JCTC Journal of Chemical Theory and Computation

pubs.acs.org/JCTC Article

GPU Acceleration of Large-Scale Full-Frequency GW Calculations

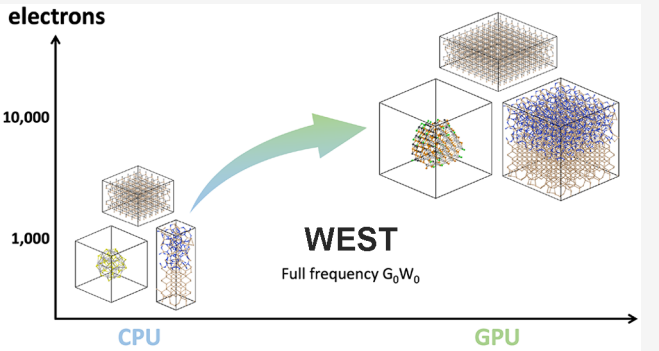
Victor Wen-zhe Yu and Marco Govoni*

Cite This: J. Chem. Theory Comput. 2022, 18, 4690−4707

Read Online

ACCESS | Metrics & More | Article Recommendations

ABSTRACT: Many-body perturbation theory is a powerful method to simulate electronic excitations in molecules and materials starting from the output of density functional theory calculations. By implementing the theory efficiently so as to run at scale on the latest leadership high-performance computing systems it is possible to extend the scope of GW calculations. We present a GPU acceleration study of the full-frequency GW method as implemented in the WEST code. Excellent performance is achieved through the use of (i) optimized GPU libraries, e.g., cuFFT and cuBLAS, (ii) a hierarchical parallelization strategy that minimizes CPU−CPU, CPU−GPU, and GPU−GPU data transfer operations, (iii) nonblocking MPI communications that overlap with GPU computations, and (iv) mixed precision in selected portions of the code. A series of performance benchmarks has been carried out on leadership high-performance computing systems, showing a substantial speedup of the GPU-accelerated version of WEST with respect to its CPU version. Good strong and weak scaling is demonstrated using up to 25 920 GPUs. Finally, we showcase the capability of the GPU version of WEST for large-scale, full-frequency GW calculations of realistic systems, e.g., a nanostructure, an interface, and a defect, comprising up to 10 368 valence electrons.



1. INTRODUCTION

First-principles simulations of materials have become mainstream computational instruments to understand energy conversion processes in several areas of materials science and chemistry, including, for instance, applications to photovoltaics and photocatalysis. Simulations using the Kohn−Sham density functional theory (KS-DFT)[1,2] are widely adopted to computationally predict the structures and properties of molecules and materials in their ground state. However, KS-DFT methods fail to provide an accurate description of electrons in excited states. The GW method, formulated within the context of many-body perturbation theory,[3,4] has been established as the main method to improve the electronic structure obtained with DFT and describe excited states. The GW self-energy was initially proposed by Hedin[5] as a numerically manageable approximation to the complex many-body nature of electron−electron interactions. The earliest applications of the GW method to the electronic structure of semiconductors and insulators obtained with DFT date back to the 1980s.[6−10] Conventional GW implementations, currently available in several electronic structure codes, have a computational cost that scales as $O(N^4)$ with respect to the system size N, limiting the tractable size of GW calculations. Method development and code optimization have been active areas of research in order to push the scope of applicability of such GW implementations to large systems. Formulations with cubic scaling algorithms[11−17] or stochastic methods[18−22] have been proposed, albeit at the cost of introducing expensive numerical integration operations or stochastic errors, respectively.

The rise of heterogeneous computing has substantially increased the throughput available in leadership high-performance computing (HPC) systems to hundreds of PFLOP/s (peta floating-point operations per second), and we are currently witnessing the transition to the exascale. On the current release (November 2021) of the TOP500 list,[23] seven of the top 10 supercomputers have graphics processing units (GPUs), including Summit, the world's second fastest computer powered by 27 648 NVIDIA V100 GPUs. GPU devices consist of hundreds to thousands of cores that operate at a relatively low frequency and can perform parallel computational tasks in a more energy efficient way than by central processing units (CPUs). This sets tremendous opportunities for first-principles simulations, including the ability to carry out GW calculations at unprecedented scales.

Received: March 10, 2022
Published: August 1, 2022

However, most software packages in the electronic structure community were initially written to target traditional CPUs with parallelization primarily managed by the message passing interface (MPI). The migration to accelerated, heterogeneous computing typically requires a redesign of the code to fully harness the parallelism of modern GPUs. GPU acceleration has been reported by a number of electronic structure theory and quantum chemistry software packages.[24−31] For the GW method in particular, the Gaussian-orbital-based VOTCA-XTP code[32] and the plane-wave-based Yambo[33] code can perform GPU-accelerated GW calculations of molecules and materials. The plane-wave-based BerkeleyGW code was recently ported to run on GPUs to carry out a large-scale GW calculation for a silicon model consisting of 10 968 valence electrons using a generalized plasmon-pole model to approximate retardation effects.[34]

In this paper, we present the GPU porting of the WEST code,[35,36] a plane-wave pseudopotential implementation of the full-frequency G_0W_0 method. In addition to featuring a massive parallelization, demonstrated using over ∼500 000 CPU cores in ref 35, WEST uses techniques to help prevent computational and memory bottlenecks for large systems; for instance, it represents the density−density response functions in a compact basis set, eliminating the need to store and manipulate large matrices. The slowly converging sum over empty KS states, commonly encountered in most GW codes, is avoided completely in WEST. WEST carries out a full integration over the frequency domain, removing the need of approximating retardation effects with plasmon-pole models. The accuracy of the full-frequency implementation in WEST was recently assessed, verifying the implementation by comparing the results obtained with WEST with the results of all-electron codes.[37] The WEST code has been used to study excited states for a variety of systems, including molecules, nanoparticles, two-dimensional (2D) materials, solids, defects in solids, liquids, amorphous, and solid/liquid interfaces.[37−49] Recent developments within WEST include the computation of electron−phonon self-energies[50,51] and absorption spectra[52,53] and the formulation of a quantum embedding approach.[54−56] The GPU porting of WEST aims to further advance the simulation of electronic excitations in large, complex materials on a variety of GPU-powered, pre-exascale and exascale HPC systems. The strategy reported here is general and can be applied to other GW codes.

The rest of the paper is organized as follows. In section 2, we briefly review the G_0W_0 theory and the current state of the art. In section 3, we summarize the implementation in the WEST code. We then introduce the GPU porting of WEST in section 4, elaborating on several optimization strategies that help maximize the efficiency of the code, especially when running on a large number of GPUs. The performance of the newly developed GPU version of WEST is discussed in section 5 with a series of benchmarks, demonstrating excellent performance and scalability on leadership HPC systems. In section 6, we report three examples of large full-frequency G_0W_0 calculations. Our conclusions are given in section 7.

2. G_0W_0 METHOD

2.1. Theory. In KS-DFT,[1,2] the ground state of a system of interacting electrons in the external field of the ions may be obtained by solving the KS set of single-particle equations

$$h^{\sigma}_{\mathrm{KS}}\psi_{i\sigma} = \epsilon_{i\sigma}\psi_{i\sigma} \tag{1}$$

where $\psi_{i\sigma}$ and $\epsilon_{i\sigma}$ correspond to the wave function and energy of the ith KS state in the σ spin channel, respectively. The KS Hamiltonian, h^{σ}_{KS}, includes the single-particle kinetic energy operator, t_s, and the Hartree, external (ionic), and exchange-correlation potential operators v_{H}, v_{ext}, and v^{σ}_{xc}, respectively. Throughout the paper we focus on large systems that do not require k-point sampling; therefore, we omit k-point indices for simplicity.

Quasiparticle (QP) states may be obtained by solving the following Dyson-like equation

$$h^{\sigma}_{\mathrm{QP}}\psi^{\mathrm{QP}}_{i\sigma} = \epsilon^{\mathrm{QP}}_{i\sigma}\psi^{\mathrm{QP}}_{i\sigma} \tag{2}$$

where the QP Hamiltonian, h^{σ}_{QP}, is obtained from the KS Hamiltonian by replacing the exchange and correlation potential with the electron self-energy Σ. The latter is a frequency-dependent and nonlocal operator that may be expressed in a compact form as

$$\Sigma = iGW\Gamma \tag{3}$$

where G, W, and Γ are the Green's function, the screened Coulomb interaction, and the vertex operator, respectively. Σ may be computed by solving Hedin's equations self-consistently.[5] Within the G_0W_0 approximation,[8−10] Γ is treated as the identity and the self-energy is evaluated not self-consistently as

$$\Sigma^{\sigma}(\boldsymbol{r}, \boldsymbol{r}'; \omega) = i\int_{-\infty}^{+\infty}\frac{\mathrm{d}\omega'}{2\pi}G^{\sigma}_0(\boldsymbol{r}, \boldsymbol{r}'; \omega+\omega')W_0(\boldsymbol{r}, \boldsymbol{r}'; \omega') \tag{4}$$

The KS states and energies may be used to evaluate all terms in the right-hand side (RHS) of eq 4, i.e., the non-self-consistent Green's function, $G^{\sigma}_0(\omega) = (\omega - h^{\sigma}_{\mathrm{KS}})^{-1}$, and the screened Coulomb potential, $W_0 = v + v^{1/2}\bar{\chi}v^{1/2}$, where $\bar{\chi}$ is the symmetrized density−density response function of the system. To obtain the latter, the irreducible density−density response function, $\chi_0 = iG_0G_0$, is first evaluated; second, $\bar{\chi}$ is obtained within the random phase approximation (RPA) using a Dyson recursive equation, $\bar{\chi} = \bar{\chi}_0 + \bar{\chi}_0\bar{\chi}$, where $\bar{\chi}_0 = v^{1/2}\chi_0v^{1/2}$.

Once the self-energy is obtained, QP energies are found using perturbation theory starting from the solution of eq 1

$$\begin{aligned}\epsilon^{\mathrm{QP}}_{i\sigma} &= \epsilon_{i\sigma} + \langle\psi_{i\sigma}|h^{\sigma}_{\mathrm{QP}} - h^{\sigma}_{\mathrm{KS}}|\psi_{i\sigma}\rangle \\ &= \epsilon_{i\sigma} + \langle\psi_{i\sigma}|\Sigma^{\sigma}(\epsilon^{\mathrm{QP}}_{i\sigma}) - v^{\sigma}_{\mathrm{xc}}|\psi_{i\sigma}\rangle\end{aligned} \tag{5}$$

The frequency integration in eq 4 can be evaluated numerically using the contour deformation technique,[57−59] i.e., by carrying out the integration in the complex plane along a contour that excludes the poles of W_0

$$\langle\psi_{i\sigma}|\Sigma^{\sigma}(\omega)|\psi_{i\sigma}\rangle = \langle\psi_{i\sigma}|\Sigma^{\sigma}_X|\psi_{i\sigma}\rangle + I_{i\sigma}(\omega) + R_{i\sigma}(\omega) \tag{6}$$

The exchange self-energy, Σ_X, is obtained by replacing W_0 in eq 4 with the frequency-independent bare Coulomb potential v. $I_{i\sigma}$ contains an integration along the imaginary axis, where G_0 and W_0 are both smooth functions

$$\begin{aligned}I_{i\sigma}(\omega) = -\int_{-\infty}^{+\infty}\frac{\mathrm{d}\omega'}{2\pi}\int\mathrm{d}\boldsymbol{r}\int\mathrm{d}\boldsymbol{r}'\psi^{*}_{i\sigma}(\boldsymbol{r})G^{\sigma}_0 \\ (\boldsymbol{r}, \boldsymbol{r}'; \omega + i\omega')W_p(\boldsymbol{r}, \boldsymbol{r}'; i\omega')\psi_{i\sigma}(\boldsymbol{r}')\end{aligned} \tag{7}$$

The $R_{i\sigma}$ term contains the residues associated with the poles of the Green's function that may fall inside the chosen contour

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707

$$R_{i\sigma}(\omega) = \sum_j f_{j\sigma}^{i\sigma} \int \mathrm{d}\boldsymbol{r} \int \mathrm{d}\boldsymbol{r}' \psi_{i\sigma}^*(\boldsymbol{r})\psi_{j\sigma}(\boldsymbol{r}) W_p (\boldsymbol{r}, \boldsymbol{r}'; \varepsilon_{j\sigma} - \omega)\psi_{j\sigma}^*(\boldsymbol{r}')\psi_{i\sigma}(\boldsymbol{r}') \quad (8)$$

We labeled W_p the part of the screened Coulomb potential that depends on the frequency, i.e., $W_p = W_0 - v$, and we defined $f_{j\sigma}^{i\sigma} = \theta(\varepsilon_{j\sigma} - \varepsilon_F)\theta(\varepsilon_{j\sigma}^{QP} - \varepsilon_{j\sigma}) - \theta(\varepsilon_F - \varepsilon_{j\sigma})\theta(\varepsilon_{j\sigma} - \varepsilon_{j\sigma}^{QP})$, where $\theta(x)$ is the Heaviside step function and ε_F is the Fermi energy; a formal derivation may be found in ref 35.

Quasiparticle energies obtained by solving eq 5 are used to compute charged excitations and yield an electronic structure that can be compared to direct and inverse photoelectron spectroscopies (UPS, XPS, ARPES).[4,60,61]

2.2. State of the Art. First-principles calculations using the G_0W_0 method are typically carried out after DFT and are computationally more demanding than the latter: the computational complexity of the GW method scales as $O(N^4)$ with respect to the system size N, whereas DFT scales as $O(N^3)$. In addition, several computational bottlenecks hinder the applicability of the G_0W_0 method to large systems containing thousands of valence electrons (N_{occ}). In the following, we focus the discussion on implementations of G_0W_0 with three-dimensional (3D) periodic boundary conditions using the plane-wave basis set where N_ρ and N_ψ are the number of plane-waves associated with the chosen kinetic energy cutoff for the density and the wave function, respectively. In the case where ionic potentials are described using norm-conserving pseudopotentials, we have $N_\rho \simeq 8N_\psi$. A first computational bottleneck occurs when one wants to evaluate G_0 using its Lehmann representation or χ_0 using the Adler–Wiser formula,[62,63] i.e., in terms of the eigenvectors and eigenvalues of h_{KS}^σ. In this case, a summation over occupied states and empty states must be taken explicitly. The bottleneck is caused by the difficulty to fully diagonalize the KS Hamiltonian in its empty manyfold because $N_\psi \gg N_{occ}$. A second computational bottleneck occurs when one wants to evaluate W_p at multiple frequencies, and the density–density response function is represented at each frequency by a large matrix with N_ρ elements per axis. The systems discussed in this manuscript have millions of plane-waves, requiring the storage and manipulation of large matrices.

To make the simulations tractable, conventional implementations of the G_0W_0 method introduce additional parameters, e.g., $N_{empty} \ll N_\psi$ and $N_\chi \ll N_\rho$, to limit the number of empty states and the size of density–density response functions, respectively. However, these parameters, not present in the DFT calculation, show a slow convergence with respect to the size of the system. In addition, several implementations of the G_0W_0 method solve eq 5 with linearization using the on-the-mass shell approximation (i.e., Σ is evaluated at the KS energy) or approximate the frequency-dependent dielectric screening using generalized plasmon-pole models.[9,10,64−67] Such models are derived for homogeneous systems and commonly applied to heterogeneous systems without formal justification.[68] Reproducibility studies have shown that these approximations can be the source of discrepancies between different implementations.[37,69−71]

Method development aimed at improving the efficiency of full-frequency G_0W_0 calculations is the focus of current research. A few techniques have been developed in order to reduce the cost of the sum over empty states, including the extrapolar approximation,[72,73] the static remainder approach,[74] the effective energy technique,[75,76] the multipole approach,[77] and methods[35,78−87] based on density functional perturbation theory (DFPT).[88,89] The stochastic formulation of G_0W_0,[18−20] which employs the time evolution of the occupied states, leads to an implementation that does not involve empty states, and its results are comparable to those obtained with the deterministic full frequency G_0W_0 method.

Implementations of the G_0W_0 method using plane-waves basis sets are available in the following codes: ABINIT,[24] BerkeleyGW,[90] GPAW,[91] OpenAtom,[92] Quantum ESPRESSO,[30] SternheimerGW,[93] VASP,[94] WEST[35] (this work), and Yambo.[33] Other implementations use Gaussian basis sets, such as Fiesta,[95] MOLGW,[96] TURBOMOLE,[97] and VOTCA-XTP,[32] Slater-type orbitals, such as ADF,[14] numerical atomic orbitals, such as FHI-aims,[98] mixed Gaussian and plane-waves, such as CP2K,[26] linearized augmented-plane-waves with local orbitals, such as Elk,[99] Exciting,[100] and FHI-gap,[101] and real-space grids, such as NanoGW[102] and StochasticGW.[18]

In the next section, we summarize the implementation of the full-frequency G_0W_0 method in the WEST code as presented in ref 35, and we discuss the implications of design choices for calculations of large-scale system. In section 4, we present the porting of WEST to GPUs.

3. G_0W_0 IMPLEMENTATION IN THE WEST CODE

The open-source software package WEST (Without Empty STates)[35,36] implements the full-frequency G_0W_0 method for large systems using 3D periodic boundary conditions. The underlying DFT electronic structure is obtained using the plane-wave pseudopotential method. Key features of the WEST code include (i) the use of algorithms to circumvent explicit summations over empty states, (ii) the use of a low-rank decomposition of response functions to avoid storage and inversion operations on large matrices, and (iii) the use of the Lanczos technique to facilitate the calculation of the density–density response at multiple frequencies.

The complete workflow for computing QP energies with the WEST software is shown in Figure 1, where the `pwscf` code (pw.x) in the Quantum ESPRESSO software suite[30,103,104] is used to carry out the ground-state DFT calculation, the `wstat` code (wstat.x) in WEST constructs the projective dielectric eigenpotentials (PDEP) basis set that is used to obtain a low-rank representation of the density–density response function, and the `wfreq` code (wfreq.x) in WEST computes the QP energies. In the following, we describe each part of the workflow.

The ground-state electronic structure is obtained with DFT using semilocal or hybrid functionals. Although in this work we focus on spin-unpolarized and spin-polarized large systems of which the Brillouin zone can be sampled using the Γ point, the WEST software supports simulations with k-points sampling and with noncollinear spin.[40] Currently, only norm-conserving pseudopotentials are supported.

Starting from the output of DFT, the PDEP algorithm is used to find the leading eigenvectors of the symmetrized irreducible density–density response function $\overline{\chi}^0$ at zero frequency.[79,80] The eigenvectors of $\overline{\chi}^0(\omega = 0)$, referred to as the PDEP basis set, are then used to construct a low-rank decomposition of the symmetrized reducible density–density response function $\overline{\chi}$ at finite frequencies. Finally, by using the PDEP basis set one may express W_p in a separable form

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707

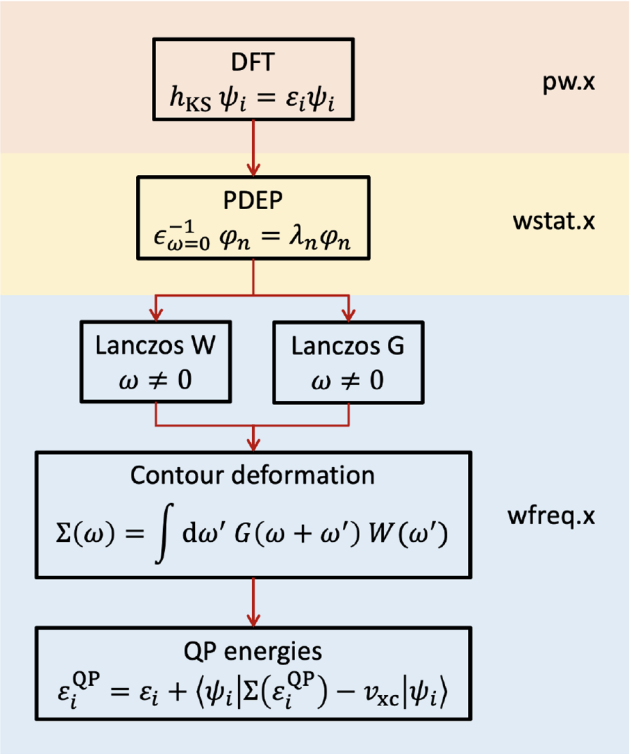


Figure 1. Key steps to compute QP energies within the G_0W_0 approximation using the WEST code. The `pwscf` code (pw.x) in Quantum ESPRESSO is employed to compute the KS wave functions and energies at the DFT level. These quantities are input to the `wstat` code (wstat.x) in WEST, which generates the PDEP basis set by iteratively diagonalizing the static dielectric matrix at zero frequency. The `wfreq` code (wfreq.x) in WEST then uses the PDEP basis set to compute G_0 and W_0 at finite frequencies with the Lanczos algorithm. Frequency integration of the self-energy in eq 4 is carried out with the contour deformation technique. Finally, the QP energies are solved using eq 5.

$$W_p(\boldsymbol{r}, \boldsymbol{r}'; \omega) = \Xi(\omega) + \frac{1}{\Omega}\sum_{nm}^{N_{PDEP}} \Lambda_{nm}(\omega)\tilde{\varphi}_n(\boldsymbol{r})\tilde{\varphi}_m(\boldsymbol{r}') \tag{9}$$

where Λ_{nm} are the matrix elements of the operator on the PDEP basis set, Ξ takes into account the frequency-dependent long-range dielectric response, Ω is the volume of the simulation cell, and $\tilde{\varphi}_n$ are symmetrized eigenpotentials, i.e., $\tilde{\varphi}_m = v^{1/2}\varphi_m$. A formal derivation may be found in ref 35.

The PDEP algorithm uses the Davidson method[105] to find the leading eigenvectors of $\overline{\chi}^0$. This is done by first constructing an orthonormal set of N_{PDEP} trial vectors $\{\varphi_j: j = 1, ..., N_{PDEP}\}$. We then repeatedly apply $\overline{\chi}^0$ to the vectors of the set and expand the set by including the residues, until the set contains N_{PDEP} leading eigenvectors of the operator. At each iteration of the Davidson algorithm, the result of the application of $\overline{\chi}^0$ on each vector of the set is obtained by computing the symmetrized density−density response of the system, $\Delta\tilde{n}_j$, subject to the symmetrized perturbation $\hat{v}_j^{pert} = \tilde{\varphi}_j$. The symmetrization operation, i.e., the multiplication by $v^{1/2}$, ensures that the response can be diagonalized and also simplifies the expression of $\overline{\chi}$ in terms of $\overline{\chi}_0$. To obtain $\overline{\chi}_0$, the response is computed using the independent particle approximation, i.e., by neglecting variations to the Hartree, exchange, and correlation potentials. In practice, the linear variation of the electron density may be computed using either linear response or a finite-field method.[106,107]

In this work, we focus on the case where the charge density response is evaluated within linear response using DFPT.[88,89] In essence, for each perturbation $\hat{v}_j^{pert} = \tilde{\varphi}_j$, we compute the linear variation, $\Delta\psi_{i\sigma}^j$, of each occupied state of the unperturbed system, $\psi_{i\sigma}$, using the Sternheimer equation[108]

$$(h_{KS}^{\sigma} - \epsilon_{i\sigma})P_c^{\sigma}\Delta\psi_{i\sigma}^j = -P_c^{\sigma}v_j^{pert}\psi_{i\sigma} \tag{10}$$

Here, P_c^{σ} is the projector onto the conduction (i.e., unoccupied) KS states. The completeness relation, i.e., $P_c^{\sigma} = 1-P_v^{\sigma}$, where P_v^{σ} is the projector onto the valence states, ensures that eq 10 can be solved without explicit summations over empty states.[89] A preconditioned conjugate gradient method is used to solve eq 10. In practice, we note that eq 10 lends itself to a nearly embarrassingly parallel implementation because it can be solved independently for each perturbation, spin channel, and orbital. Finally, the linear variation of the density caused by the jth perturbation is obtained as

$$\Delta n_j(\boldsymbol{r}) = \sum_{\sigma}\sum_{i=1}^{N_{occ}^{\sigma}} [\psi_{i\sigma}^*(\boldsymbol{r})\Delta\psi_{i\sigma}^j(\boldsymbol{r}) + \Delta\psi_{i\sigma}^{j*}(\boldsymbol{r})\psi_{i\sigma}(\boldsymbol{r})] \tag{11}$$

In this way, the calculation of the response function scales as $N_{occ}^2 \times N_{PDEP} \times N_{PW}$, which is more favorable than conventional implementations based on the Adler−Wiser formula[62,63] that scales as $N_{occ} \times N_{empty} \times N_{PW}^2$, where N_{occ} (N_{empty}) is the number of occupied (empty) states. Here, we use N_{PW} as the number of plane-waves needed to represent the wave function (previously defined as N_ψ). The PDEP basis set allows us to achieve a low-rank decomposition of density−density response matrices, reducing the size of the matrices from N_χ^2 to N_{PDEP}^2 (with $N_{PDEP} \ll N_\chi$). In practice, N_{PDEP} is the only parameter of the method, and ad hoc energy cutoffs to truncate, for instance, the response function or the number of empty states are completely sidestepped. A recent verification study[37] showed that N_{PDEP} is just a few times the number of electrons and $N_{PDEP} \ll N_{PW}$.

WEST solves the nonlinear eq 5 using a root finding algorithm, e.g., the secant method, and implements the full-frequency integration in eq 6. G_0 and W_0 are evaluated at multiple frequencies using Lanczos chains.[82,85] For instance, using eq 9 in eq 7 we obtain that

$$I_{i\sigma}(\omega) = I_{i\sigma}^{LR}(\omega) + I_{i\sigma}^{SR}(\omega) \tag{12}$$

where the long-range (LR) contribution and short-range (SR) contributions are

$$I_{i\sigma}^{LR}(\omega) = \int_{-\infty}^{+\infty}\frac{d\omega'}{2\pi}\frac{\Xi(i\omega')}{\epsilon_{i\sigma} - \omega - i\omega'} \tag{13}$$

$$I_{i\sigma}^{SR}(\omega) = \frac{1}{\Omega}\sum_{nm}^{N_{PDEP}}\int_{-\infty}^{+\infty}\frac{d\omega'}{2\pi}\Lambda_{nm}(i\omega') \langle\psi_{i\sigma}\tilde{\varphi}_n|(h_{KS}^{\sigma} - \omega - i\omega')^{-1}|\psi_{j\sigma}\tilde{\varphi}_m\rangle \tag{14}$$

The shifted-inverted problem in the RHS of eq 14 is computed introducing the Lanczos vectors

$$|q_{mi\sigma}^{l+1}\rangle = \frac{1}{\beta_{mi\sigma}^{l+1}}[(h_{KS}^{\sigma} - \alpha_{mi\sigma}^{l})|q_{mi\sigma}^{l}\rangle - \beta_{mi\sigma}^{l}|q_{mi\sigma}^{l-1}\rangle] \;\forall\; l \geq 1 \tag{15}$$

where

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707

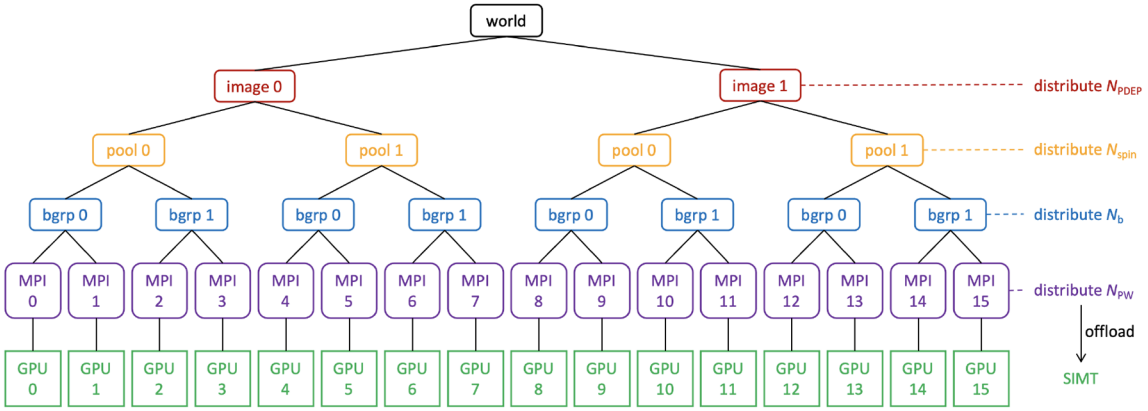


Figure 2. Multilevel parallelization of the WEST code exemplified for the case of 16 total MPI processes. The processes are divided into two images. Each image is divided into two pools, each of which is further divided into two band groups. Within each band group there are two MPI processes, each of which is capable of offloading computations to a GPU device using the single instruction, multiple threads (SIMT) protocol.

$$|q^0_{mi\sigma}\rangle = 0,\ |q^1_{mi\sigma}\rangle = |\psi_{i\sigma}\tilde{\varphi}_m\rangle \tag{16}$$

$$\alpha^l_{mi\sigma} = \langle q^l_{mi\sigma}|h^\sigma_{KS}|q^l_{mi\sigma}\rangle \tag{17}$$

$$\beta^{l+1}_{mi\sigma} = \left\|(h^\sigma_{KS} - \alpha^l_{mi\sigma})|q^l_{mi\sigma}\rangle - \beta^l_{mi\sigma}|q^{l-1}_{mi\sigma}\rangle\right\| \tag{18}$$

By defining $d^l_{mi\sigma}$ and $U^{ll'}_{mi\sigma}$ as the eigenvalues and the eigenvectors of the tridiagonal matrix that has $\alpha^l_{mi\sigma}$ along the diagonal and $\beta^l_{mi\sigma}$ along the sub- and superdiagonal, we hence arrive at the following equation

$$I^{SR}_{i\sigma}(\omega) = \frac{1}{\Omega}\sum_{nm}^{N_{PDEP}}\int_{-\infty}^{+\infty}\frac{d\omega'}{2\pi}\Lambda_{nm}(i\omega')\sum_{ll'}^{N_{Lanczos}}\langle q^1_{ni\sigma}|q^l_{mi\sigma}\rangle U^{ll'}_{mi\sigma}\frac{1}{d^{l'}_{mi\sigma} - \omega - i\omega'}U^{1l'}_{mi\sigma} \tag{19}$$

In eq 19 we see that the dependence of I^{SR} on the frequency ω is known analytically, i.e., the U and d coefficients and the integral in brakets do not depend on the frequency. This enables us to easily evaluate frequency-dependent quantities, which facilitates the solution of eq 5 without linearization, i.e., beyond the on-the-mass-shell approximation, and without using plasmon-pole models, i.e., with full frequency. Moreover, the Lanczos vectors are obtained using a recursive algorithm that orthogonalizes newly generated vectors against previous ones. Each chain of vectors can be computed individually for each perturbation, spin channel, and orbital, resulting in a nearly embarrassingly parallel implementation.

4. GPU ACCELERATION OF THE WEST CODE

In this work, we present the porting to GPUs of the WEST code, focusing on the complete full-frequency G_0W_0 workflow shown in Figure 1, including the construction of the PDEP basis set (with the standalone `wstat` application), the computation of G_0 and W_0 using the Lanczos algorithm, the integration of the self-energy using contour deformation, and the final solution of the QP energy levels (with the standalone `wfreq` application). For future reference, the CPU-only and GPU-accelerated versions of the WEST code are hereafter referred to as WEST-CPU and WEST-GPU, respectively.

To meet the challenge posed by heterogeneous computing, we increased the number of parallelization levels implemented within the code, so that we can harness the embarrassingly parallel parts of the algorithms implemented in WEST as well as the data parallelism offered by GPU devices. For instance, the PDEP algorithm may be solved using N_{proc} MPI processes and GPUs by implementing a multilevel parallelization strategy, as summarized in Figure 2. The first level of parallelization, already introduced in ref 35, divides N_{proc} processes into N_{image} subgroups, called images. Perturbations are distributed across images using a block-cyclic data distribution scheme. Each image contains a copy of the DFT data structures, such as the KS single-particle wave functions, and is responsible for computing the density response only for the perturbations owned by the image. The second and third parallelization levels, newly introduced in this work, further split the processes within an image into N_{pool} and N_{bgrp} subgroups, called pools and band groups, respectively. Each pool and band group stores and manipulates only a subset of the wave functions by distributing the spin polarization (for spin-polarized systems only) and band indices, respectively. The remaining $N_{proc}/(N_{image}N_{pool}N_{bgrp})$ processes within a band group distribute the plane-wave coefficients of wave functions and densities, forming the fourth level of parallelization. Finally, each MPI process is capable of offloading instructions to one GPU, which offers single instruction, multiple thread (SIMT) parallelization by leveraging the processing cores within the GPU device.

This flexible parallelization scheme helps optimize MPI communications as well as fit hardware constraints (e.g., number of GPUs within one node). Global MPI communications involving all MPI processes are avoided, except for the broadcast of the input parameters (a few scalars) from one MPI process to the others.

For offloading data-parallel regions to the GPU, we specifically focused this initial study to target NVIDIA devices. Wherever applicable, mathematical operations are performed on the GPU by calling optimized CUDA libraries, such as cuFFT for fast Fourier transforms and cuBLAS for matrix−matrix multiplications and other basic linear algebra operations. If a compute loop cannot be organized to use an existing library function, the loop is offloaded using CUDA Fortran kernel directives, which automatically generate CUDA kernels from regions of annotated CPU code.[109] In order to avoid the performance degradation caused by frequent data transfer operations between the CPUs and the GPUs, WEST-GPU copies the necessary data from the CPU to the GPU at

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707

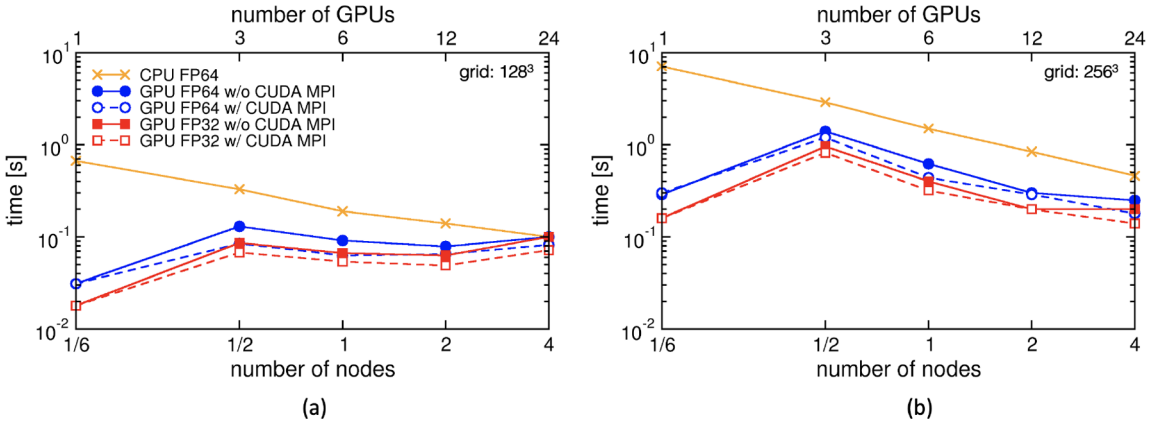


Figure 3. Time required to carry out a complex-to-complex FFT operation on a 128^3 grid (a) or on a 256^3 grid (b) using the FFTXlib library on Summit (each node has six GPUs, see also the specification listed in section 5). Blue circles and red squares identify the execution on GPUs using the cuFFT backend with double precision (FP64) or single precision (FP32), respectively. Timing results do not include the time needed to initially copy the data to the GPU and the time to copy the final result back to the CPU. Orange crosses identify the execution on CPUs using the ESSL backend with FP64. Filled symbols identify the use of CUDA-aware MPI and GPUDirect. Open symbols identify the use of conventional, non-CUDA-aware MPI. The slab decomposition scheme was employed for parallel FFTs.

the very beginning of a calculation. The data is copied back to the CPU only when absolutely necessary, e.g., for input/output (I/O) operations.

Work is underway to extend the current implementation to other GPU devices as more software and hardware become available. We anticipate that the multilevel parallelization strategy introduced so far will grant flexibility of distributing the computational workload also on GPU devices other than NVIDIA ones. However, a discussion of the performance portability and how it may be achieved by translating CUDA Fortran into OpenMP directives[110] goes beyond the scope of this manuscript.

In the next subsections, we elaborate on specific optimization strategies introduced in WEST-GPU on top of the multilevel parallelization. In section 4.1, we point out key factors that maximize the performance of GPU-accelerated fast Fourier transforms (FFTs). In section 4.2, we benchmark various eigensolver libraries, identifying the most efficient solver for diagonalizing large matrices on multiple GPUs. In section 4.3, we demonstrate that the overhead of MPI communications can be diminished by overlapping communications with computations.

4.1. Fast Fourier Transforms. The performance of FFTs is of crucial importance to the overall efficiency of any plane-wave-based electronic structure code. FFTs are extensively used in WEST to express quantities such as wave functions, densities, and perturbations in either the reciprocal or the direct space. Most importantly, the application of the KS Hamiltonian to a trial wave function, a key step for the calculation of G_0 and W_0 without explicit summations over empty states, is implemented using the dual-space technique, i.e., the kinetic operator and the local potential are applied in the reciprocal or direct space, respectively. The dual-space technique takes advantage of the convolution theorem and the fact that FFTs scale as $O(N \log(N))$. It follows that at least two FFTs (one forward and one backward) are required at every application of the KS Hamiltonian, and their performance greatly impacts the overall time-to-solution of both `wstat` and `wfreq` (see Figure 1). FFTs are also invoked in other parts of the code, for example, to obtain the electron density.

WEST uses the FFTXlib library to implement parallel 3D FFTs. This library retains only the Fourier components that correspond to a chosen kinetic energy cutoff and is part of the Quantum ESPRESSO distribution.[30] FFTXlib may perform a 3D FFT using one MPI process or using several MPI processes by decomposing the 3D grid into slabs or pencils. The slab decomposition partitions the 3D grid into slabs, completing a 3D FFT by a set of 2D FFTs, an all-to-all communication, and a set of one-dimensional (1D) FFTs. The pencil decomposition partitions the 3D grid into pencils, completing a 3D FFT as a set of 1D FFTs, an all-to-all communication, another set of 1D FFTs, another all-to-all communication, and a final set of 1D FFTs.[111] When multiple MPI processes are used, the Fourier components are distributed among the processes avoiding data duplication. 3D, 2D, or 1D FFTs on a single MPI process are performed using vendor-optimized libraries. FFTXlib supports a variety of backends, currently including FFTW3, Intel MKL, and IBM ESSL for CPUs and cuFFT for NVIDIA GPUs.

We benchmarked the performance of the FFTXlib library on the Summit supercomputer located at Oak Ridge National Laboratory. Each node of Summit has two IBM POWER9 CPUs (21 cores each) and six NVIDIA V100 GPUs (see also the specification of Summit listed in section 5). We used FFTXlib (version 6.8) with IBM ESSL (version 6.3.0) for the CPU backend and cuFFT (version 10.2.1.245) for the GPU backend. In Figure 3, we report the time needed to perform one double-precision (FP64) or single-precision (FP32) complex-to-complex (C2C) FFT for a 128^3 or 256^3 cubic grid using up to four nodes of Summit. Each data point in Figure 3 represents the average value of 100 tests. When the FFT is GPU accelerated, we used one MPI process per GPU. In WEST-GPU, data is preallocated on the GPU so that GPU-enabled FFT operations act on data that resides on the GPU. The majority of the data is therefore initialized on the GPU at the beginning of the calculation with a CPU-to-GPU copy, then the data undergoes multiple FFT operations, and finally the data is copied back to the CPU. Hence, because data transfer operations are decoupled from FFTs, Figure 3 does not include the time needed to initially copy the data to the GPU and the time to copy the final result back to the CPU.

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707

Table 1. Time to Solve Real Symmetric Eigenproblems on Summit Using Eigensolvers in the ESSL (multithreaded), ScaLAPACK (MPI-parallel), cuSOLVER (GPU-accelerated), and ELPA (GPU-accelerated, MPI-parallel) Libraries[a]

	time [s]			
matrix size	ESSL (42 CPU cores)	ScaLAPACK (336 CPU cores)	cuSOLVER (1 GPU)	ELPA (48 GPUs)
$10\,000^2$	129.1	48.1	3.2	3.3
$20\,000^2$	1050.4	350.6	22.6	8.5
$30\,000^2$	3483.4	1012.2	OOM	18.4
$40\,000^2$	7811.5	2181.0	OOM	30.8

[a]Matrix size n^2 corresponds to a square matrix with n rows and n columns. "OOM" (out of memory) indicates failed cuSOLVER executions due to insufficient device memory.

We can see that one GPU (corresponding to 1/6 of a Summit node in Figure 3) outperforms one CPU (21 cores, 1/2 of a node) by more than an order of magnitude. However, while the time-to-solution on CPUs decreases linearly by increasing the number of nodes, one GPU is still faster than any other number of GPUs. Using three GPUs, for instance, slows down the calculation by nearly an order of magnitude compared to using only one GPU. Further increasing the number of GPUs leads to a moderate speedup, especially for the 256^3 grid. However, even 24 GPUs (four nodes) cannot outperform one GPU; this is consistent with previously published benchmarks on the same machine[111] and clearly reveals the high cost of communications (relative to the computation) involved in parallel distributed FFTs. As expected, using FP32 instead of FP64, FFT operations achieve a ∼2× speedup on one GPU (red lines in Figure 3). The speedup gradually decreases as the GPU count increases, which may be attributed to the increasingly high MPI overhead relative to the small amount of computation being performed. From Figure 3, we conclude that FFTs should be carried out on as few GPUs as possible, ideally only one GPU, for maximum efficiency.

We note that the slab decomposition is used throughout this manuscript. The implementation of 3D FFTs with pencil decomposition is ∼30% slower than the slab decomposition for the system sizes considered in Figure 3. In cases where one is interested in using more MPI processes than the number of slabs, the pencil decomposition will potentially become advantageous as it enables more parallelism than the slab decomposition does. This case is, however, unlikely to be relevant as Figure 3 suggests that the least number of GPUs shall be used to perform FFTs due to the overhead of all-to-all communications compared to the cost of computation.

It is worth mentioning that for denser cubic grids, the overhead associated with all-to-all communications may become negligible compared to the amount of computation that needs to be performed on the GPU. A nearly ideal strong scaling of GPU-accelerated FFTs using the heFFTe library has been reported for a grid size of 1024^3.[111] However, the large-scale applications reported within this manuscript are performed using grids with up to 216 points per axis. Tests using the heFFTe library for smaller grids, such as 128^3 and 256^3, reveal performance characteristics that are similar to those of FFTXlib.

The multilevel parallelization introduced in Figure 2 is key to run WEST with as many GPUs as possible while performing FFTs using the least amount of GPUs. Specifically, the FFTs in WEST-GPU are carried out using $N_{proc}/(N_{image}N_{pool}N_{bgrp})$ MPI processes instead of all of the N_{proc} processes. In practice, N_{image}, N_{pool}, and N_{bgrp} are chosen to restrict FFTs to the smallest number of GPUs, so that the total GPU memory is sufficiently large to accommodate the simulation data.

In the case where FFT operations involve more than one GPU, Figure 3 shows that a performance gain can be obtained by taking advantage of CUDA-aware MPI and GPUDirect. Without CUDA-aware MPI, data residing on the GPU must be explicitly copied to the host CPU in order to participate in an MPI communication. If the data is needed by the GPU after the MPI communication, the data must be explicitly copied back. With CUDA-aware MPI, data on the GPU can be directly passed to MPI functions. However, depending on the hardware and software settings, the data may still be communicated through the CPU. The GPUDirect technology enhances data movement between NVIDIA GPUs. Specifically, for GPUs directly connected with each other through NVLink,[112] the data transfer takes advantage of the high bandwidth of NVLink without going through the CPU; similarly, for internode communications, GPU data can be directly put onto the node interconnect. In Figure 3, the dashed lines correspond to FFTs employing CUDA-aware MPI and GPUDirect. For the grid sizes considered in Figure 3, switching on CUDA-aware MPI and GPUDirect results in a performance improvement ranging from 20% to 50%.

4.2. Solution of Large Eigenvalue Problems. As introduced in section 3, WEST relies on the Davidson algorithm[105] to iteratively diagonalize the irreducible density−density response function. In each iteration, a Hermitian matrix needs to be explicitly diagonalized. The dimension of the matrix is proportional to N_{PDEP}, and by default, matrices up to $(4N_{PDEP})^2$ are diagonalized. WEST-CPU is capable of treating systems containing a few thousand electrons, leading to eigenvalue problems as large as $10\,000^2$. Solving such eigenvalue problems by serial or multithreaded solvers from the LAPACK library accounts for a negligible fraction of the total computational cost of WEST-CPU. For WEST-GPU, given that the most compute-intensive operations have been moved to GPUs, the eigenvalue problem stands out as roadblock that limits the performance of the code for large systems with 10 000s of electrons. For instance, the largest GW calculation reported in section 6 has $N_{PDEP} = 10\,368$, requesting the diagonalization of matrices up to $(4 \times 10\,368)^2 = 41\,472^2$. Solving such large eigenvalue problems on CPUs takes a significant amount of time (see Table 1).

To circumvent this bottleneck, we compared the performance on CPUs and GPUs of four eigensolvers on Summit, namely, the multithreaded LAPACK implementation in the IBM ESSL library (version 6.3.0), the MPI-parallel and memory-distributed eigensolver in the ScaLAPACK library (version 2.1.0), the GPU-accelerated eigensolver in the cuSOLVER library (version 10.6.0.245), and the MPI-parallel, memory-distributed, GPU-accelerated eigensolver in the ELPA library (version 2020.11.001).[113] ESSL and ScaLAPACK used one node (one MPI process, 42 OpenMP threads) and eight

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707

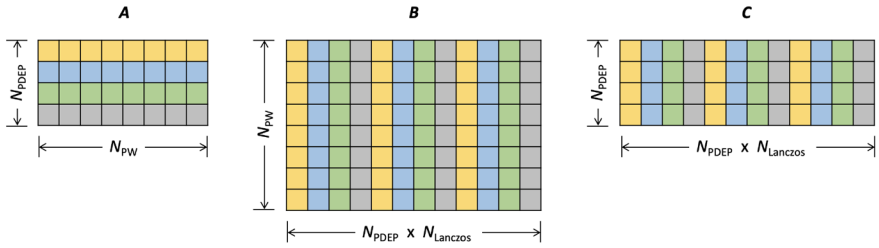


Figure 4. Schematic visualization of the data distribution in the multiplication ($\boldsymbol{C} = \boldsymbol{A} \times \boldsymbol{B}$) of the two distributed matrices discussed in the text. N_{PW} and N_{PDEP} denote the numbers of plane-waves and PDEPs, respectively. $N_{Lanczos}$ denotes the length of the Lanczos chain. $N_{Lanczos} = 30$ typically yields converged results. Data is color coded so that the data owned by different MPI processes have different colors.

nodes (42 MPI processes per node), respectively. cuSOLVER and ELPA used one NVIDIA V100 GPU and eight nodes (six MPI processes per node, totaling 48 CPU cores and 48 GPUs), respectively. Table 1 shows the performance of each eigensolver for matrix sizes ranging from $10\,000^2$ to $40\,000^2$. Using only one GPU, cuSOLVER exhibits a significant speedup over both ESSL and ScaLAPACK for matrix sizes up to $20\,000^2$. For larger matrix sizes, however, the available GPU memory (16 GB for the V100 GPU on Summit) can no longer accommodate the matrix and the workspace required by cuSOLVER. In such scenarios, the memory-distributed, GPU-accelerated ELPA eigensolver provides the fastest time to solution at a relatively low memory cost per GPU. On the basis of these benchmarks, WEST-GPU uses cuSOLVER to diagonalize matrices smaller than 8000^2 and switches to GPU-accelerated ELPA for larger matrices.

Other multi-GPU eigensolvers, not considered in this work, include, for instance, SLATE[114] and cuSOLVER-MG.[115] At present, SLATE is limited to compute the eigenvalues only. The commonly used 2D block-cyclic matrix distribution is not yet supported in cuSOLVER-MG, which only supports 1D block-cyclic distribution. We plan to continue assessing the performance and compatibility of these libraries as they evolve.

4.3. Overlapping Computation and Communication. Communication overheads are reduced using nonblocking MPI functions to overlap computation and communication. Nonblocking MPI functions immediately return control to the host even if the communication has not been completed; in this way, the host is allowed to perform other operations while the communication continues in the background. When using GPUs, MPI communications can be overlapped with GPU computations and CPU−GPU communications.

Nonblocking MPI calls are extensively utilized in WEST-GPU. One example is the calculation of the braket integral in the RHS of eq 19. This term may be evaluated for the ith orbital in the σ spin polarization as the matrix−matrix multiplication $\boldsymbol{C} = \boldsymbol{A} \times \boldsymbol{B}$ depicted in Figure 4, where $\boldsymbol{A}_{nk}$ is the kth coefficient of the Fourier expansion of the product $\psi_{i\sigma}(\mathbf{r})\tilde{\varphi}_n(\mathbf{r})$ and $\boldsymbol{B}_{kml}$ is the kth coefficient of the Fourier expansion of $q^l_{mi\sigma}(\mathbf{r})$. According to Figure 2, the indices n and m are distributed using the image parallelization, whereas the Fourier coefficients are distributed using the MPI processes within one band group. This distribution can lead to tall-and-skinny matrices on each process, i.e., one of the dimensions is significantly greater than the other. The multiplication of tall-and-skinny matrices is memory bound. It performs poorly on both CPUs and GPUs, which is a well-known outstanding problem. The flexible multilevel parallelization scheme reported in Figure 2 allows us to tune the shape of the local matrices, which is implemented by carefully choosing the number of images and band groups. As a result, tall-and-skinny local matrices can be avoided, pushing the matrix multiplication into the compute-bound regime and therefore achieving better performance.

The distributed matrix multiplication is completed as follows. First, each MPI process computes the product of its local portion of $\boldsymbol{A}$ and $\boldsymbol{B}$, contributing to a portion of $\boldsymbol{C}$. Second, the ith MPI process sends its local portion of $\boldsymbol{A}$ to process $(i - 1)$ and receives another portion of $\boldsymbol{A}$ from process $(i + 1)$. This communication pattern is known as a circular shift. There is no need to communicate $\boldsymbol{B}$ or $\boldsymbol{C}$. These steps are repeated until all elements of $\boldsymbol{C}$ are obtained.

The pseudocode in Figure 5a is a straightforward GPU implementation of the above procedure, which includes three sequential steps, copying $\boldsymbol{A}$ to the GPU, computing the local matrix multiplication on the GPU, and communicating $\boldsymbol{A}$ via MPI. The right part of Figure 5a shows the timeline corresponding to the pseudocode. Using nonblocking MPI functions, MPI communications can be overlapped with other operations. As shown in Figure 5b, while the GPU is computing a portion of $\boldsymbol{C} = \boldsymbol{A} \times \boldsymbol{B}$, MPI communications take place asynchronously in the background to prepare the next portion of $\boldsymbol{A}$. As such, the cost of the CPU−GPU data transfer and the GPU matrix multiplication can be hidden behind the more expensive MPI communication, as the timeline in Figure 5 indicates. In practice, this optimization leads to a speedup of 15−30%.

The matrix−matrix multiplication operation in Figure 5 can be further accelerated by performing MPI communications in single precision instead of double precision. The pseudocode of our implementation is reported in Figure 5c, where $\boldsymbol{A}$ is truncated from FP64 to FP32, communicated in FP32, then cast back to FP64, and multiplied with $\boldsymbol{B}$. The precision conversions and matrix multiplications take place on the GPU, and the MPI communications are launched asynchronously to overlap with the CPU−GPU data transfers, data conversions, and matrix multiplications, as indicated by the timeline in the right part of Figure 5c. The QP energies obtained using FP64 are in good agreement with the results obtained using mixed precision (FP32/FP64).

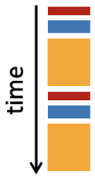
5. PERFORMANCE OF WEST-GPU

We report an assessment of the performance of WEST-GPU over WEST-CPU and of its strong and weak scaling using leadership HPC systems. Our benchmarks are carried out on the Summit supercomputer at Oak Ridge National Laboratory, the Perlmutter supercomputer[116] at the National Energy Research Scientific Computing Center, and the Theta

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707

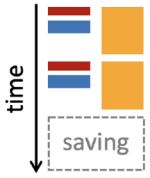
(a)

```
do i = 1, n_mpi
   copy a_h to a_d
   compute c_d(i) = a_d * b_d
   circular shift a_h
end do
```



(b)

```
do i = 1, n_mpi
   async circular shift a_h
   copy a_h to a_d
   compute c_d(i) = a_d * b_d
end do
```



(c)

```
convert a_h to a_h_sp
do i = 1, n_mpi
   async circular shift a_h_sp
   copy a_h_sp to a_d_sp
   convert a_d_sp to a_d
   compute c_d(i) = a_d * b_d
end do
```

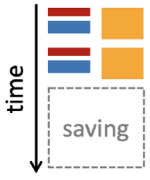


Figure 5. Pseudocodes representing three alternative strategies to multiply the two distributed matrices discussed in the text. (a) No overlap between computation and communication. (b) MPI communication, CPU−GPU communication, and GPU computation are overlapped. (c) MPI communication, CPU−GPU communication, and GPU computation are overlapped, with MPI communication carried out in single precision (FP32). Suffixes "_h" and "_d" indicate arrays allocated on the host (CPU) and the device (GPU), respectively. Suffix "_sp" indicates a single-precision copy of a double-precision array. CPU−GPU communications, MPI communications, and GPU computation are reported in red, yellow, and blue, respectively.

supercomputer at Argonne National Laboratory. While the nodes of the first two supercomputers have GPUs, the nodes of the latter have only CPUs. The specifications of these computers are listed in Table 2.

We conduct benchmarks of the two standalone parts of the WEST code, namely `wstat` and `wfreq`, which compute the static dielectric matrix and the full-frequency G_0W_0 self-energy, respectively (see section 3). Quantum ESPRESSO 6.8 is used

Table 2. Specifications (per node) of the Summit, Perlmutter, and Theta Supercomputers[a]

	Summit	Perlmutter	Theta
CPU	2 × IBM POWER9 (2 × 21 cores)	1 × AMD EPYC Milan (64 cores)	1 × Intel Knights Landing (64 cores)
GPU	6 × NVIDIA V100	4 × NVIDIA A100	none
TFLOP/s	43.5	39.0	2.7
compiler	nvFortran 21.7	nvFortran 21.7	ifort 19.1.0.166
	IBM Spectrum MPI 10.4	Cray MPICH 8.1.9	Cray MPICH 7.7.14
libraries	NVIDIA HPC SDK 21.7	NVIDIA HPC SDK 21.7	Intel MKL 2020 initial release
	CUDA 11.0.3	CUDA 11.0.3	

[a]Theoretical peak performance (TFLOP/s) is reported for double precision. The Fortran compiler and performance-critical libraries used in the benchmark calculations are also listed.

for all ground-state DFT calculations. We use the SG15[117] optimized norm-conserving Vanderbilt (ONCV) pseudopotentials[118] and the PBE exchange-correlation functional.[119] As the test systems considered here are either isolated structures or large cells of periodic structures, the Brillouin zone is sampled only at the Γ point. In `wstat`, the size of the PDEP basis set is set equal to the number of electrons in the system. In `wfreq`, we compute the full-frequency G_0W_0 self-energy for a variable number of states.

Benchmarks conducted on Summit and Perlmutter use CUDA-aware MPI and GPUDirect. As discussed in section 4.1, these technologies facilitate the data exchange between GPUs. Timing results reported in this section correspond to the total wall clock time, including the time spent on I/O operations and CPU−GPU communications.

5.1. Performance of WEST-GPU Relative to WEST-CPU. We compare the performance of WEST-GPU relative to the performance of WEST-CPU considering three benchmark systems: a negatively charged nitrogen-vacancy center in diamond with 215 atoms (NV_DIA),[54] a $Cd_{34}Se_{34}$ nanoparticle (CdSe_NP), and a COOH-Si/H_2O solid/liquid interface consisting of a total of 492 atoms (S/L)[35,120] (see Figure 6). Details about each system are summarized in Table

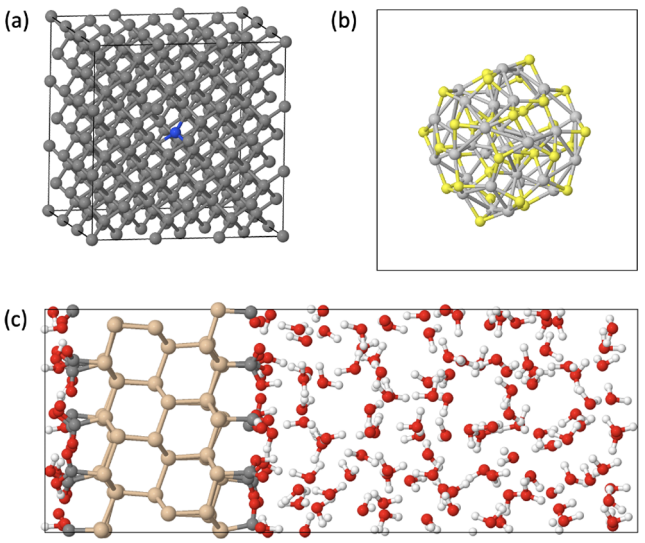


Figure 6. Benchmark systems: (a) negatively charged nitrogen-vacancy center in diamond (NV_DIA), (b) cadmium selenide nanoparticle (CdSe_NP), and (c) COOH-Si/H_2O solid/liquid interface (S/L). For the ball-and-stick atomic structures, the C, N, Cd, Se, Si, O, and H atoms are colored in dark gray, blue, yellow, light gray, beige, red, and white, respectively. Details of the systems are reported in Table 3.

3. Because the peak performance of one node of Summit or of Perlmutter is considerably higher than the peak performance of one node of Theta (see Table 2), we benchmark WEST-GPU using 16 Summit or 16 Perlmutter nodes against the performance of WEST-CPU using 256 Theta nodes to have similar total peak performances (∼696 TFLOP/s on 16 Summit nodes, ∼624 TFLOP/s on 16 Perlmutter nodes, ∼691 TFLOP/s on 256 Theta nodes).

Figure 7a and 7b shows the performance of the GPU-accelerated `wstat` and `wfreq` parts of WEST, respectively. Using only double precision (FP64 in Figure 7), WEST-GPU on 16 Summit nodes outperforms WEST-CPU on 256 Theta

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707

Table 3. Simulation Parameters for the Systems Depicted in Figure 6[a]

system	N_{atom}	$N_{electron}$	N_{spin}	E_{cut} [Ry]	N_{PW}
NV_DIA	215	862	2	60	64 973
CdSe_NP	68	884	1	50	382 323
S/L	492	1560	1	60	295 387

[a] N_{atom}, $N_{electron}$, N_{spin}, and N_{PW} denote the numbers of atoms, electrons, spin channels, and plane-waves, respectively. E_{cut} denotes the cutoff energy used in the calculations.

nodes by a factor of 2.0−2.2×. This imputes an effective 32−35× speedup for one Summit node over one Theta node, which is higher than the value of 16× estimated by taking the ratio between the two theoretical peak performances. This may be attributed to two factors: (i) the higher node count on Theta than on Summit, which generates more internode communication, and (ii) the use of GPUs on Summit to carry out FFTs, which are notoriously memory-bound operations and therefore benefit from the higher memory bandwidth of the GPU (900 GB/s in V100 GPUs, whereas each KNL node on Theta has 16 GB fast memory with a bandwidth of 400 GB/s). By running WEST-GPU on 16 Perlmutter nodes we observe an additional 30−40% speedup over 16 Summit nodes. This is caused by the fact that FFTs are carried out using one MPI process (one GPU) on Perlmutter, while on Summit we are forced to use three MPI processes (three GPUs) due to the memory limitation (40 GB in A100 GPUs, 16 GB in V100 GPUs). FFTs in the latter case incur the overheads described in section 4.1. Moreover, the A100 GPU has a higher memory bandwidth (1555 GB/s) than the V100 GPU and features FP64 tensor cores that can be automatically utilized by the CUDA libraries wherever possible. On the contrary, the V100 GPU features tensor cores only for half precision, which are not utilized by the current version of WEST. Similar conclusions are drawn analyzing the FP64 performance of `wfreq`, where we computed 40 QP energies (around the Fermi level, 20 below and 20 above) for each system.

Figure 7 also reports the performance of the mixed-precision (FP32/FP64) version of WEST-GPU. In the case of mixed precision, the code operates in FP64 except for the regions of the code with distributed matrix multiplication or FFTs, which are carried out using FP32, as discussed in section 4. The FP32/FP64 code outperforms the FP64 counterpart on both Summit and Perlmutter by up to 45%, due to a nearly two-fold speedup in the corresponding FFT and MPI operations. It is important to note that the QP energies obtained using the FP32/FP64 code are in good agreement with the results obtained with FP64. The mean absolute error of the 40 QP energies computed with the FP32/FP64 code lies well below 10^{-4} eV for the three systems studied here, justifying the utilization of mixed precision in production calculations. For all calculations reported in sections 5.2 and 6, the FP32/FP64 version of WEST was employed.

Table 4 reports the performance of WEST-GPU in terms of FLOP/s, computed as the ratio of the total number of FLOPs to the total time of the simulation. FLOPs are counted by inserting counters into the source code. This approach comes with a lower overhead than using external profiling tools. Nevertheless, we measure the FLOPs and the time in two separate runs in order to obtain accurate timing results. The performance of WEST-GPU is compared against the theoretical peak performance of Summit and Perlmutter. For both `wstat` and `wfreq`, a higher fraction of the theoretical peak was reached on Perlmutter, ranging from 36.0% to 72.9%, than on Summit, ranging from 23.6% to 49.7%. GPUs are better utilized for larger systems, as the workload associated with larger systems is more likely to saturate the GPUs. We note that the ratio to peak performance shown in Table 4 is used to approximately estimate how efficiently the GPUs are being utilized. In WEST-GPU, the FFTs benefit from the use of FP32 and the matrix−matrix multiplications and possibly other linear algebra operations benefit from the FP64 tensor cores on Perlmutter. These factors are not reflected in Table 4.

5.2. Strong and Weak Scaling of WEST-GPU. We report the strong and weak scaling of WEST-GPU as benchmarked on the Summit supercomputer with a series of silicon supercell models with up to 1728 atoms, as described in Table 5. The strong scaling of WEST-GPU is presented in Figure 8a for 2 Si supercells containing 1000 or 1728 atoms. The weak scaling is

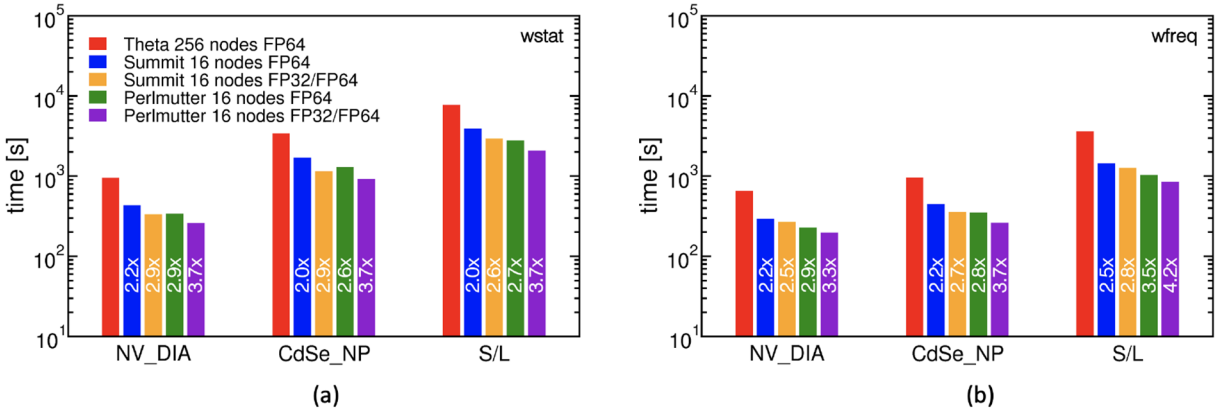


Figure 7. Performance of `wstat` (a) or `wfreq` (b) using 16 nodes of Summit, 16 nodes of Perlmutter, or 256 nodes of Theta. Performance (reported within the bars) is obtained by taking the ratio of the time measured for WEST-GPU to the time measured for WEST-CPU. `wstat` and `wfreq` codes are described in section 3. Timing results correspond to the total wall clock time, including the time spent on I/O operations and CPU−GPU communications. Forty quasiparticle energies (around the Fermi level, 20 below and 20 above) of the atomic structures in Figure 6 were calculated. Calculations on Theta used the double-precision (FP64) WEST-CPU code. Calculations on Summit and Perlmutter used the FP64 and mixed-precision (FP32/FP64) WEST-GPU code (see text).

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707

Table 4. Performance (TFLOP/s) of WEST-GPU on 16 Nodes of Summit and Perlmutter for the Systems Described in Table 3[a]

			Summit			Perlmutter		
code	system	N_{FLOP} [TFLOPs]	time [s]	perf. [TFLOP/s]	% peak	time [s]	perf. [TFLOP/s]	% peak
wstat	NV_DIA	9.21×10^4	332.0	277.5	39.9	260.1	354.3	56.8
	CdSe_NP	3.96×10^5	1147.0	345.6	49.7	919.2	431.2	69.1
	S/L	9.42×10^5	2937.1	320.6	47.4	2070.9	454.8	72.9
wfreq	NV_DIA	4.41×10^4	267.8	164.8	23.6	196.4	224.7	36.0
	CdSe_NP	1.18×10^5	354.7	333.5	47.9	262.7	450.1	70.1
	S/L	3.63×10^5	1269.1	286.0	41.1	851.8	426.1	68.2

[a]The wstat and wfreq codes are described in section 3. Timing results correspond to the total wall clock time, including the time spent on I/O operations and CPU−GPU communications. Performance (perf.) is measured as the ratio of the total number of FLOPs (N_{FLOP}) to the total time of the simulation. "% Peak" denotes the ratio of the measured performance to the theoretical peak performance; the latter is calculated as $N_{node} \times$ 43.5 TFLOP/s and $N_{node} \times$ 39.0 TFLOP/s for the Summit and Perlmutter computers, respectively.

Table 5. Simulation Parameters of the Silicon Supercells Used As Benchmarks[a]

supercell	N_{atom}	$N_{electron}$	N_{spin}	E_{cut} [Ry]	N_{PW}
$3 \times 3 \times 3$	216	864	1	16	31 463
$4 \times 4 \times 4$	512	2048	1	16	74 773
$5 \times 5 \times 5$	1000	4000	1	16	145 837
$6 \times 6 \times 6$	1728	6912	1	16	251 991

[a]Supercells are obtained by considering replicas of the eight-atom conventional unit cell. N_{atom}, $N_{electron}$, N_{spin}, and N_{PW} denote the numbers of atoms, electrons, spin channels, and plane-waves, respectively. E_{cut} denotes the cutoff energy used in the calculations.

presented in Figure 8b for 4 Si supercells containing 216, 512, 1000, and 1728 atoms. Eighty QP energies (around the Fermi level, 40 below and 40 above) were calculated for each system. Strong and weak scaling close to the ideal one (dashed lines) is observed for both wstat and wfreq. The 1728-atom system exhibits a strong scaling closer to the ideal one than that of the 1000-atom system. This stems from the higher computation-to-communication ratio of the larger system, and it demonstrates the applicability of WEST-GPU to large-scale simulations.

In Table 6 we report an estimate of the performance of WEST-GPU by measuring the total number of floating-point operations recorded for running wstat and wfreq and dividing it by the total time, including the time to carry out I/O operations. For the 1000-atom silicon supercell, wstat reaches 47.3% and 16.6% of the theoretical peak performance on 250 and 4000 Summit nodes, respectively. Internode MPI communications are responsible for the drop in the performance at a large number of nodes. When we increase the size of the system to comprise 1728 silicon atoms, wstat reaches 42.5% of the peak on 864 nodes and sustains 31.2% of the peak even on 4320 nodes (25 920 V100 GPUs), amounting to a mixed-precision performance of 58.80 PFLOP/s. The performance of wfreq is slightly inferior to that of wstat due to the larger amount of internode MPI communications in wfreq. Nevertheless, for the 1728-atom silicon supercell, wfreq achieves a mixed-precision performance of 35.88 PFLOP/s on 4320 nodes, corresponding to 19.1% of the peak. In all cases, we observe that the full applications (including I/O operations) scale to the entire Summit machine. We note again that the ratio to peak performance is discussed for a qualitative understanding of how the GPUs are utilized by WEST-GPU. It does not take into consideration that WEST-

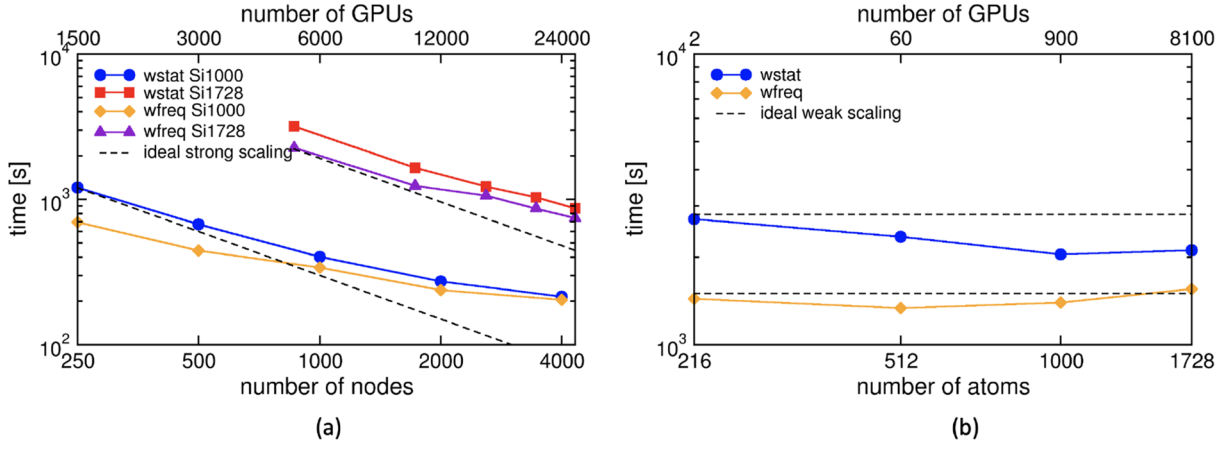


Figure 8. Strong (a) and weak (b) scaling of WEST-GPU. The wstat and wfreq codes are described in section 3. (a) Blue circles (yellow diamonds) represent the strong scaling of wstat (wfreq) for the 1000-atom silicon supercell reported in Table 5; red squares (violet triangles) represent the strong scaling of wstat (wfreq) for a 1728-atom silicon supercell reported in Table 5. (b) Blue circles (yellow diamonds) represent the weak scaling of wstat (wfreq). Black dashed lines indicate the slope of ideal scaling. Eighty quasiparticle energies (around the Fermi level, 40 below and 40 above) were calculated for each system. Timing results correspond to the total wall clock time, including the time spent on I/O operations and CPU−GPU communications.

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707

Table 6. Performance (PFLOP/s) of WEST-GPU on Summit for the 1000 and 1728 Silicon Atoms Supercell Models Described in Table 5[a]

code	N_{atom}	N_{FLOP} [PFLOPs]	N_{node}	time [s]	perf. [PFLOP/s]	% peak
wstat	1000	6.20×10^3	250	1204.6	5.15	47.3
			4000	214.1	28.95	16.6
	1728	5.09×10^4	864	3182.7	16.01	42.5
			1728	1648.4	30.89	41.1
			2592	1226.5	41.52	36.8
			3456	1030.8	49.40	32.9
			4320	866.0	58.80	31.2
wfreq	1000	2.95×10^3	250	695.7	4.24	39.0
			4000	203.5	14.50	8.3
	1728	2.67×10^4	864	2259.0	11.82	31.4
			1728	1239.1	21.55	28.7
			2592	1062.0	25.14	22.3
			3456	864.7	30.88	20.5
			4320	744.1	35.88	19.1

[a]The wstat and wfreq codes are described in section 3. Timing results correspond to the total wall clock time, including the time spent on I/O operations and CPU−GPU communications. The performance (perf.) is measured as the ratio of the total number of FLOPs (N_{FLOP}) to the total time of the simulation. N_{node} denotes the number of Summit nodes used in the calculations (each node has six GPUs, see Table 2). % peak denotes the ratio of the measured performance to the theoretical peak performance; the latter is calculated as $N_{node} \times 43.5$ TFLOP/s.

GPU carries out FFTs and MPI communications in single precision.

For the system with 1000 silicon atoms, the FLOP count (N_{FLOP}) required to compute the quasiparticle energy for N_{QP} bands using wfreq is $N_{FLOP} = (2553 + 8.17 \times N_{QP})$ PFLOPs. The prefactor (2553 PFLOPs) identifies the FLOPs required to compute the dielectric screening at all frequencies and without empty states using the PDEP basis set, while the multiplicative factor (8.17 PFLOPs) is attributed to the cost of computing the full-frequency G_0W_0 self-energy for one band. The FLOP count indicates that it becomes cost effective to compute the self-energy for many states; this is convenient, for instance, for the simulation of photoelectron spectra over an extended region of energies.[39]

Finally, we note that the FLOP count in Table 6 indicates that a few EFLOPs are necessary in order to compute the full-frequency G_0W_0 electronic structure of both benchmark systems. At the measured sustained 30−60 PFLOP/s throughput, the calculations can be carried out within tens of minutes. We also note that the current results are obtained with an implementation that, in addition to avoiding approximating the screened Coulomb interaction with generalized plasmon-pole models, sidesteps altogether the need to compute many empty states using DFT and the need to introduce a stringent energy cutoff in reciprocal space to represent dielectric matrices.

6. LARGE-SCALE FULL-FREQUENCY G_0W_0 CALCULATIONS

Finally, we demonstrate the capability of WEST-GPU for computing the full-frequency G_0W_0 electronic structure of large-scale systems. The structures shown in Figure 9 are representative examples of large heterogeneous systems of interest for energy sustainability and quantum information science research. The structure in Figure 9a is a Janus nanoparticle (CdS/PbS) consisting of 301 atoms and 2816 electrons. In this system, investigated for its applicability to photovoltaics,[121] we compute with G_0W_0 the band offsets between the cadmium sulfide (CdS) and the lead sulfide (PbS) hemispheres of the heterostructured nanoparticle. The structure in Figure 9b is an interface model of silicon and silicon nitride (Si/Si_3N_4), which was used to model high dielectric constant materials for electronics.[122] Also for this system, which has 2376 atoms and 10 368 electrons, we compute the band offsets between the two materials using G_0W_0. The structure in Figure 9c is a neutral hh divacancy in a $10 \times 10 \times 2$ supercell of 4H silicon carbide (VV_SiC).[49] This is a representative system for solid-state quantum information technologies where G_0W_0 is used to identify the energy of deep defect states. The system has 1598 atoms and 6392 electrons and, unlike the previous two systems, requires an explicit treatment of spin polarization. The details of the structures in Figure 9 are summarized in Table 7.

For all considered systems we computed the local density of states (LDOS), defined as

$$\mathrm{LDOS}(z, E) = \sum_{i\sigma} \int \frac{\mathrm{d}x}{L_x} \int \frac{\mathrm{d}y}{L_y} |\psi_{i\sigma}(x, y, z)|^2 \delta(E - \epsilon_{i\sigma}) \tag{20}$$

where $\psi_{i\sigma}$ and $\epsilon_{i\sigma}$ are the wave functions and their G_0W_0 or PBE energies, L_x and L_y are the lengths of the x and y axes of the simulation box, respectively, whereas z corresponds to the z axis of the simulation box, and δ is the Dirac delta function (modeled by a Gaussian function with a width of 0.05 eV). The middle panel of Figure 9 reports the LDOS computed using PBE wave functions and energy levels. The bottom panel of Figure 9 reports the LDOS computed using PBE wave functions and full-frequency G_0W_0 energy levels. To compute the LDOS, the QP energies of 480, 2000, and 1200 single-particle states were computed for the CdS/PbS, Si/Si_3N_4, and VV_SiC structures, respectively. As expected, the LDOS at the G_0W_0@PBE level exhibits a larger energy gap compared to the PBE result for all structures. For the Janus-like nanoparticle and the Si/Si_3N_4 interface, the LDOS allows us to track the density of states as a function of the coordinate z that is perpendicular to the interface. For the system in Figure 9c, the energy gap of 4H-SiC obtained at the G_0W_0@PBE level, 3.17 eV, is in close agreement with the experimental value of 3.2 eV.[123] At the G_0W_0 (PBE) level of theory, the energy difference between the e and the a_1 defect states in the

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707

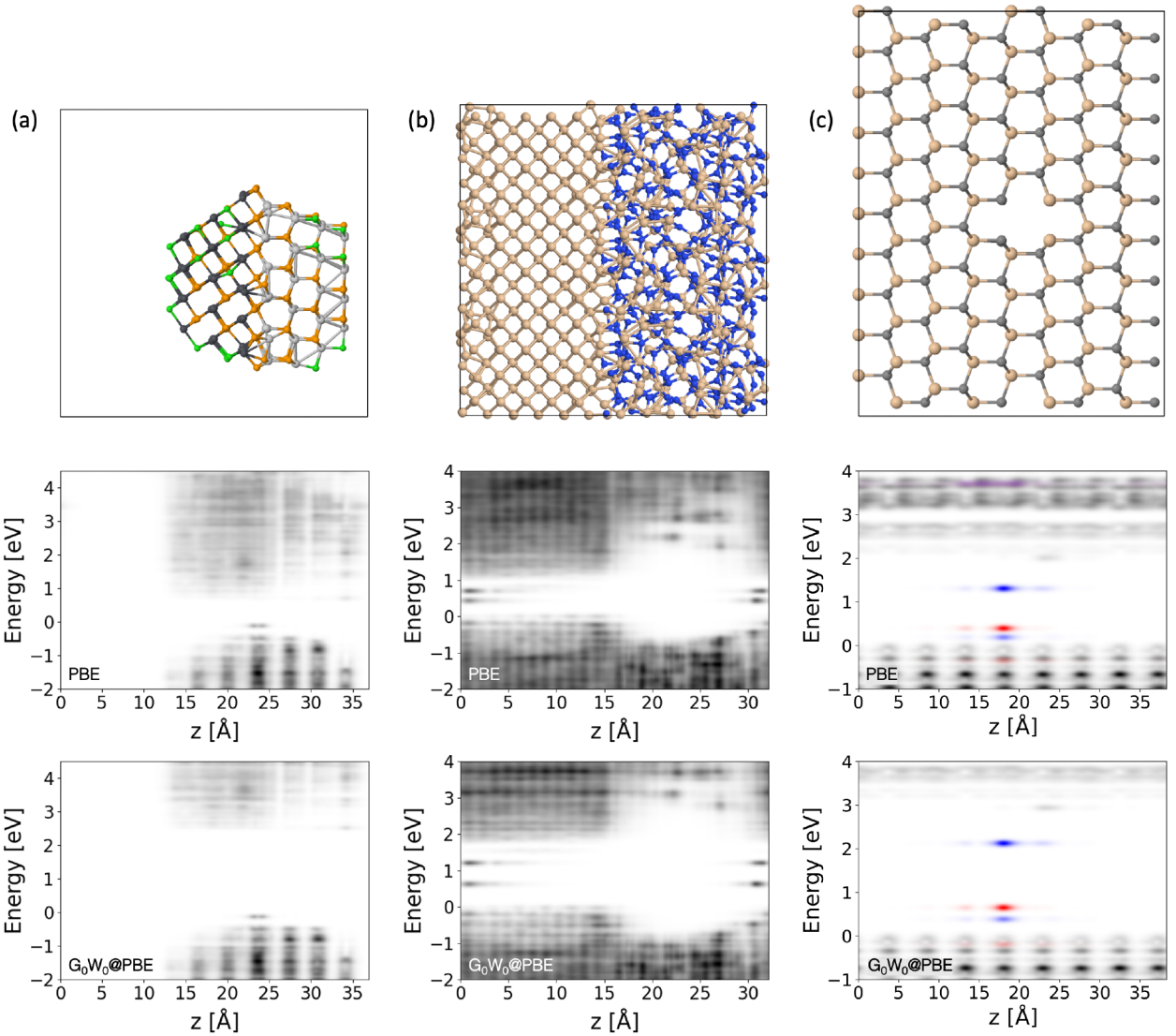


Figure 9. Large-scale full-frequency G_0W_0 calculations considered in this work: (a) Janus-like heterostructure formed by a chlorine-terminated nanoparticle made of cadmium sulfide and lead sulfide (CdS/PbS), (b) interface of silicon and silicon nitride (Si/Si_3N_4), and (c) neutral hh divacancy in 4H silicon carbide (VV_SiC). Top panels report a side view of the simulation cells. For the ball-and-stick atomic structures, the Cl, Cd, S, Pb, Si, N, and C atoms are colored in green, black, orange, light gray, beige, blue, and dark gray, respectively. Bottom and middle panels report the local density of states (LDOS, see text) obtained using G_0W_0@PBE or KS-DFT energies in eq 20, respectively. LDOS is plotted using a color scale ranging from white to black; white areas indicate energy gaps. For VV_SiC, the defect states in the up (down) spin channel are shown in red (blue). Details of the systems are reported in Table 7.

Table 7. Simulation Parameters for the Systems Depicted in Figure 9[a]

system	N_{atom}	$N_{electron}$	N_{spin}	E_{cut} [Ry]	N_{PW}
CdS/PbS	301	2816	1	30	948 557
Si/Si_3N_4	2376	10 368	1	30	638 633
VV_SiC	1598	6392	2	30	314 653

[a] N_{atom}, $N_{electron}$, N_{spin}, and N_{PW} denote the numbers of atoms, electrons, spin channels, and plane-waves, respectively. E_{cut} denotes the cutoff energy used in the calculations.

minority spin channel is $e - a_1 = 1.73(1.12)$ eV. We obtain an exciton binding energy and an ionic relaxation energy of 0.45 and 0.10 eV, respectively, from a DFT calculation[124] of the hh divacancy in an $8 \times 8 \times 2$ supercell of 4H-SiC using the dielectric dependent hybrid functional (DDH).[125] Subtracting these energies from $e - a_1$ computed at the G_0W_0 level of theory, we obtain 1.18 eV, which is close to the measured zero-phonon line (ZPL) of 1.095 eV.[126]

The calculations reported in this section were carried out on the Summit supercomputer using ~10 000 NVIDIA V100 GPUs (6 GPUs per node). The measured number of FLOPs, time to solution, and performance in terms of FLOP/s are shown in Table 8. The `wstat` (`wfreq`) code achieves up to 35.8% (23.2%) of the theoretical peak performance. Due to the size of the memory available in V100 GPUs, we had to distribute FFT operations within each band group on 12 GPUs. This configuration does not yield optimal performance for FFTs (one GPU per band group), as discussed in section 4.1. We anticipate seeing improved performance on GPUs that have more device memory than the V100 GPUs.

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707

Table 8. Performance (PFLOP/s) of WEST-GPU on Summit for the Systems Described in Table 7.[a]

code	system	N_{FLOP} [PFLOPs]	N_{node}	time [s]	perf. [PFLOP/s]	% peak
wstat	CdS/PbS	1.43×10^4	1408	731.9	19.54	31.9
	Si/Si_3N_4	4.04×10^5	1728	18 356.9	21.95	29.2
	VV_SiC	8.81×10^4	1600	3536.4	24.91	35.8
wfreq	CdS/PbS	7.99×10^3	1408	562.0	14.22	23.2
	Si/Si_3N_4	2.54×10^5	1728	16 309.3	15.58	20.7
	VV_SiC	6.39×10^4	1600	3333.0	19.17	20.9

[a]The wstat and wfreq codes are described in section 3. Timing results correspond to the total wall clock time, including the time spent on I/O operations and CPU−GPU communications. The performance (perf.) is measured as the ratio of the total number of FLOPs (N_{FLOP}) to the total time of the simulation. N_{node} denotes the number of Summit nodes used in the calculations (each node has six GPUs, see Table 2). % peak denotes the ratio of the measured performance to the theoretical peak performance; the latter is calculated as $N_{node} \times 43.5$ TFLOP/s

7. CONCLUSIONS

We reported the use of GPUs to carry out large-scale full-frequency G_0W_0 calculations using WEST, a code for many-body perturbation theory calculations based on the plane-wave and pseudopotential method. Compared to other conventional implementations of G_0W_0, the algorithms implemented in WEST do not require the calculation of many empty states nor the definition of a stringent energy cutoff for response functions. In this work, we introduced a multilevel parallelization strategy in WEST that is devised to distribute the computational workload and reduce the overhead cost associated with MPI communications. We discussed a number of optimizations that improve the performance of the code on GPU-equipped supercomputers, including the use of mature high-performance libraries, and strategies to minimize data transfer operations between CPUs and GPUs. In addition, the utilization of mixed precision led to a considerable performance improvement over the solely double-precision version without sacrificing accuracy.

The GPU-accelerated version of WEST realizes substantial speedup over its CPU-only counterpart and displays excellent strong and weak scaling, as benchmarked on the Summit supercomputer using up to 25 920 NVIDIA V100 GPUs. The code reaches a mixed-precision (FP32/FP64) performance of 58.8 PFLOP/s. The same code runs seamlessly on the Perlmutter supercomputer equipped with NVIDIA A100 GPUs, which are one generation newer than the V100 GPUs on Summit. The newly developed GPU code has the capability of advancing the simulation of electronic excitations in large heterogeneous materials, as demonstrated by carrying out full-frequency G_0W_0 calculations of a nanostructure, an interface, and a defect in a wide band gap material using supercells with up to 10 368 electrons. These calculations are carried out overcoming commonly adopted approximations, e.g., the use of generalized plasmon-pole models, and are, to the best of our knowledge, the largest deterministic full-frequency G_0W_0 calculations performed to date.

We are exploring the possibility to extend mixed-precision regions to other memory-intensive or compute-intensive portions of the code, e.g., the storage of the nonlocal part of the pseudopotential and the evaluation of the exact exchange needed in hybrid density functionals.[127] Work is in progress to optimize the performance on GPUs of the electron−phonon,[50,51] the Bethe−Salpeter equation (BSE) in finite field,[52,53] and the quantum embedding[54−56] kernels.

Data related to this publication are organized using the Qresp software[128] and are available online at https://paperstack.uchicago.edu.

AUTHOR INFORMATION

Corresponding Author

Marco Govoni − *Materials Science Division, Argonne National Laboratory, Lemont, Illinois 60439, United States; Pritzker School of Molecular Engineering, The University of Chicago, Chicago, Illinois 60637, United States;* orcid.org/0000-0001-6303-2403; Email: mgovoni@anl.gov

Author

Victor Wen-zhe Yu − *Materials Science Division, Argonne National Laboratory, Lemont, Illinois 60439, United States;* orcid.org/0000-0002-6568-1244

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jctc.2c00241

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

The authors acknowledge support provided by the Midwest Integrated Center for Computational Materials (MICCoM), as part of the Computational Materials Sciences Program funded by the U.S. Department of Energy, Office of Science, Basic Energy Sciences, Materials Sciences, and Engineering Division through Argonne National Laboratory (ANL). This research used resources of the Oak Ridge Leadership Computing Facility (OLCF) at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725. This research used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility operated under Contract No. DE-AC02-05CH11231. This work was additionally supported by participation in the NERSC Exascale Application Readiness Program. This research used resources of the Argonne Leadership Computing Facility (ALCF) at ANL, a U.S. Department of Energy Office of Science User Facility operated under Contract No. DE-AC02-06CH11357. We thank Dr. Brandon Cook (NERSC), Dr. Soham Ghosh (NERSC), Prof. Giulia Galli (University of Chicago and ANL), Prof. François Gygi (University of California, Davis), Dr. Christopher Knight (ALCF), Ryan Prout (OLCF), Dr. Marcello Puligheddu (ANL), Dr. Jonathan Skone (NERSC), and Dr. John Vinson (National Institute of Standards and Technology) for fruitful discussions. We gratefully acknowledge the organizers of a GPU hackathon event co-organized by ALCF and NVIDIA Corp. and Dr. William Huhn (ALCF) and

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707

Dr. Kristopher Keipert (NVIDIA) for their advice during this event.

REFERENCES

(1) Hohenberg, P.; Kohn, W. Inhomogeneous electron gas. *Phys. Rev.* **1964**, *136*, B864−B871.
(2) Kohn, W.; Sham, L. J. Self-consistent equations including exchange and correlation effects. *Phys. Rev.* **1965**, *140*, A1133−A1138.
(3) Strinati, G. Application of the Green's functions method to the study of the optical properties of semiconductors. *Riv. Nuovo Cim.* **1988**, *11*, 1−86.
(4) Golze, D.; Dvorak, M.; Rinke, P. The GW compendium: A practical guide to theoretical photoemission spectroscopy. *Front. Chem.* **2019**, *7*, 377.
(5) Hedin, L. New method for calculating the one-particle Green's function with application to the electron-gas problem. *Phys. Rev.* **1965**, *139*, A796−A823.
(6) Strinati, G.; Mattausch, H. J.; Hanke, W. Dynamical correlation effects on the quasiparticle Bloch states of a covalent crystal. *Phys. Rev. Lett.* **1980**, *45*, 290−294.
(7) Strinati, G.; Mattausch, H. J.; Hanke, W. Dynamical aspects of correlation corrections in a covalent crystal. *Phys. Rev. B* **1982**, *25*, 2867−2888.
(8) Hybertsen, M. S.; Louie, S. G. First-principles theory of quasiparticles: Calculation of band gaps in semiconductors and insulators. *Phys. Rev. Lett.* **1985**, *55*, 1418.
(9) Hybertsen, M. S.; Louie, S. G. Electron correlation in semiconductors and insulators: Band gaps and quasiparticle energies. *Phys. Rev. B* **1986**, *34*, 5390.
(10) Aryasetiawan, F.; Gunnarsson, O. The GW method. *Rep. Prog. Phys.* **1998**, *61*, 237.
(11) Foerster, D.; Koval, P.; Sánchez-Portal, D. An $O(N^3)$ implementation of Hedin's GW approximation for molecules. *J. Chem. Phys.* **2011**, *135*, 074105.
(12) Liu, P.; Kaltak, M.; Klimeš, J.; Kresse, G. Cubic scaling GW: Towards fast quasiparticle calculations. *Phys. Rev. B* **2016**, *94*, 165109.
(13) Wilhelm, J.; Golze, D.; Talirz, L.; Hutter, J.; Pignedoli, C. A. Toward GW calculations on thousands of atoms. *J. Phys. Chem. Lett.* **2018**, *9*, 306−312.
(14) Förster, A.; Visscher, L. Low-order scaling G_0W_0 by pair atomic density fitting. *J. Chem. Theory Comput.* **2020**, *16*, 7381−7399.
(15) Wilhelm, J.; Seewald, P.; Golze, D. Low-scaling GW with benchmark accuracy and application to phosphorene nanosheets. *J. Chem. Theory Comput.* **2021**, *17*, 1662−1677.
(16) Duchemin, I.; Blase, X. Cubic-scaling all-electron GW calculations with a separable density-fitting space-time approach. *J. Chem. Theory Comput.* **2021**, *17*, 2383−2393.
(17) Förster, A.; Visscher, L. Low-order scaling quasiparticle self-consistent GW for molecules. *Front. Chem.* **2021**, *9*, 736591.
(18) 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. *Phys. Rev. Lett.* **2014**, *113*, 076402.
(19) Vlček, V.; Rabani, E.; Neuhauser, D.; Baer, R. Stochastic GW calculations for molecules. *J. Chem. Theory Comput.* **2017**, *13*, 4997−5003.
(20) Vlček, V.; Li, W.; Baer, R.; Rabani, E.; Neuhauser, D. Swift GW beyond 10,000 electrons using sparse stochastic compression. *Phys. Rev. B* **2018**, *98*, 075107.
(21) Brooks, J.; Weng, G.; Taylor, S.; Vlček, V. Stochastic many-body perturbation theory for moiré states in twisted bilayer phosphorene. *J. Phys.: Condens. Matter* **2020**, *32*, 234001.
(22) Romanova, M.; Vlček, V. Stochastic many-body calculations of moiré states in twisted bilayer graphene at high pressures. *npj Comput. Mater.* **2022**, *8*, 11.
(23) *TOP500*; https://top500.org (accessed May 23, 2022).
(24) Gonze, X.; Jollet, F.; Abreu Araujo, F.; Adams, D.; Amadon, B.; Applencourt, T.; Audouze, C.; Beuken, J.-M.; Bieder, J.; Bokhanchuk, A.; Bousquet, E.; Bruneval, F.; Caliste, D.; Côté, M.; Dahm, F.; Da Pieve, F.; Delaveau, M.; Di Gennaro, M.; Dorado, B.; Espejo, C.; Geneste, G.; Genovese, L.; Gerossier, A.; Giantomassi, M.; Gillet, Y.; Hamann, D. R.; He, L.; Jomard, G.; Laflamme Janssen, J.; Le Roux, S.; Levitt, A.; Lherbier, A.; Liu, F.; Lukačević, I.; Martin, A.; Martins, C.; Oliveira, M. J. T.; Poncé, S.; Pouillon, Y.; Rangel, T.; Rignanese, G.-M.; Romero, A. H.; Rousseau, B.; Rubel, O.; Shukri, A. A.; Stankovski, M.; Torrent, M.; Van Setten, M. J.; Van Troeye, B.; Verstraete, M. J.; Waroquiers, D.; Wiktor, J.; Xu, B.; Zhou, A.; Zwanziger, J. W. Recent developments in the ABINIT software package. *Comput. Phys. Commun.* **2016**, *205*, 106−131.
(25) Ratcliff, L. E.; Degomme, A.; Flores-Livas, J. A.; Goedecker, S.; Genovese, L. Affordable and accurate large-scale hybrid functional calculations on GPU-accelerated supercomputers. *J. Phys.: Condens. Matter* **2018**, *30*, 095901.
(26) Kühne, T. D.; Iannuzzi, M.; Del Ben, M.; Rybkin, V. V.; Seewald, P.; Stein, F.; Laino, T.; Khaliullin, R. Z.; Schütt, O.; Schiffmann, F.; Golze, D.; Wilhelm, J.; Chulkov, S.; Bani-Hashemian, M. H.; Weber, V.; Borštnik, U.; Taillefumier, M.; Jakobovits, A. S.; Lazzaro, A.; Pabst, H.; Müller, T.; Schade, R.; Guidon, M.; Andermatt, S.; Holmberg, N.; Schenter, G. K.; Hehn, A.; Bussy, A.; Belleflamme, F.; Tabacchi, G.; Glöß, A.; Lass, M.; Bethune, I.; Mundy, C. J.; Plessl, C.; Watkins, M.; VandeVondele, J.; Krack, M.; Hutter, J. CP2K: An electronic structure and molecular dynamics software package − Quickstep: Efficient and accurate electronic structure calculations. *J. Chem. Phys.* **2020**, *152*, 194103.
(27) Huhn, W. P.; Lange, B.; Yu, V. W.-z.; Yoon, M.; Blum, V. GPU acceleration of all-electron electronic structure theory using localized numeric atom-centered basis functions. *Comput. Phys. Commun.* **2020**, *254*, 107314.
(28) Aprà, E.; Bylaska, E. J.; de Jong, W. A.; Govind, N.; Kowalski, K.; Straatsma, T. P.; Valiev, M.; van Dam, H. J. J.; Alexeev, Y.; Anchell, J.; Anisimov, V.; Aquino, F. W.; Atta-Fynn, R.; Autschbach, J.; Bauman, N. P.; Becca, J. C.; Bernholdt, D. E.; Bhaskaran-Nair, K.; Bogatko, S.; Borowski, P.; Boschen, J.; Brabec, J.; Bruner, A.; Cauët, E.; Chen, Y.; Chuev, G. N.; Cramer, C. J.; Daily, J.; Deegan, M. J. O.; Dunning, T. H.; Dupuis, M.; Dyall, K. G.; Fann, G. I.; Fischer, S. A.; Fonari, A.; Früchtl, H.; Gagliardi, L.; Garza, J.; Gawande, N.; Ghosh, S.; Glaesemann, K.; Götz, A. W.; Hammond, J.; Helms, V.; Hermes, E. D.; Hirao, K.; Hirata, S.; Jacquelin, M.; Jensen, L.; Johnson, B. G.; J'onsson, H.; Kendall, R. A.; Klemm, M.; Kobayashi, R.; Konkov, V.; Krishnamoorthy, S.; Krishnan, M.; Lin, Z.; Lins, R. D.; Littlefield, R. J.; Logsdail, A. J.; Lopata, K.; Ma, W.; Marenich, A. V.; Martin del Campo, J.; Mejia-Rodriguez, D.; Moore, J. E.; Mullin, J. M.; Nakajima, T.; Nascimento, D. R.; Nichols, J. A.; Nichols, P. J.; Nieplocha, J.; Otero-de-la-Roza, A.; Palmer, B.; Panyala, A.; Pirojsirikul, T.; Peng, B.; Peverati, R.; Pittner, J.; Pollack, L.; Richard, R. M.; Sadayappan, P.; Schatz, G. C.; Shelton, W. A.; Silverstein, D. W.; Smith, D. M. A.; Soares, T. A.; Song, D.; Swart, M.; Taylor, H. L.; Thomas, G. S.; Tipparaju, V.; Truhlar, D. G.; Tsemekhman, K.; Van Voorhis, T.; Vázquez-Mayagoitia, A.; Verma, P.; Villa, O.; Vishnu, A.; Vogiatzis, K. D.; Wang, D.; Weare, J. H.; Williamson, M. J.; Windus, T. L.; Woliński, K.; Wong, A. T.; Wu, Q.; Yang, C.; Yu, Q.; Zacharias, M.; Zhang, Z.; Zhao, Y.; Harrison, R. J. NWChem: Past, present, and future. *J. Chem. Phys.* **2020**, *152*, 184102.
(29) Tancogne-Dejean, N.; Oliveira, M. J. T.; Andrade, X.; Appel, H.; Borca, C. H.; Le Breton, G.; Buchholz, F.; Castro, A.; Corni, S.; Correa, A. A.; De Giovannini, U.; Delgado, A.; Eich, F. G.; Flick, J.; Gil, G.; Gomez, A.; Helbig, N.; Hübener, H.; Jestädt, R.; Jornet-Somoza, J.; Larsen, A. H.; Lebedeva, I. V.; Lüders, M.; Marques, M. A. L.; Ohlmann, S. T.; Pipolo, S.; Rampp, M.; Rozzi, C. A.; Strubbe, D. A.; Sato, S. A.; Schäfer, C.; Theophilou, I.; Welden, A.; Rubio, A. Octopus, a computational framework for exploring light-driven phenomena and quantum dynamics in extended and finite systems. *J. Chem. Phys.* **2020**, *152*, 124119.
(30) Giannozzi, P.; Baseggio, O.; Bonfà, 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. *J. Chem. Phys.* **2020**, *152*, 154105.

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707

(31) Seritan, S.; Bannwarth, C.; Fales, B. S.; Hohenstein, E. G.; Kokkila-Schumacher, S. I.; Luehr, N.; Snyder, J. W., Jr; Song, C.; Titov, A. V.; Ufimtsev, I. S.; Martínez, T. J. TeraChem: Accelerating electronic structure and ab initio molecular dynamics with graphical processing units. *J. Chem. Phys.* **2020**, *152*, 224110.
(32) Tirimbó, G.; Sundaram, V.; Çaylak, O.; Scharpach, W.; Sijen, J.; Junghans, C.; Brown, J.; Ruiz, F. Z.; Renaud, N.; Wehner, J.; Baumeier, B. Excited-state electronic structure of molecules using many-body Green's functions: Quasiparticles and electron−hole excitations with VOTCA-XTP. *J. Chem. Phys.* **2020**, *152*, 114103.
(33) Sangalli, D.; Ferretti, A.; Miranda, H.; Attaccalite, C.; Marri, I.; Cannuccia, E.; Melo, P.; Marsili, M.; Paleari, F.; Marrazzo, A.; Prandini, G.; Bonfà, P.; Atambo, M. O.; Affinito, F.; Palummo, M.; Molina-S'anchez, A.; Hogan, C.; Grüning, M.; Varsano, D.; Marini, A. Many-body perturbation theory calculations using the yambo code. *J. Phys.: Condens. Matter* **2019**, *31*, 325902.
(34) Del Ben, M.; Yang, C.; Li, Z.; da Jornada, F. H.; 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*, 9−19 Nov 2020, Atlanta, GA; IEEE, 2020; pp 1−11; DOI: 10.1109/SC41405.2020.00008.
(35) Govoni, M.; Galli, G. Large scale GW calculations. *J. Chem. Theory Comput.* **2015**, *11*, 2680−2696.
(36) *WEST*; http://www.west-code.org (accessed May 23, 2022).
(37) Govoni, M.; Galli, G. GW100: Comparison of methods and accuracy of results obtained with the WEST code. *J. Chem. Theory Comput.* **2018**, *14*, 1895−1909.
(38) Seo, H.; Govoni, M.; Galli, G. Design of defect spins in piezoelectric aluminum nitride for solid-state hybrid quantum technologies. *Sci. Rep.* **2016**, *6*, 20803.
(39) Gaiduk, A. P.; Govoni, M.; Seidel, R.; Skone, J. H.; Winter, B.; Galli, G. Photoelectron spectra of aqueous solutions from first principles. *J. Am. Chem. Soc.* **2016**, *138*, 6912−6915.
(40) Scherpelz, P.; Govoni, M.; Hamada, I.; Galli, G. Implementation and validation of fully relativistic GW calculations: Spin-orbit coupling in molecules, nanocrystals, and solids. *J. Chem. Theory Comput.* **2016**, *12*, 3523−3544.
(41) Pham, T. A.; Govoni, M.; Seidel, R.; Bradforth, S. E.; Schwegler, E.; Galli, G. Electronic structure of aqueous solutions: Bridging the gap between theory and experiments. *Sci. Adv.* **2017**, *3*, No. e1603210.
(42) Seo, H.; Ma, H.; Govoni, M.; Galli, G. Designing defect-based qubit candidates in wide-gap binary semiconductors for solid-state quantum technologies. *Phys. Rev. Mater.* **2017**, *1*, 075002.
(43) Gaiduk, A. P.; Pham, T. A.; Govoni, M.; Paesani, F.; Galli, G. Electron affinity of liquid water. *Nat. Commun.* **2018**, *9*, 247.
(44) Smart, T. J.; Wu, F.; Govoni, M.; Ping, Y. Fundamental principles for calculating charged defect ionization energies in ultrathin two-dimensional materials. *Phys. Rev. Mater.* **2018**, *2*, 124002.
(45) Gerosa, M.; Gygi, F.; Govoni, M.; Galli, G. The role of defects and excess surface charges at finite temperature for optimizing oxide photoabsorbers. *Nat. Mater.* **2018**, *17*, 1122−1127.
(46) Zheng, H.; Govoni, M.; Galli, G. Dielectric-dependent hybrid functionals for heterogeneous materials. *Phys. Rev. Mater.* **2019**, *3*, 073803.
(47) Yang, H.; Govoni, M.; Galli, G. Improving the efficiency of G_0W_0 calculations with approximate spectral decompositions of dielectric matrices. *J. Chem. Phys.* **2019**, *151*, 224102.
(48) Kundu, A.; Govoni, M.; Yang, H.; Ceriotti, M.; Gygi, F.; Galli, G. Quantum vibronic effects on the electronic properties of solid and molecular carbon. *Phys. Rev. Mater.* **2021**, *5*, L070801.
(49) Jin, Y.; Govoni, M.; Wolfowicz, G.; Sullivan, S. E.; Heremans, F. J.; Awschalom, D. D.; Galli, G. Photoluminescence spectra of point defects in semiconductors: Validation of first-principles calculations. *Phys. Rev. Mater.* **2021**, *5*, 084603.
(50) McAvoy, R. L.; Govoni, M.; Galli, G. Coupling first-principles calculations of electron−electron and electron−phonon scattering, and applications to carbon-based nanostructures. *J. Chem. Theory Comput.* **2018**, *14*, 6269−6275.
(51) Yang, H.; Govoni, M.; Kundu, A.; Galli, G. Combined first-principles calculations of electron−electron and electron−phonon self-energies in condensed systems. *J. Chem. Theory Comput.* **2021**, *17*, 7468−7476.
(52) Nguyen, N. L.; Ma, H.; Govoni, M.; Gygi, F.; Galli, G. Finite-field approach to solving the Bethe-Salpeter equation. *Phys. Rev. Lett.* **2019**, *122*, 237402.
(53) Dong, S. S.; Govoni, M.; Galli, G. Machine learning dielectric screening for the simulation of excited state properties of molecules and materials. *Chem. Sci.* **2021**, *12*, 4970−4980.
(54) Ma, H.; Govoni, M.; Galli, G. Quantum simulations of materials on near-term quantum computers. *npj Comput. Mater.* **2020**, *6*, 85.
(55) Ma, H.; Sheng, N.; Govoni, M.; Galli, G. First-principles studies of strongly correlated states in defect spin qubits in diamond. *Phys. Chem. Chem. Phys.* **2020**, *22*, 25522−25527.
(56) Ma, H.; Sheng, N.; Govoni, M.; Galli, G. Quantum embedding theory for strongly correlated states in materials. *J. Chem. Theory Comput.* **2021**, *17*, 2116−2125.
(57) Godby, R. W.; Schlüter, M.; Sham, L. J. Self-energy operators and exchange-correlation potentials in semiconductors. *Phys. Rev. B* **1988**, *37*, 10159−10175.
(58) Lebègue, S.; Arnaud, B.; Alouani, M.; Bloechl, P. E. Implementation of an all-electron GW approximation based on the projector augmented wave method without plasmon pole approximation: Application to Si, SiC, AlAs, InAs, NaH, and KH. *Phys. Rev. B* **2003**, *67*, 155208.
(59) Gonze, X.; Amadon, B.; Anglade, P.-M.; Beuken, J.-M.; Bottin, F.; Boulanger, P.; Bruneval, F.; Caliste, D.; Caracas, R.; Côté, M.; Deutsch, T.; Genovese, L.; Ghosez, P.; Giantomassi, M.; Goedecker, S.; Hamann, D. R.; Hermet, P.; Jollet, F.; Jomard, G.; Leroux, S.; Mancini, M.; Mazevet, S.; Oliveira, M. J. T.; Onida, G.; Pouillon, Y.; Rangel, T.; Rignanese, G.-M.; Sangalli, D.; Shaltaf, R.; Torrent, M.; Verstraete, M. J.; Zerah, G.; Zwanziger, J. W. ABINIT: First-principles approach to material and nanosystem properties. *Comput. Phys. Commun.* **2009**, *180*, 2582−2615.
(60) Onida, G.; Reining, L.; Rubio, A. Electronic excitations: Density-functional versus many-body Green's-function approaches. *Rev. Mod. Phys.* **2002**, *74*, 601−659.
(61) Ping, Y.; Rocca, D.; Galli, G. Electronic excitations in light absorbers for photoelectrochemical energy conversion: First principles calculations based on many body perturbation theory. *Chem. Soc. Rev.* **2013**, *42*, 2437.
(62) Adler, S. L. Quantum theory of the dielectric constant in real solids. *Phys. Rev.* **1962**, *126*, 413−420.
(63) Wiser, N. Dielectric constant with local field effects included. *Phys. Rev.* **1963**, *129*, 62−69.
(64) von der Linden, W.; Horsch, P. Precise quasiparticle energies and Hartree-Fock bands of semiconductors and insulators. *Phys. Rev. B* **1988**, *37*, 8351−8362.
(65) Engel, G. E.; Farid, B. Generalized plasmon-pole model and plasmon band structures of crystals. *Phys. Rev. B* **1993**, *47*, 15931−15934.
(66) Shaltaf, R.; Rignanese, G.-M.; Gonze, X.; Giustino, F.; Pasquarello, A. Band offsets at the Si/SiO_2 interface from many-body perturbation theory. *Phys. Rev. Lett.* **2008**, *100*, 186401.
(67) Stankovski, M.; Antonius, G.; Waroquiers, D.; Miglio, A.; Dixit, H.; Sankaran, K.; Giantomassi, M.; Gonze, X.; Côté, M.; Rignanese, G.-M. G_0W_0 band gap of ZnO: Effects of plasmon-pole models. *Phys. Rev. B* **2011**, *84*, 241201.
(68) Larson, P.; Dvorak, M.; Wu, Z. Role of the plasmon-pole model in the GW approximation. *Phys. Rev. B* **2013**, *88*, 125205.
(69) van Setten, M. J.; Caruso, F.; Sharifzadeh, S.; Ren, X.; Scheffler, M.; Liu, F.; Lischner, J.; Lin, L.; Deslippe, J.; Louie, S. G.; Yang, C.; Weigend, F.; Neaton, J. B.; Evers, F.; Rinke, P. GW100: Benchmarking G_0W_0 for molecular systems. *J. Chem. Theory Comput.* **2015**, *11*, 5665−5687.

(70) Maggio, E.; Liu, P.; van Setten, M. J.; Kresse, G. GW100: A plane wave perspective for small molecules. *J. Chem. Theory Comput.* **2017**, *13*, 635−648.
(71) Rangel, T.; Del Ben, M.; Varsano, D.; Antonius, G.; Bruneval, F.; da Jornada, F. H.; van Setten, M. J.; Orhan, O. K.; O'Regan, D. D.; Canning, A.; Ferretti, A.; Marini, A.; Rignanese, G.-M.; Deslippe, J.; Louie, S. G.; Neaton, J. B. Reproducibility in G_0W_0 calculations for solids. *Comput. Phys. Commun.* **2020**, *255*, 107242.
(72) Bruneval, F.; Gonze, X. Accurate GW self-energies in a plane-wave basis using only a few empty states: Towards large systems. *Phys. Rev. B* **2008**, *78*, 085125.
(73) Bruneval, F. Optimized virtual orbital subspace for faster GW calculations in localized basis. *J. Chem. Phys.* **2016**, *145*, 234110.
(74) Kang, W.; Hybertsen, M. S. Enhanced static approximation to the electron self-energy operator for efficient calculation of quasiparticle energies. *Phys. Rev. B* **2010**, *82*, 195108.
(75) Berger, J. A.; Reining, L.; Sottile, F. Ab initio calculations of electronic excitations: Collapsing spectral sums. *Phys. Rev. B* **2010**, *82*, 041103.
(76) Berger, J. A.; Reining, L.; Sottile, F. Efficient GW calculations for SnO_2, ZnO, and rubrene: The effective-energy technique. *Phys. Rev. B* **2012**, *85*, 085126.
(77) Leon, D. A.; Cardoso, C.; Chiarotti, T.; Varsano, D.; Molinari, E.; Ferretti, A. Frequency dependence in GW made simple using a multipole approximation. *Phys. Rev. B* **2021**, *104*, 115157.
(78) Reining, L.; Onida, G.; Godby, R. W. Elimination of unoccupied-state summations in ab initio self-energy calculations for large supercells. *Phys. Rev. B* **1997**, *56*, R4301−R4304.
(79) Wilson, H. F.; Gygi, F.; Galli, G. Efficient iterative method for calculations of dielectric matrices. *Phys. Rev. B* **2008**, *78*, 113303.
(80) Wilson, H. F.; Lu, D.; Gygi, F.; Galli, G. Iterative calculations of dielectric eigenvalue spectra. *Phys. Rev. B* **2009**, *79*, 245106.
(81) Umari, P.; Stenuit, G.; Baroni, S. Optimal representation of the polarization propagator for large-scale GW calculations. *Phys. Rev. B* **2009**, *79*, 201104.
(82) Umari, P.; Stenuit, G.; Baroni, S. GW quasiparticle spectra from occupied states only. *Phys. Rev. B* **2010**, *81*, 115104.
(83) Giustino, F.; Cohen, M. L.; Louie, S. G. GW method with the self-consistent Sternheimer equation. *Phys. Rev. B* **2010**, *81*, 115105.
(84) Lambert, H.; Giustino, F. Ab initio Sternheimer-GW method for quasiparticle calculations using plane waves. *Phys. Rev. B* **2013**, *88*, 075117.
(85) Nguyen, H.-V.; Pham, T. A.; Rocca, D.; Galli, G. Improving accuracy and efficiency of calculations of photoemission spectra within the many-body perturbation theory. *Phys. Rev. B* **2012**, *85*, 081101.
(86) Pham, T. A.; Nguyen, H.-V.; Rocca, D.; Galli, G. GW calculations using the spectral decomposition of the dielectric matrix: Verification, validation, and comparison of methods. *Phys. Rev. B* **2013**, *87*, 155148.
(87) Laflamme Janssen, J.; Rousseau, B.; Côté, M. Efficient dielectric matrix calculations using the Lanczos algorithm for fast many-body G_0W_0 implementations. *Phys. Rev. B* **2015**, *91*, 125120.
(88) Baroni, S.; Giannozzi, P.; Testa, A. Green's-function approach to linear response in solids. *Phys. Rev. Lett.* **1987**, *58*, 1861−1864.
(89) Baroni, S.; de Gironcoli, S.; Dal Corso, A.; Giannozzi, P. Phonons and related crystal properties from density-functional perturbation theory. *Rev. Mod. Phys.* **2001**, *73*, 515−562.
(90) 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. *Comput. Phys. Commun.* **2012**, *183*, 1269−1289.
(91) Hüser, F.; Olsen, T.; Thygesen, K. S. Quasiparticle GW calculations for solids, molecules, and two-dimensional materials. *Phys. Rev. B* **2013**, *87*, 235132.
(92) Kim, M.; Mandal, S.; Mikida, E.; Chandrasekar, K.; Bohm, E.; Jain, N.; Li, Q.; Kanakagiri, R.; Martyna, G. J.; Kale, L.; Ismail-Beigi, S. Scalable GW software for quasiparticle properties using OpenAtom. *Comput. Phys. Commun.* **2019**, *244*, 427−441.
(93) Schlipf, M.; Lambert, H.; Zibouche, N.; Giustino, F. SternheimerGW: A program for calculating GW quasiparticle band structures and spectral functions without unoccupied states. *Comput. Phys. Commun.* **2020**, *247*, 106856.
(94) Kresse, G.; Furthmüller, J. Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set. *Phys. Rev. B* **1996**, *54*, 11169−11186.
(95) Blase, X.; Attaccalite, C.; Olevano, V. First-principles GW calculations for fullerenes, porphyrins, phtalocyanine, and other molecules of interest for organic photovoltaic applications. *Phys. Rev. B* **2011**, *83*, 115103.
(96) 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. *Comput. Phys. Commun.* **2016**, *208*, 149−161.
(97) Balasubramani, S. G.; Chen, G. P.; Coriani, S.; Diedenhofen, M.; Frank, M. S.; Franzke, Y. J.; Furche, F.; Grotjahn, R.; Harding, M. E.; Hättig, C.; Hellweg, A.; Helmich-Paris, B.; Holzer, C.; Huniar, U.; Kaupp, M.; Marefat Khah, A.; Karbalaei Khani, S.; Müller, T.; Mack, F.; Nguyen, B. D.; Parker, S. M.; Perlt, E.; Rappoport, D.; Reiter, K.; Roy, S.; Rückert, M.; Schmitz, G.; Sierka, M.; Tapavicza, E.; Tew, D. P.; van Wüllen, C.; Voora, V. K.; Weigend, F.; Wodyński, A.; Yu, J. M. TURBOMOLE: Modular program suite for ab initio quantum-chemical and condensed-matter simulations. *J. Chem. Phys.* **2020**, *152*, 184107.
(98) Ren, X.; Merz, F.; Jiang, H.; Yao, Y.; Rampp, M.; Lederer, H.; Blum, V.; Scheffler, M. All-electron periodic G_0W_0 implementation with numerical atomic orbital basis functions: Algorithm and benchmarks. *Phys. Rev. Mater.* **2021**, *5*, 013807.
(99) *The Elk code*; https://elk.sourceforge.io (accessed Mar 8, 2022).
(100) Gulans, A.; Kontur, S.; Meisenbichler, C.; Nabok, D.; Pavone, P.; Rigamonti, S.; Sagmeister, S.; Werner, U.; Drax, C. Exciting: A full-potential all-electron package implementing density-functional theory and many-body perturbation theory. *J. Phys.: Condens. Matter* **2014**, *26*, 363202.
(101) Jiang, H.; Gómez-Abal, R. I.; Li, X.-Z.; Meisenbichler, C.; Ambrosch-Draxl, C.; Scheffler, M. FHI-gap: A GW code based on the all-electron augmented plane wave method. *Comput. Phys. Commun.* **2013**, *184*, 348−366.
(102) Tiago, M. L.; Chelikowsky, J. R. Optical excitations in organic molecules, clusters, and defects studied by first-principles Green's function methods. *Phys. Rev. B* **2006**, *73*, 205334.
(103) Giannozzi, P.; Baroni, S.; Bonini, N.; Calandra, M.; Car, R.; Cavazzoni, C.; Ceresoli, D.; Chiarotti, G. L.; Cococcioni, M.; Dabo, I.; Dal Corso, A.; de Gironcoli, S.; Fabris, S.; Fratesi, G.; Gebauer, R.; Gerstmann, U.; Gougoussis, C.; Kokalj, A.; Lazzeri, M.; Martin-Samos, L.; Marzari, N.; Mauri, F.; Mazzarello, R.; Paolini, S.; Pasquarello, A.; Paulatto, L.; Sbraccia, C.; Scandolo, S.; Sclauzero, G.; Seitsonen, A. P.; Smogunov, A.; Umari, P.; Wentzcovitch, R. M. Quantum ESPRESSO: a modular and open-source software project for quantum simulations of materials. *J. Phys.: Condens. Matter* **2009**, *21*, 395502.
(104) Giannozzi, P.; Andreussi, O.; Brumme, T.; Bunau, O.; Nardelli, M. B.; Calandra, M.; Car, R.; Cavazzoni, C.; Ceresoli, D.; Cococcioni, M.; Colonna, N.; Carnimeo, I.; Corso, A. D.; de Gironcoli, S.; Delugas, P.; Jr, R. A. D.; Ferretti, A.; Floris, A.; Fratesi, G.; Fugallo, G.; Gebauer, R.; Gerstmann, U.; Giustino, F.; Gorni, T.; Jia, J.; Kawamura, M.; Ko, H.-Y.; Kokalj, A.; Küçükbenli, E.; Lazzeri, M.; Marsili, M.; Marzari, N.; Mauri, F.; Nguyen, N. L.; Nguyen, H.-V.; de-la Roza, A. O.; Paulatto, L.; Poncé, S.; Rocca, D.; Sabatini, R.; Santra, B.; Schlipf, M.; Seitsonen, A. P.; Smogunov, A.; Timrov, I.; Thonhauser, T.; Umari, P.; Vast, N.; Wu, X.; Baroni, S. Advanced capabilities for materials modelling with Quantum ESPRESSO. *J. Phys.: Condens. Matter* **2017**, *29*, 465901.

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707

(105) Davidson, E. R. The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices. *J. Comput. Phys.* **1975**, *17*, 87−94.
(106) Ma, H.; Govoni, M.; Gygi, F.; Galli, G. A finite-field approach for GW calculations beyond the random phase approximation. *J. Chem. Theory Comput.* **2019**, *15*, 154−164.
(107) Govoni, M.; Whitmer, J.; de Pablo, J.; Gygi, F.; Galli, G. Code interoperability extends the scope of quantum simulations. *npj Comput. Mater.* **2021**, *7*, 32.
(108) Sternheimer, R. M. Electronic polarizabilities of ions from the Hartree-Fock wave functions. *Phys. Rev.* **1954**, *96*, 951.
(109) *CUDA Fortran*; https://developer.nvidia.com/cuda-fortran (accessed May 23, 2022).
(110) *The OpenMP API specification for parallel programming*; https://www.openmp.org (accessed May 23, 2022).
(111) Ayala, A.; Tomov, S.; Haidar, A.; Dongarra, J. heFFTe: Highly efficient FFT for exascale. *Computational Science − ICCS 2020. ICCS 2020. Lecture Notes in Computer Science*; Springer: Cham, 2020; Vol. *12137*, pp 262−275; DOI: 10.1007/978-3-030-50371-0_19.
(112) Foley, D.; Danskin, J. Ultra-performance Pascal GPU and NVLink interconnect. *IEEE Micro* **2017**, *37*, 7−17.
(113) Yu, V. W.-z.; Moussa, J.; Kůs, P.; Marek, A.; Messmer, P.; Yoon, M.; Lederer, H.; Blum, V. GPU-acceleration of the ELPA2 distributed eigensolver for dense symmetric and hermitian eigenproblems. *Comput. Phys. Commun.* **2021**, *262*, 107808.
(114) *SLATE*; https://icl.utk.edu/slate (accessed May 23, 2022).
(115) *Using the cuSOLVER-MG API*; https://docs.nvidia.com/cuda/cusolver/index.html#mg-api (accessed May 23, 2022).
(116) The Perlmutter supercomputer has two phases: phase I with GPU-accelerated nodes and phase II with CPU-only nodes. Throughout this section, "Perlmutter" refers to the GPU-accelerated nodes in phase I.
(117) Schlipf, M.; Gygi, F. Optimization algorithm for the generation of ONCV pseudopotentials. *Comput. Phys. Commun.* **2015**, *196*, 36−44.
(118) Hamann, D. R. Optimized norm-conserving Vanderbilt pseudopotentials. *Phys. Rev. B* **2013**, *88*, 085117.
(119) Perdew, J. P.; Burke, K.; Ernzerhof, M. Generalized gradient approximation made simple. *Phys. Rev. Lett.* **1996**, *77*, 3865−3868.
(120) Pham, T. A.; Lee, D.; Schwegler, E.; Galli, G. Interfacial effects on the band edges of functionalized Si surfaces in liquid water. *J. Am. Chem. Soc.* **2014**, *136*, 17071−17077.
(121) Kroupa, D. M.; Pach, G. F.; Vörös, M.; Giberti, F.; Chernomordik, B. D.; Crisp, R. W.; Nozik, A. J.; Johnson, J. C.; Singh, R.; Klimov, V. I.; Galli, G.; Beard, M. C. Enhanced multiple exciton generation in PbSlCdS Janus-like heterostructured nanocrystals. *ACS Nano* **2018**, *12*, 10084−10094.
(122) Pham, T. A.; Li, T.; Nguyen, H.-V.; Shankar, S.; Gygi, F.; Galli, G. Band offsets and dielectric properties of the amorphous Si_3N_4/Si(100) interface: A first-principles study. *Appl. Phys. Lett.* **2013**, *102*, 241603.
(123) In *Properties of advanced semiconductor materials: GaN, AIN, InN, BN, SiC, SiGe*; Levinshtein, M. E., Rumyantsev, S. L., Shur, M. S., Eds.; John Wiley & Sons: New York, 2001.
(124) The exciton binding energy and ionic relaxation energy are estimated from the results of three DFT calculations using the DDH functional. The occupations of the KS orbitals are constrained in these calculations. The first calculation uses the structure and occupations of the $^3A_2$ ground state. We obtain the total energy of the $^3A_2$ state, E_{tot}, and the difference between the e and the a_1 defect states in the minority spin channel, $e - a_1$. The second calculation uses the structure of the $^3A_2$ state and the occupations of the ^3E excited state, resulting in a total energy E'_{tot}. The total energy difference, $E'_{tot} - E_{tot}$, corresponds to the vertical neutral excitation energy from the $^3A_2$ state to the ^3E state. The exciton binding energy E_{bind} is estimated as $(e - a_1) - (E'_{tot} - E_{tot})$. The third calculation uses the structure and occupations of the ^3E state. We obtain the total energy of the ^3E state, E''_{tot}. The total energy difference, $E''_{tot} - E_{tot}$, corresponds to the ionic relaxation energy E_{relax}.
(125) Skone, J. H.; Govoni, M.; Galli, G. Nonempirical range-separated hybrid functionals for solids and molecules. *Phys. Rev. B* **2016**, *93*, 235106.
(126) Falk, A. L.; Buckley, B. B.; Calusine, G.; Koehl, W. F.; Dobrovitski, V. V.; Politi, A.; Zorman, C. A.; Feng, P. X.-L.; Awschalom, D. D. Polytype control of spin qubits in silicon carbide. *Nat. Commun.* **2013**, *4*, 1819.
(127) Vinson, J. Faster exact exchange in periodic systems using single-precision arithmetic. *J. Chem. Phys.* **2020**, *153*, 204106.
(128) Govoni, M.; Munakami, M.; Tanikanti, A.; Skone, J. H.; Runesha, H. B.; Giberti, F.; de Pablo, J.; Galli, G. Qresp, a tool for curating, discovering and exploring reproducible scientific papers. *Sci. Data* **2019**, *6*, 190002.

https://doi.org/10.1021/acs.jctc.2c00241
J. Chem. Theory Comput. 2022, 18, 4690−4707