have timings of 2700 ± 120 s, where the main factor behind the difference is the size of the spatial grid (128^3 vs 256^3 for the smaller and larger clusters, respectively). For a given spatial grid, one expects an increase in runtime

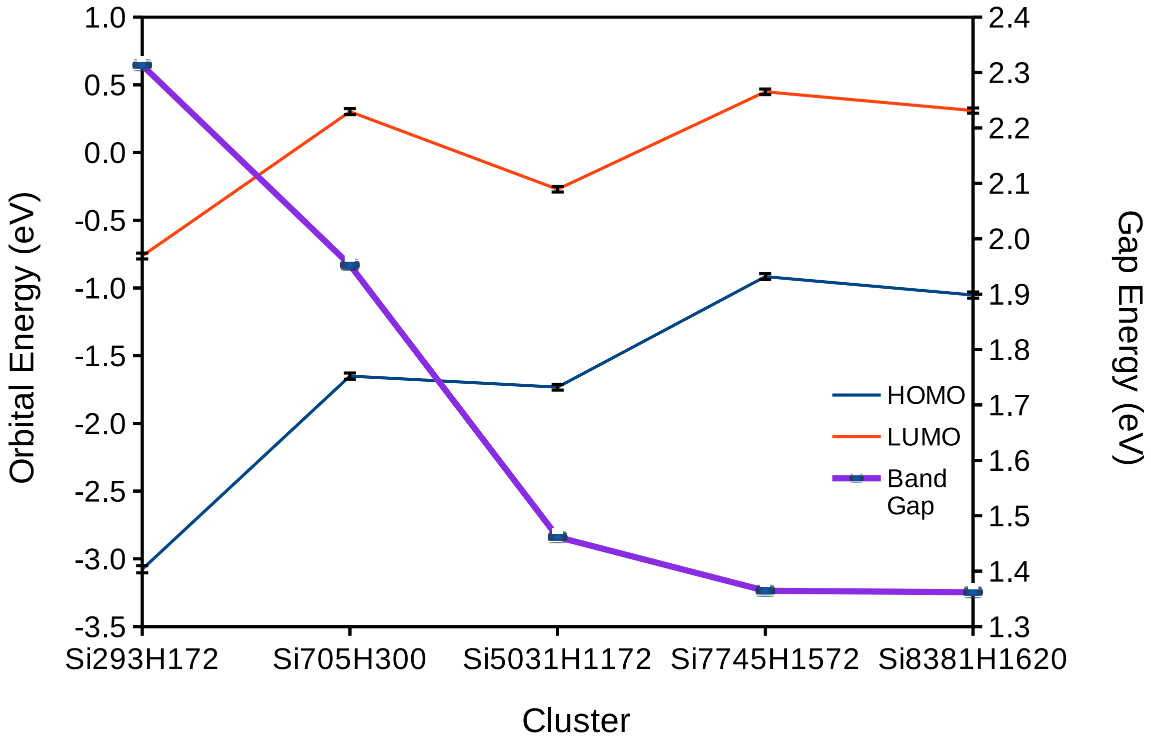


Figure 4: QP orbital energies and band gaps for each cluster.

for the larger systems for two reasons: first, the potential energy terms containing the pseudopotential contribution must be applied via atomic operations on grid points where pseudopotentials for neighboring atoms overlap. Second, the higher density of states in larger systems causes the spectral range estimation to converge more slowly. However, these differences are smaller than the variation in performancein MPI and I/O operations that occur over the large scale of these runs.

Table 3: Results of stochastic GW calculations on each cluster including quasiparticle energies and band gaps (eV) and calculation wall times (s).

System	E^{QP}_{HOMO}	E^{QP}_{LUMO}	QP Band Gap	t^{wall}_{HOMO}	t^{wall}_{LUMO}
$Si_{293}H_{172}$	-3.077 ± 0.027	-0.764 ± 0.022	2.313	836	770
$Si_{705}H_{300}$	-1.650 ± 0.023	0.302 ± 0.022	1.953	835	788
$Si_{5031}H_{1172}$	-1.732 ± 0.021	-0.271 ± 0.020	1.462	2609	2617
$Si_{7745}H_{1572}$	-0.916 ± 0.022	0.449 ± 0.021	1.365	2702	2688
$Si_{8381}H_{1620}$	-1.052 ± 0.023	0.310 ± 0.029	1.362	2669	2812

We also performed a set of tests to measure the efficiency of parallelizing over Monte

Carlo samples for the HOMO calculation of the largest $Si_{8381}H_{1620}$ system (Figure 5). Here, the number of samples, N_ζ, is set equal to both the number of MPI ranks and the number of GPUs, so this test is a measure of the weak scaling of the code. Note that the total runtimes are all ∼2500 s, similar to the full 1024-sample run, demonstrating nearly ideal scaling with number of samples.

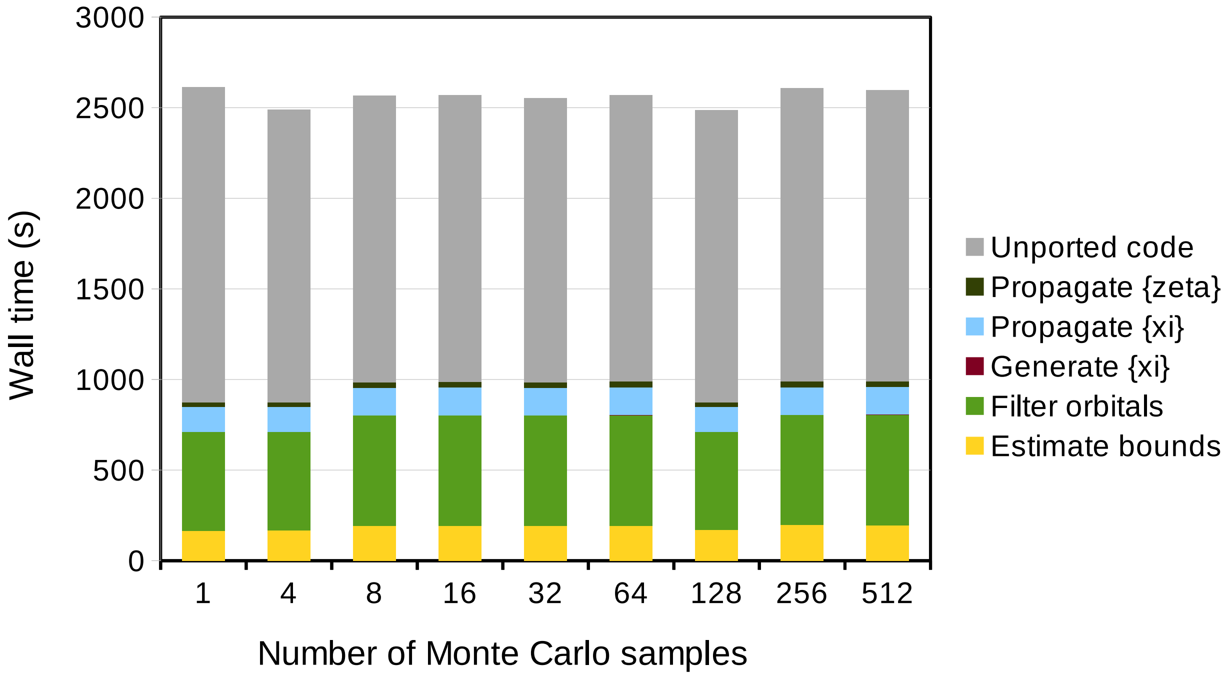


Figure 5: Wall times spent in each portion of the code for calculations on the HOMO state of $Si_{8381}H_{1620}$, with different numbers of Monte Carlo samples.

From Figure 5, one can see that the portions of the code that have been ported to GPUs collectively account for ∼38 % of the total runtime. The remaining, unported portion of the code includes I/O operations, preparation of the grid representation of the Hamiltonian, constructing the $\{\zeta\}$ and $\{\xi\}$ orbitals prior to filtering, solving the linear system to generate the filter coefficients[51], and collecting and post-processing the samples to produce the final QP energies.

We also measured the speedup factors for the individual GPU-ported sections of the code relative to their CPU counterparts. A full-scale run of the original CPU code is not possible for the larger clusters due to the wall time limit on NERSC-Perlmutter, so we instead performed several comparative tests focusing on individual portions of the code (Table 4). We ran each test for the HOMO state of $\mathrm{Si}_{8381}\mathrm{H}_{1620}$ on a single MC sample with other parameters same as in Table 2 except that here we decreased the filter length by a factor of 16 (to $N_{chb} = 512$) to reduce the CPU time needed for this step. As Table 4 shows, the GPU filtering step achieves a $\sim 50\times$ speedup over its CPU counterpart. The propagation and spectral estimation steps achieve even higher speedups of $150 - 250\times$. This is due not only to porting the routines to GPUs but also to optimizations that were not present in the CPU code, such as premultiplying potential energy factors before offloading them to the GPU. The step to generate the $\{\xi\}$ segments, while requiring much less time than the propagation and filtering steps, exhibited the largest performance improvement resulting from replacing the serial calls to the KISS random number generator with calls to the `cuRAND` library. Finally, the last two rows of Table 4 list the timings of the unported code and the sum of all timings, including unported portions of the code, showing that the overall speedup of the GPU implementation of `StochasticGW` is $\sim 45\times$ that of the CPU code.

Table 4: Timings and speedups of GPU portions of `StochasticGW` relative to the CPU portions, for calculation on the HOMO state of $\mathrm{Si}_{8381}\mathrm{H}_{1620}$.

Portion	t_{CPU} (s)	t_{GPU} (s)	t_{CPU}/t_{GPU}
Propagate $\{\zeta\}$	4947	31	160
Propagate $\{\eta\}$	37711	153	246
Generate $\{\xi\}$	662	0.08	8764
Filter $\{\zeta\}$,$\{\eta\}$	9796	199	49
Estimate $[E_{min}, E_{max}]$	26364	191	138
Unported code	1237		1
Total (incl. unported)	81292	1811	45

Conclusion

In this work, we describe a new implementation of the `StochasticGW` code. Our code utilizes the stochastic Resolution of the Identity (sROI) technique, which allows one to decouple the main steps of the GW method into independent, statistical operations that can be performed massively in parallel. In deterministic GW methods, the cost is dominated by computing matrix elements over indices representing the occupied and unoccupied orbitals. In contrast, in the stochastic method the cost depends on operations over the full spatial grid and accumulating a sufficient number of Monte Carlo samples to achieve a desired statistical accuracy. Therefore, compared to deterministic GW, the cost of stochastic GW grows much more slowly with respect to the size of the molecule or material system of interest.

Motivated by the large-scale parallelism available in modern GPU hardware, we have ported the major computational motifs of the algorithm to GPUs. These include estimating the spectral width of the Kohn-Sham Hamiltonian, filtering the initial orbitals by projecting onto an occupied subspace, and propagating the orbitals under the influence of a time-dependent Hamiltonian. Each of these steps is applied via a sequence of vectorized OpenACC kernels and calls to GPU-optimized FFT libraries.

We showcased the GPU implementation by computing the quasi-particle energies of the HOMO and LUMO orbitals of five hydrogen-passivated silicon clusters. The band gaps show convergent behavior towards a bulk-like limit at ca. 1.36 eV. QP calculations on the largest system, $Si_{8381}H_{1620}$, with 10001 atoms and 35144 electrons, can be completed in only ~ 45 minutes with the workload partitioned with one MC sample per GPU. For this system, the GPU version of `StochasticGW` achieves roughly 45× speedup in time-to-solution relative to the CPU version over the entire execution of the code. This work opens the way for computing QP energies of even larger systems.

Author information

Corresponding author Phillip Thomas – National Energy Research Scientific Computing Center (NERSC), Lawrence Berkeley National Laboratory, Berkeley, California 94720, United States; orcid.org/0009-0007-3794-0150; Email: pthomas@lbl.gov

Authors Daniel Neuhauser – Department of Chemistry and Biochemistry, University of California Los Angeles, Los Angeles, California 90095, United States; orcid.org/ 0000-0003-3160-386X; Email: dxn@ucla.edu

Minh Nguyen – Department of Chemistry and Biochemistry, University of California, Los Angeles, California 90095, United States

Dimitri Bazile – Department of Chemistry and Biochemistry, University of California Los Angeles, Los Angeles, California 90095, United States

Tucker Allen – Department of Chemistry and Biochemistry, University of California Los Angeles, Los Angeles, California 90095, United States

Barry Y. Li – Department of Chemistry and Biochemistry, University of California Los Angeles, Los Angeles, California 90095, United States

Wenfei Li – Department of Chemistry and Biochemistry, University of California Los Angeles, Los Angeles, California 90095, United States

Mauro Del Ben – Applied Mathematics and Computational Research Division, Lawrence Berkeley National Laboratory, Berkeley, California 94720, United States; orcid.org/0000-0003-0755-4797

Jack Deslippe – National Energy Research Scientific Computing Center (NERSC), Lawrence Berkeley National Laboratory, Berkeley, California 94720, United States; orcid.org/0000-0003-1785-4187

Author Contributions [§]P.T. is first author.

Notes The authors declare no competing financial interest.

Acknowledgement

This work was supported through the Center for Computational Study of Excited State Phenomena in Energy Materials (C2SEPEM), which is funded by the U.S. Department of

Energy, Office of Science, Basic Energy Sciences, Materials Sciences and Engineering Division, via contract no. DE-AC02-05CH11231, as part of the Computational Materials Sciences Program. Work in Daniel Neuhauser's group was supported by the National Science Foundation Grant No. CHE2245253. Computational resources were provided by the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility operating under contract no. DE-AC02-05CH11231 using NERSC awards BES-ERCAP0032245 (project m2651) and DDR-NESAP-ERCAP0033923 (project m4941).

References

(1) Werner, H.-J.; Knowles, P. J. An efficient internally contracted multiconfiguration reference configuration interaction method. *The Journal of Chemical Physics* **1988**, *89*, 5803–5814.

(2) Buenker, R. J.; Peyerimhoff, S. D.; Butscher, W. Applicability of the multi-reference double-excitation CI (MRD-CI) method to the calculation of electronic wavefunctions and comparison with related techniques. *Molecular Physics* **1978**, *35*, 771–791.

(3) Krylov, A. I. Equation-of-Motion Coupled-Cluster Methods for Open-Shell and Electronically Excited Species: The Hitchhikers Guide to Fock Space. *Annual Review of Physical Chemistry* **2008**, *59*, 433–462.

(4) Stanton, J. F.; Bartlett, R. J. The equation of motion coupled-cluster method. A systematic biorthogonal approach to molecular excitation energies, transition probabilities, and excited state properties. *The Journal of Chemical Physics* **1993**, *98*, 7029–7039.

(5) Hohenberg, P.; Kohn, W. Inhomogeneous Electron Gas. *Physical Review* **1964**, *136*, B864.

(6) Das, S.; Kanungo, B.; Subramanian, V.; Panigrahi, G.; Motamarri, P.; Rogers, D.; Zimmerman, P.; Gavini, V. Large-Scale Materials Modeling at Quantum Accuracy: Ab Initio Simulations of Quasicrystals and Interacting Extended Defects in Metallic Alloys. Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. 2023; pp 1–12.

(7) Martin, R. M. *Electronic Structure: Basic Theory and Practical Methods*; Cambridge University Press, 2004; p 624.

(8) Dreizler, R. M.; Gross, E. K. U. *Density Functional Theory*; Springer Berlin Heidelberg, 1990.

(9) Aryasetiawan, F.; Gunnarsson, O. The GW method. *Reports on Progress in Physics* **1998**, *61*, 237–312.

(10) Runge, E.; Gross, E. K. U. Density-Functional Theory for Time-Dependent Systems. *Physical Review Letters* **1984**, *52*, 997–1000.

(11) Hedin, L. New Method for Calculating the One-Particle Greens Function with Application to the Electron-Gas Problem. *Physical Review* **1965**, *139*, A796–A823.

(12) Martin, R. M. In *Interacting electrons*; Reining, L., Ceperley, D. M., Eds.; Cambridge University Press: Cambridge, 2016.

(13) Li, Z.; Antonius, G.; Wu, M.; da Jornada, F. H.; Louie, S. G. Electron-Phonon Coupling from AbÂ Initio Linear-Response Theory within the GW Method: Correlation-Enhanced Interactions and Superconductivity in $Ba_{1-x}K_xBiO_3$. *Physical Review Letters* **2019**, *122*, 186402.

(14) Onida, G.; Reining, L.; Rubio, A. Electronic excitations: density-functional versus many-body Green's-function approaches. *Reviews of Modern Physics* **2002**, *74*, 601–659.

(15) Del Ben, M.; da Jornada, F. H.; Canning, A.; Wichmann, N.; Raman, K.; Sasanka, R.; Yang, C.; Louie, S. G.; Deslippe, J. Large-scale GW calculations on pre-exascale HPC systems. *Computer Physics Communications* **2019**, *235*, 187–195.

(16) Del Ben, M.; Yang, C.; Li, Z.; Jornada, F. H. d.; Louie, S. G.; Deslippe, J. Accelerating Large-Scale Excited-State GW Calculations on Leadership HPC Systems. SC20: International Conference for High Performance Computing, Networking, Storage and Analysis. 2020; pp 1–11.

(17) Brooks, J.; Weng, G.; Taylor, S.; Vlcek, V. Stochastic many-body perturbation theory for Moire states in twisted bilayer phosphorene. *Journal of Physics: Condensed Matter* **2020**, *32*, 234001.

(18) Yu, V. W.-z.; Govoni, M. GPU Acceleration of Large-Scale Full-Frequency GW Calculations. *Journal of Chemical Theory and Computation* **2022**, *18*, 4690–4707.

(19) Wu, W. et al. Enabling 13K-Atom Excited-State GW Calculations via Low-Rank Approximations and HPC on the New Sunway Supercomputer. SC24: International Conference for High Performance Computing, Networking, Storage and Analysis. 2024; pp 1–14.

(20) Zhang, B.; Weinberg, D.; Hsu, C.-E.; Altman, A. R.; Shi, Y.; White, J. B.; Vigil-Fowler, D.; Louie, S. G.; Deslippe, J. R.; da Jornada, F. H.; Li, Z.; Del Ben, M. Advancing Quantum Many-Body GW Calculations on Exascale Supercomputing Platforms. Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. 2025; pp 48–59.

(21) Hybertsen, M. S.; Louie, S. G. First-Principles Theory of Quasiparticles: Calculation of Band Gaps in Semiconductors and Insulators. *Physical Review Letters* **1985**, *55*, 1418–1421.

(22) Hybertsen, M. S.; Louie, S. G. Electron correlation in semiconductors and insulators: Band gaps and quasiparticle energies. *Physical Review B* **1986**, *34*, 5390–5413.

(23) Kresse, G.; Hafner, J. Ab initiomolecular dynamics for liquid metals. *Physical Review B* **1993**, *47*, 558–561.

(24) Kresse, G.; Hafner, J. Ab initio molecular-dynamics simulation of the liquid-metal-amorphous-semiconductor transition in germanium. *Physical Review B* **1994**, *49*, 14251–14269.

(25) Giannozzi, P. et al. QUANTUM ESPRESSO: a modular and open-source software project for quantum simulations of materials. *Journal of Physics: Condensed Matter* **2009**, *21*, 395502.

(26) Gonze, X. et al. ABINIT: First-principles approach to material and nanosystem properties. *Computer Physics Communications* **2009**, *180*, 2582–2615.

(27) Marini, A.; Hogan, C.; Gruning, M.; Varsano, D. Yambo: An ab initio tool for excited state calculations. *Computer Physics Communications* **2009**, *180*, 1392–1403.

(28) Blum, V.; Gehrke, R.; Hanke, F.; Havu, P.; Havu, V.; Ren, X.; Reuter, K.; Scheffler, M. Ab initio molecular simulations with numeric atom-centered orbitals. *Computer Physics Communications* **2009**, *180*, 2175–2196.

(29) Deslippe, J.; Samsonidze, G.; Strubbe, D. A.; Jain, M.; Cohen, M. L.; Louie, S. G. BerkeleyGW: A massively parallel computer package for the calculation of the quasiparticle and optical properties of materials and nanostructures. *Computer Physics Communications* **2012**, *183*, 1269–1289.

(30) The Elk code. `http://elk.sourceforge.net/`.

(31) Gulans, A.; Kontur, S.; Meisenbichler, C.; Nabok, D.; Pavone, P.; Rigamonti, S.; Sagmeister, S.; Werner, U.; Draxl, C. exciting: a full-potential all-electron package im-

plementing density-functional theory and many-body perturbation theory. *Journal of Physics: Condensed Matter* **2014**, *26*, 363202.

(32) Govoni, M.; Galli, G. Large Scale GW Calculations. *Journal of Chemical Theory and Computation* **2015**, *11*, 2680–2696.

(33) Jacquemin, D.; Duchemin, I.; Blase, X. Benchmarking the Bethe-Salpeter Formalism on a Standard Organic Molecular Set. *Journal of Chemical Theory and Computation* **2015**, *11*, 3290–3304.

(34) Bruneval, F.; Rangel, T.; Hamed, S. M.; Shao, M.; Yang, C.; Neaton, J. B. MOLGW 1: Many-body perturbation theory software for atoms, molecules, and clusters. *Computer Physics Communications* **2016**, *208*, 149–161.

(35) Kuhne, T. D. et al. CP2K: An electronic structure and molecular dynamics software package - Quickstep: Efficient and accurate electronic structure calculations. *The Journal of Chemical Physics* **2020**, *152*, 194103.

(36) Schlipf, M.; Lambert, H.; Zibouche, N.; Giustino, F. SternheimerGW: A program for calculating GW quasiparticle band structures and spectral functions without unoccupied states. *Computer Physics Communications* **2020**, *247*, 106856.

(37) Bruneval, F.; Vast, N.; Reining, L. Effect of self-consistency on quasiparticles in solids. *Physical Review B* **2006**, *74*, 045102.

(38) van Schilfgaarde, M.; Kotani, T.; Faleev, S. Quasiparticle Self-Consistent GW Theory. *Physical Review Letters* **2006**, *96*, 226402.

(39) Vlcek, V.; Baer, R.; Rabani, E.; Neuhauser, D. Simple eigenvalue-self-consistent ΔGW. *The Journal of Chemical Physics* **2018**, *149*, 174107.

(40) Allen, T.; Nguyen, M.; Neuhauser, D. GW with hybrid functionals for large molecular systems. *The Journal of Chemical Physics* **2024**, *161*, 164116.

(41) Rieger, M. M.; Steinbeck, L.; White, I.; Rojas, H.; Godby, R. The GW space-time method for the self-energy of large systems. *Computer Physics Communications* **1999**, *117*, 211–228.

(42) Liu, P.; Kaltak, M.; Klimes, J.; Kresse, G. Cubic-scaling GW: Towards fast quasiparticle calculations. *Physical Review B* **2016**, *94*, 165109.

(43) Wilhelm, J.; Golze, D.; Talirz, L.; Hutter, J.; Pignedoli, C. A. Toward GW Calculations on Thousands of Atoms. *The Journal of Physical Chemistry Letters* **2018**, *9*, 306–312.

(44) Kim, M.; Martyna, G. J.; Ismail-Beigi, S. Complex-time shredded propagator method for large-scale GW calculations. *Physical Review B* **2020**, *101*, 035139.

(45) Yeh, C.-N.; Morales, M. A. Low-Scaling Algorithms for GW and Constrained Random Phase Approximation Using Symmetry-Adapted Interpolative Separable Density Fitting. *Journal of Chemical Theory and Computation* **2024**, *20*, 3184–3198.

(46) Altman, A. R.; Kundu, S.; da Jornada, F. H. Mixed Stochastic-Deterministic Approach for Many-Body Perturbation Theory Calculations. *Physical Review Letters* **2024**, *132*, 086401.

(47) Vetsch, N.; Maeder, A.; Maillou, V.; Winka, A.; Cao, J.; Kwasniewski, G.; Deuschle, L.; Hoefler, T.; Ziogas, A. N.; Luisier, M. Ab-initio Quantum Transport with the GW Approximation, 42,240 Atoms, and Sustained Exascale Performance. **2025**, 1–13.

(48) Neuhauser, D.; Gao, Y.; Arntsen, C.; Karshenas, C.; Rabani, E.; Baer, R. Breaking the Theoretical Scaling Limit for Predicting Quasiparticle Energies: The Stochastic GW Approach. *Physics Review Letters* **2014**, *113*, 076402.

(49) Vlcek, V.; Rabani, E.; Neuhauser, D.; Baer, R. Stochastic GW Calculations for Molecules. *Journal of Chemical Theory and Computation* **2017**, *13*, 4997–5003.

(50) Vlcek, V.; Li, W.; Baer, R.; Rabani, E.; Neuhauser, D. Swift GW beyond 10,000 electrons using sparse stochastic compression. *Physical Review B* **2018**, *98*, 075107.

(51) Nguyen, M.; Neuhauser, D. Gapped-filtering for efficient Chebyshev expansion of the density projection operator. *Chemical Physics Letters* **2022**, *806*, 140036.

(52) Hutchinson, M. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics - Simulation and Computation* **1990**, *19*, 433–450.

(53) Baer, R.; Neuhauser, D. Real-time linear response for time-dependent density-functional theory. *The Journal of Chemical Physics* **2004**, *121*, 9803–9807.

(54) Baer, R.; Head-Gordon, M. Chebyshev expansion methods for electronic structure calculations on large molecular systems. *The Journal of Chemical Physics* **1997**, *107*, 10003–10013.

(55) Fetter, A. L.; Walecka, J. D. *Quantum Theory of Many Particle Systems*; McGraw-Hill: New York, 1971; p 299.

(56) The StochasticGW code. `https://github.com/stochasticGW/stochasticGW`.

(57) Frigo, M.; Johnson, S. The Design and Implementation of FFTW3. *Proceedings of the IEEE* **2005**, *93*, 216–231.

(58) Giannozzi, P.; Baseggio, O.; Bonfa, P.; Brunato, D.; Car, R.; Carnimeo, I.; Cavazzoni, C.; de Gironcoli, S.; Delugas, P.; Ferrari Ruffino, F.; Ferretti, A.; Marzari, N.; Timrov, I.; Urru, A.; Baroni, S. Quantum ESPRESSO toward the exascale. *The Journal of Chemical Physics* **2020**, *152*, 154105.

(59) Briggs, E. L.; Sullivan, D. J.; Bernholc, J. Real-space multigrid-based approach to large-scale electronic structure calculations. *Physical Review B* **1996**, *54*, 14362–14375.

(60) Hodak, M.; Wang, S.; Lu, W.; Bernholc, J. Implementation of ultrasoft pseudopotentials in large-scale grid-based electronic structure calculations. *Physical Review B* **2007**, *76*, 085108.

(61) Hunter, J. D. Matplotlib: A 2D Graphics Environment. *Computing in Science and Engineering* **2007**, *9*, 90–95.

(62) Hom, T.; Kiszenik, W.; Post, B. Accurate lattice constants from multiple reflection measurements. II. Lattice constants of germanium silicon, and diamond. *Journal of Applied Crystallography* **1975**, *8*, 457–458.

(63) Halgren, T. A. Merck molecular force field. I. Basis, form, scope, parameterization, and performance of MMFF94. *Journal of Computational Chemistry* **1996**, *17*, 490–519.

(64) O'Boyle, N. M.; Banck, M.; James, C. A.; Morley, C.; Vandermeersch, T.; Hutchison, G. R. Open Babel: An open chemical toolbox. *Journal of Cheminformatics* **2011**, *3*.

(65) Troullier, N.; Martins, J. L. Efficient pseudopotentials for plane-wave calculations. *Physical Review B* **1991**, *43*, 1993–2006.

(66) Perdew, J. P.; Burke, K.; Ernzerhof, M. Generalized Gradient Approximation Made Simple. *Physical Review Letters* **1996**, *77*, 3865–3868.

(67) Lehtola, S.; Steigemann, C.; Oliveira, M. J.; Marques, M. A. Recent developments in libxc - A comprehensive library of functionals for density functional theory. *SoftwareX* **2018**, *7*, 1–5.