Original software publication

SPARC: Simulation Package for Ab-initio Real-space Calculations

Qimen Xu [a], Abhiraj Sharma [a], Benjamin Comer [a], Hua Huang [b], Edmond Chow [b], Andrew J. Medford [a], John E. Pask [c], Phanish Suryanarayana [a,*]

[a] *College of Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA*
[b] *College of Computing, Georgia Institute of Technology, Atlanta, GA 30332, USA*
[c] *Physics Division, Lawrence Livermore National Laboratory, Livermore, CA 94550, USA*

ARTICLE INFO

ABSTRACT

We present SPARC: Simulation Package for Ab-initio Real-space Calculations. SPARC can perform Kohn–Sham density functional theory calculations for isolated systems such as molecules as well as extended systems such as crystals and surfaces, in both static and dynamic settings. It is straightforward to install/use and highly competitive with state-of-the-art planewave codes, demonstrating comparable performance on a small number of processors and increasing advantages as the number of processors grows. Notably, SPARC brings solution times down to a few seconds for systems with $\mathcal{O}(100\text{--}500)$ atoms on large-scale parallel computers, outperforming planewave counterparts by an order of magnitude and more.

Code Metadata

Current code version	v1.0.0
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX_2020_221
Code Ocean compute capsule	
Legal Code License	GNU General Public License v3.0
Code versioning system used	git
Software code languages, tools, and services used	C, MPI, BLAS, LAPACK, ScaLAPACK (optional), MKL (optional)
Compilation requirements, operating environments & dependencies	OS: Unix, Linux, or MacOS
If available Link to developer documentation/manual	https://github.com/SPARC-X/SPARC/tree/master/doc
Support email for questions	phanish.s@gmail.com

1. Motivation and significance

Over the course of the past few decades, quantum mechanical calculations based on Kohn–Sham density functional theory (DFT) [1,2] have become a cornerstone of materials research by virtue of the predictive power and fundamental insights they provide. The widespread use of the methodology can be attributed to its generality, simplicity, and high accuracy-to-cost ratio relative to other such ab initio approaches [3,4]. However, while less expensive than wavefunction based methods, the solution of the Kohn–Sham equations remains a formidable task. In particular, the computational cost scales cubically with the number of atoms, severely limiting the range of physical systems accessible to such first principles investigation. These limitations become even more acute in quantum molecular dynamics (QMD) simulations, wherein the equations for the electronic ground state may be solved tens or hundreds of thousands of times to reach time scales relevant to phenomena of interest [3].

The planewave pseudopotential method [5] has been among the most widely used techniques for the solution of the Kohn–Sham problem [6–13]. The underlying Fourier basis is complete, orthonormal, independent of atomic positions, diagonalizes the Laplacian, and provides spectral convergence for smooth problems. As a result, the planewave method is accurate, simple to use since it relies on a single convergence parameter, has negligible egg box effect [14,15], and is highly efficient on moderate computational resources with the use of well optimized Fast Fourier Transforms (FFTs) and efficient preconditioning schemes. However, the Fourier basis restricts the method to periodic boundary conditions, whereby finite systems such as clusters and

---

* Corresponding author.
  E-mail address: phanish.suryanarayana@ce.gatech.edu (P. Suryanarayana).

molecules, as well as semi-infinite systems such as surfaces and nanowires, require the introduction of artificial periodicity with large vacuum regions. This limitation also necessitates the introduction of an unphysical neutralizing background density when treating charged systems in order to avoid Coulomb divergences. Moreover, the global nature of the Fourier basis hampers scalability on parallel computing platforms and complicates the development of linear-scaling methods [16–18], limiting the system sizes and time scales accessible.

The limitations of the planewave method have motivated the development of a number of alternative solution strategies employing systematically improvable, localized representations [19–39]. Among these, perhaps the most mature and widely used to date are the finite-difference methods [40,41], wherein computational locality is maximized by discretizing all quantities of interest on a uniform real-space grid that is independent of atomic positions.[1] As a result, convergence is controlled by a single parameter and both periodic and Dirichlet boundary conditions are naturally accommodated, thus enabling the efficient and accurate treatment of finite, semi-infinite, bulk, and charged systems alike. Moreover, real-space methods are amenable to the development of linear scaling methods, and large-scale parallel computational resources can be efficiently leveraged by virtue of the method's simplicity, locality, and freedom from communication-intensive transforms such as FFTs [24,35,43–45]. With these and other advances, real-space methods have been applied to systems containing thousands of atoms, and have demonstrated substantially reduced solution times compared to established planewave codes in applications to both finite [46] and extended [36] systems.

However, despite the significant advantages afforded by real-space methods, the planewave method has remained the method of choice in practice for the better part of the past two decades. This is largely due to the ease of use, extensive feature sets, established accuracy/robustness, and straightforward installation of the associated codes, having been in development and production for a longer period of time. Perhaps most importantly, however, planewave codes have typically yielded shorter times to solution using moderate computational resources, as most widely available to researchers in practice [36,46]. Moreover, even with access to larger-scale machines, real-space codes have not always yielded shorter times to solution, further hindering wider adoption in practice.

In this work, we present an open-source software package for the accurate, efficient, and scalable solution of the Kohn–Sham equations, referred to as SPARC. The package is straightforward to install/use and highly competitive with state-of-the-art planewave codes, demonstrating comparable performance on a small number of processors and order-of-magnitude advantages as the number of processors increases.

2. Software description

The central focus of SPARC is the accurate and efficient solution of the finite-temperature Kohn–Sham equations for the electronic ground state [1,2,47]:

$$\left(\mathcal{H}^\sigma \equiv -\frac{1}{2}\nabla^2 + V_{\text{eff}}^\sigma\left[\rho^\alpha, \rho^\beta; \mathbf{R}\right]\right)\psi_n^\sigma = \lambda_n^\sigma \psi_n^\sigma,$$
$$n = 1, 2, \ldots, N_s^\sigma, \quad \sigma \in \{\alpha, \beta\}, \tag{1}$$

---

[1] Nonuniform adaptive grids [42] have the potential to reduce number of degrees of freedom significantly. However, they introduce additional parameters to be tuned, additional computations to effect transformations, atom-position dependence and associated Pulay forces, and complicate load balancing in parallel computations.

where the superscript σ denotes the spin, i.e., spin-up or spin-down, $\mathcal{H}^\sigma$ is the Hamiltonian, ψ_n^σ are the orthonormal orbitals with energies λ_n^σ, V_{eff}^σ is the effective potential, N_s^σ is the number of states, and $\mathbf{R}$ denotes the set of atomic positions. In addition, ρ^σ represents the spin-resolved electron density:

$$\rho^\sigma(\mathbf{x}) = \sum_{n=1}^{N_s^\sigma} g_n^\sigma |\psi_n^\sigma(\mathbf{x})|^2, \quad \sigma \in \{\alpha, \beta\}, \quad \mathbf{x} \in \mathbb{R}^3, \tag{2}$$

where g_n^σ are the orbital occupations, typically given by the Fermi–Dirac distribution. In implementations of the above equations, once a suitable fundamental domain/unit cell has been identified, zero Dirichlet or Bloch-periodic boundary conditions are prescribed on the orbitals along the directions in which the system is finite or extended, respectively.

2.1. Software architecture

SPARC employs the pseudopotential approximation [5] to facilitate the efficient solution of the Kohn–Sham equations for the whole of the periodic table of elements. In addition, it employs a local real-space formulation of the electrostatics [48,49], wherein the electrostatic potential – component of V_{eff}^σ that is the sum of ionic and Hartree contributions – is given by the solution of a Poisson problem, with Dirichlet or periodic boundary conditions prescribed along directions in which the system is finite or extended, respectively. In this framework, SPARC performs a uniform real-space discretization of the equations, using a high-order centered finite-difference approximation for differential operators and the trapezoidal rule for integral operators. The actual code is written in the C language and achieves parallelism through the message passing interface (MPI) [50]. An overview of the SPARC framework for performing Kohn–Sham DFT calculations is shown in Fig. 1.

SPARC can perform single-point calculations, structural relaxations (atom and/or cell), and QMD simulations. For single-point calculations, the electronic ground state is determined for fixed ionic positions and cell dimensions, whereas for structural relaxations, positions and/or cell dimensions are varied to minimize the Kohn–Sham energy using the Hellmann–Feynman atomic forces [36,46] and/or stress tensor [51]. For QMD, the ionic positions, velocities, and accelerations are evolved by integrating the equations of motion, with or without a thermostat, using the atomic forces. In all cases, the calculations can be either spin-polarized or unpolarized, with various choices of local and semilocal exchange–correlation functionals.

SPARC requires two input files for every calculation: (i) a `.inpt` file containing the options and parameters to be used in the calculation, including the choice of exchange–correlation functional, flag for spin-polarization, type of static/dynamic calculation, ionic temperature in the case of QMD, cell dimensions, boundary condition in each direction, and finite-difference grid specification; and (ii) a `.ion` file containing information on the atomic configuration, including atom types, positions, and paths to corresponding pseudopotential files. Note that, in order to enable detailed control of the simulation, a large number of parameters and options can be specified, as described in the accompanying user guide. However, by virtue of carefully chosen defaults, relatively few parameters typically need be specified in practice. Note also that, since all files are simple, human-readable text, series of simulations are readily scripted. A Python package containing helper functions for generating input files and submitting simulations is also available.

The Kohn–Sham problem for the electronic ground state needs to be solved for every configuration of atoms encountered during the DFT simulation. In SPARC, this is achieved using the self-consistent field (SCF) method [5], which represents a fixed-point
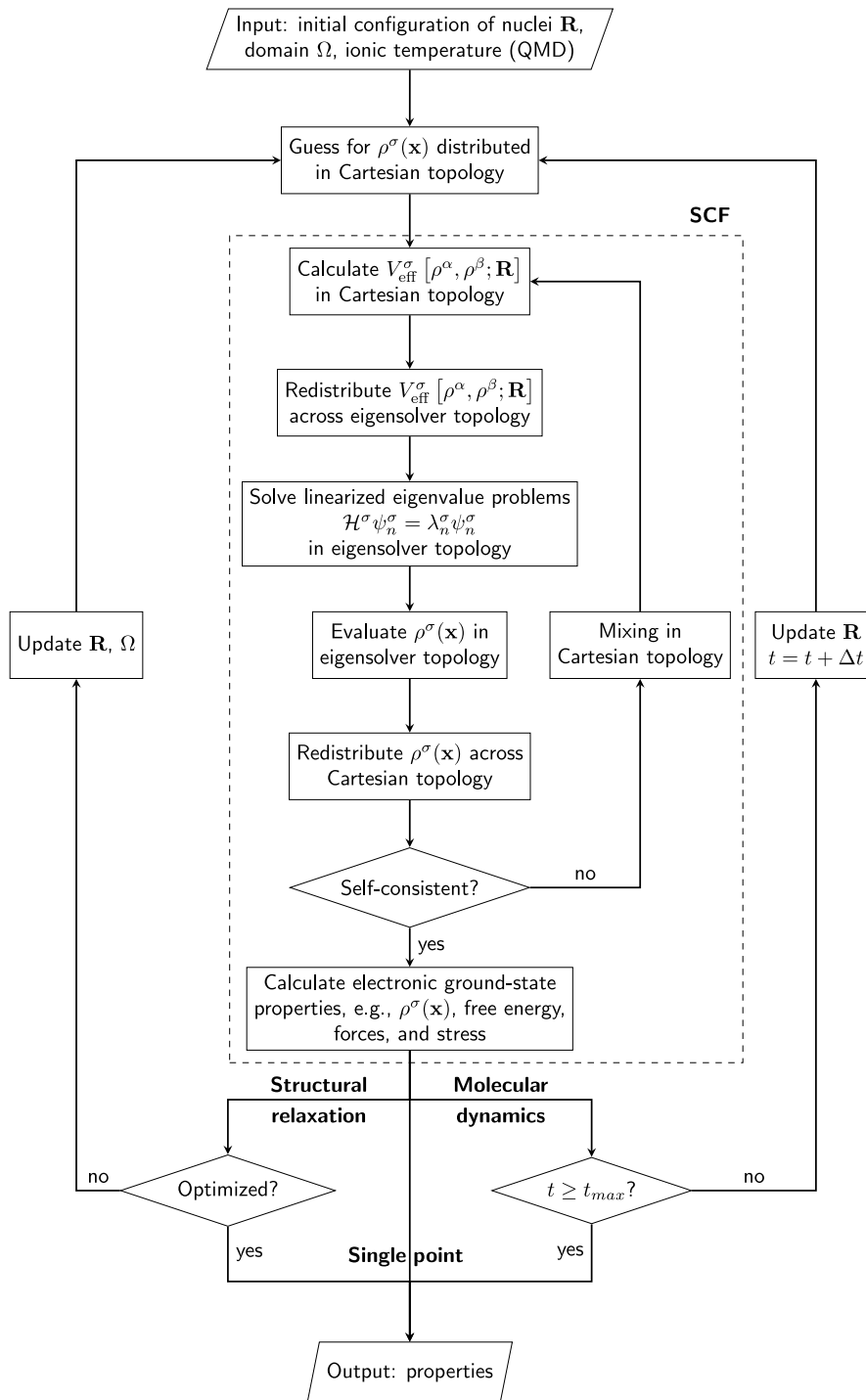
Fig. 1. Overview of the SPARC framework for performing Kohn–Sham DFT calculations. The Cartesian topology is formed by embedding a three-dimensional processor grid into the `MPI_COMM_WORLD` communicator. The eigensolver topology is a collection of smaller Cartesian topologies, created by first splitting the `MPI_COMM_WORLD` communicator into multiple spin groups, then splitting each spin group into multiple Brillouin zone integration groups, then splitting each Brillouin zone integration group into multiple band groups, and finally, embedding each band group with a Cartesian topology.

iteration with respect to either the electron density or potential. For the first SCF iteration in the simulation, a superposition of isolated atom electron densities is used as the initial guess, whereas for every subsequent atomic configuration encountered, extrapolation based on solutions to previous configurations is employed [52]. The convergence of the SCF iteration is accelerated using the restarted variant [53] of the Periodic Pulay mixing scheme [54] with a real-space preconditioner [55]. In the case

of spin-polarized calculations, mixing is performed simultaneously on both components, i.e., on a vector of twice the original length containing both spin-up and spin-down density/potential components.

In each SCF iteration, SPARC performs a partial diagonalization (i.e., eigenvalues and eigenvectors calculated approximately) of the linear eigenproblem using the Chebyshev filtered subspace iteration (CheFSI) [56,57], with multiple Chebyshev filtering
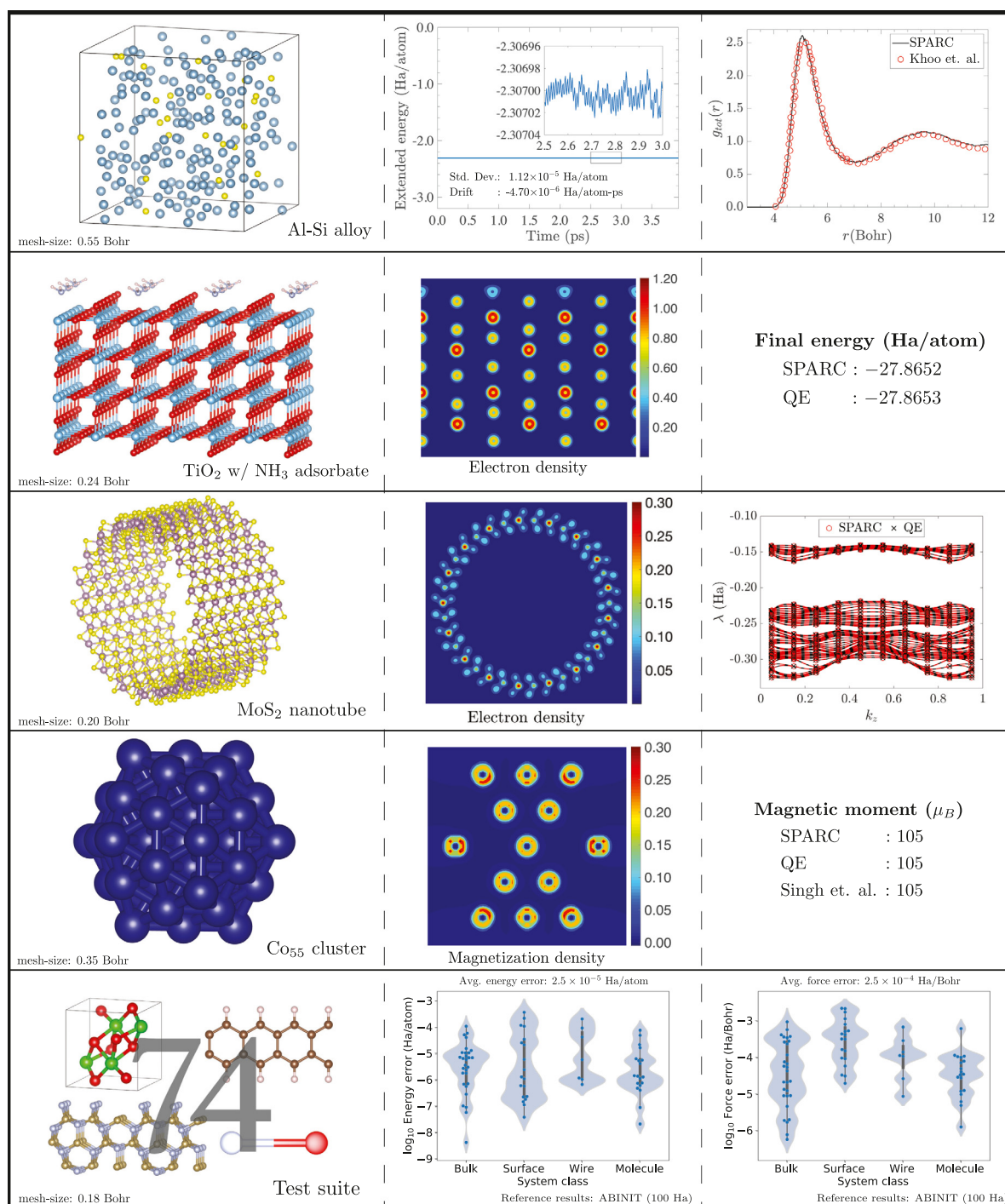
Fig. 2. Examples demonstrating the major functionalities of SPARC.

steps performed in the first iteration of the simulation [58]. The Hamiltonian-matrix/vector products are performed in matrix-free fashion, using the finite-difference stencil for the Laplacian and the outer product nature of the nonlocal pseudopotential operator. While doing so, zero-Dirichlet or Bloch-periodic boundary conditions are prescribed on the orbitals along directions in which the system is finite or extended, respectively. In calculating the effective potential, the Poisson problem for the electrostatic potential is solved using the alternating Anderson–Richardson (AAR) method [59,60], with Laplacian-vector products again performed in matrix-free fashion using the finite-difference stencil. While doing so, Dirichlet or periodic boundary conditions are

prescribed on the electrostatic potential along directions in which the system is finite or extended, respectively. In particular, Dirichlet values are determined using a multipole expansion for isolated systems and a dipole correction for surfaces and nanowires [61, 62].

In SPARC, information pertaining to the overall DFT simulation is written to the `.out` file, including progression of the SCF iteration, electronic ground state energy, maximum atomic force, maximum stress, and various timings. Based on the type of calculation, a `.static`, `.geopt`, or `.aimd` file may also be written. The `.static` file contains information about the single-point calculation, including atom positions, electronic ground state energy,
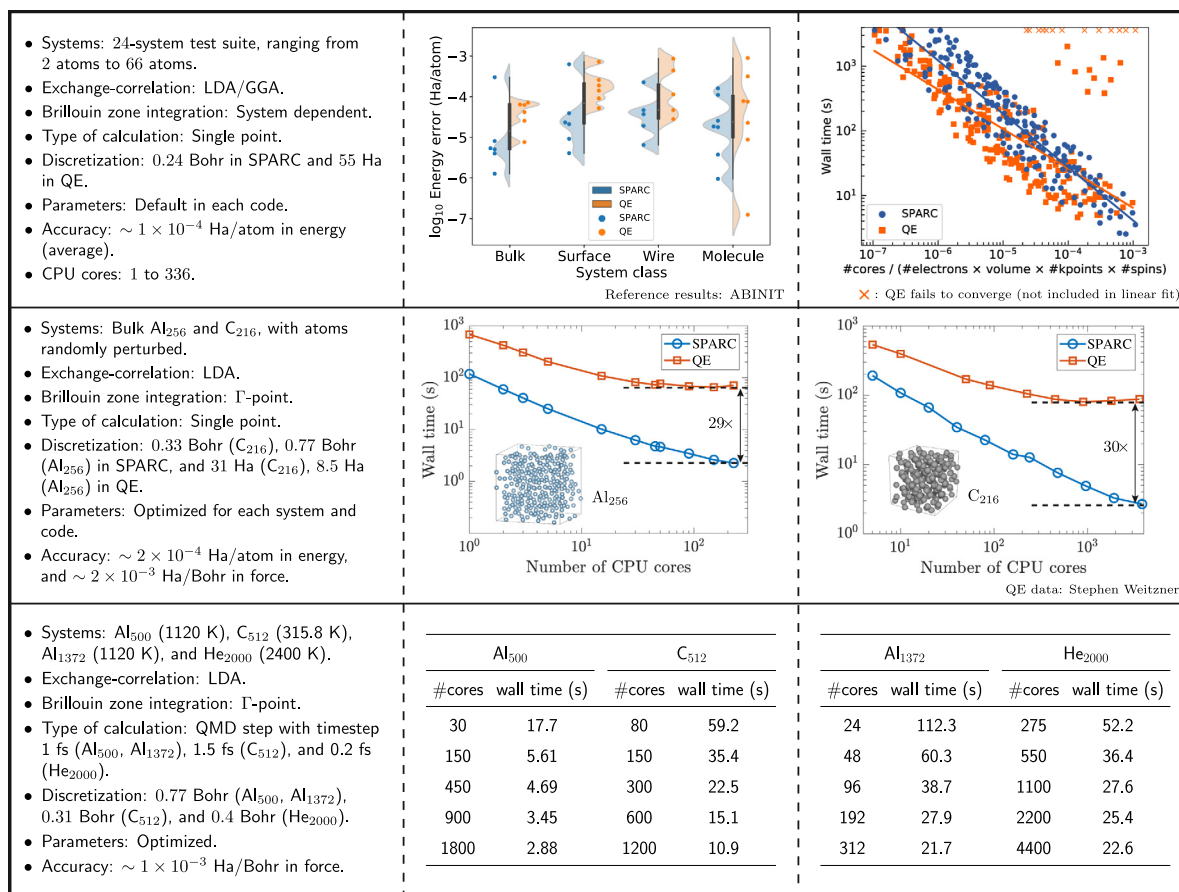
- Systems: 24-system test suite, ranging from 2 atoms to 66 atoms.
- Exchange-correlation: LDA/GGA.
- Brillouin zone integration: System dependent.
- Type of calculation: Single point.
- Discretization: 0.24 Bohr in SPARC and 55 Ha in QE.
- Parameters: Default in each code.
- Accuracy: $\sim 1 \times 10^{-4}$ Ha/atom in energy (average).
- CPU cores: 1 to 336.

Reference results: ABINIT

$\times$: QE fails to converge (not included in linear fit)

- Systems: Bulk Al_{256} and C_{216}, with atoms randomly perturbed.
- Exchange-correlation: LDA.
- Brillouin zone integration: Γ-point.
- Type of calculation: Single point.
- Discretization: 0.33 Bohr (C_{216}), 0.77 Bohr (Al_{256}) in SPARC, and 31 Ha (C_{216}), 8.5 Ha (Al_{256}) in QE.
- Parameters: Optimized for each system and code.
- Accuracy: $\sim 2 \times 10^{-4}$ Ha/atom in energy, and $\sim 2 \times 10^{-3}$ Ha/Bohr in force.

QE data: Stephen Weitzner

- Systems: Al_{500} (1120 K), C_{512} (315.8 K), Al_{1372} (1120 K), and He_{2000} (2400 K).
- Exchange-correlation: LDA.
- Brillouin zone integration: Γ-point.
- Type of calculation: QMD step with timestep 1 fs (Al_{500}, Al_{1372}), 1.5 fs (C_{512}), and 0.2 fs (He_{2000}).
- Discretization: 0.77 Bohr (Al_{500}, Al_{1372}), 0.31 Bohr (C_{512}), and 0.4 Bohr (He_{2000}).
- Parameters: Optimized.
- Accuracy: $\sim 1 \times 10^{-3}$ Ha/Bohr in force.

Al_{500}		C_{512}		Al_{1372}		He_{2000}	
#cores	wall time (s)	#cores	wall time (s)	#cores	wall time (s)	#cores	wall time (s)
30	17.7	80	59.2	24	112.3	275	52.2
150	5.61	150	35.4	48	60.3	550	36.4
450	4.69	300	22.5	96	38.7	1100	27.6
900	3.45	600	15.1	192	27.9	2200	25.4
1800	2.88	1200	10.9	312	21.7	4400	22.6

Fig. 3. Examples demonstrating the performance of SPARC.

forces, and stress tensor. The `.geopt` file contains information about the structural relaxation, including (i) atom positions, electronic ground state energy, and forces (atomic relaxation), and (ii) cell information and stress tensor (cell relaxation). The `.aimd` file contains information about the QMD simulation, including atom positions, forces, and velocities. To seamlessly continue from a previously stopped simulation, a `.restart` file is written for structural relaxation and QMD calculations. SPARC provides other outputs if specified as well, e.g., a `.eigen` file containing eigenvalues and occupations and `.dens` file containing the charge density.

2.2. Software functionalities

The current version of SPARC is capable of performing spin-polarized and unpolarized ab initio calculations based on Kohn–Sham DFT for isolated systems such as molecules as well as extended systems such as crystals, surfaces, and nanowires, in both static and dynamic settings. Specifically, it can perform single-point calculations for a given atomic configuration, structural relaxations with respect to atom positions and/or cell dimensions [63–66], and NVE/NVT/NVK QMD simulations [67–69]. Available exchange–correlation functionals include various forms of local density approximation (LDA) [70,71] and generalized gradient approximation (GGA) [72–74]. Types of pseudopotentials employed are optimized norm conserving Vanderbilt (ONCV) [75] and Troullier–Martins [76], both in `psp8` format [8]. Over the course of simulations, in addition to electronic density and free energy, SPARC can calculate atomic forces, pressure, and the stress tensor for extended systems. The outputs from such DFT calculations can be used to calculate a number of properties, including lattice constant, cohesive energy, polarization, elastic moduli, density of states, electronic band structure, pair distribution function, equations of state, shear viscosity, defect energy, surface energy, absorption energy, equilibrium bond lengths, HOMO–LUMO gap, and dipole moment.

3. Illustrative examples

We now demonstrate the major functionalities of SPARC through examples representative of physical applications. Specifically, we consider (i) 200-atom NVT QMD simulation for liquid $Al_{88}Si_{12}$ alloy at 973 K, with LDA and Γ-point for Brillouin zone integration; (ii) structural atomic relaxation for a 52-atom system modeling a NH_3 adsorbate on a (110) TiO_2 surface with GGA and 4 × 4 grid for Brillouin zone integration; (iii) structural cell relaxation for a 102-atom MoS_2 nanotube of diameter 3 nm with GGA and 10 points for Brillouin zone integration; (iv) single-point calculation of a 55-atom icosahedral Co nanoparticle with GGA and spin polarization; and (v) single-point calculations for a 74-system test suite containing isolated systems such as clusters and molecules as well as extended systems such as crystals, surfaces, and nanowires, ranging from 2 to 204 atoms, encompassing 48 different chemical elements and spin-polarized as well as unpolarized calculations.

We employ the Perdew–Zunger parametrization for LDA [70], PBE variant for GGA [72], and ONCV pseudopotentials [75,77]. We present the results in Fig. 2 [78] (illustrations using VESTA [79]) and compare them to established planewave codes Quantum Espresso (QE) [9] and ABINIT [8], as well results from the literature [80,81]. It is clear that there is excellent agreement between SPARC and established planewave codes, with errors substantially

smaller than required in typical applications. Note that SPARC demonstrates excellent energy conservation with negligible drift in QMD simulations, in agreement with previous results [36], further verifying the accuracy of atomic forces computed. Indeed, the accuracy/quality of the results is further increased as the discretization is refined in SPARC. Overall, these examples demonstrate the capability of SPARC to obtain highly accurate results for a broad range of system compositions, configurations, and dimensionalities.

4. Impact

Kohn–Sham DFT simulations occupy a large fraction of high-performance computing resources around the world every day [82,83], a consequence of the unique insights and robust predictions they have been shown to provide. The majority of these calculations are performed using established planewave codes [6–10,13]. Therefore, any new implementation that is able to consistently outperform these state-of-the-art DFT codes, thereby enabling the ab initio investigation of larger length and time scales than previously accessible, with the accuracy required, stands to have significant and immediate impact. This is particularly true of a code like SPARC that is open-source with minimal dependencies so that it can be easily installed on computers large and small around the world.

Accordingly, we compare the accuracy and efficiency of SPARC to Quantum Espresso (QE) [9], an established state-of-the-art planewave DFT code. We employ the same pseudopotentials [75, 77] and exchange–correlation functionals [70,72] in both codes. Results and computational parameters for the study, containing a wide range of system sizes, are shown in Fig. 3. [78] It is clear that SPARC demonstrates comparable performance to QE on a small number of processors and increasing advantages as the number of processors grows. In particular, SPARC brings solution times down to a few seconds for systems with $\mathcal{O}(100–500)$ atoms on large-scale parallel computers, outperforming QE by more than an order of magnitude. Furthermore, SPARC achieves wall times per QMD step of just over 20 seconds for the largest systems containing more than a thousand atoms, achieved on only 312 cores for Al_{1372}. For such systems and larger, the ability of SPARC to efficiently scale to many thousands of processors and more is currently limited by the subspace diagonalization step performed in each SCF iteration, which due to its cubic scaling and limited parallel scalability takes a larger fraction of wall time as the system size grows.

Going forward, we plan to first implement a structure-adapted eigensolver in SPARC to push back the cubic-scaling bottleneck, and then the Discrete Discontinuous Basis Projection (DDBP) method [26] to enable strong scaling of SPARC to still larger numbers of processors, bringing down time to solution still further. The DDBP method will also enable efficient DFT calculations with hybrid functionals and the linear-scaling Spectral Quadrature (SQ) method [45,84], which will be implemented subsequently. In order to enable the effective use of exascale computing platforms, a parallel engine for SPARC that enables highly efficient distributed memory communication and offloading to GPUs will be completed. Moreover, machine-learning methods will be explored to improve efficiency still further. Along with these developments, we plan to implement cyclic and helical symmetry-adapted DFT formulations that allow for the highly efficient study of associated mechanical deformations as well as systems with such symmetries [85–88]; and a coarse-grained DFT formulation that enables the study of crystal defects at realistic concentrations [89]. Indeed, many of these developments will be accelerated by using the M-SPARC code [90]—same structure, algorithms, input, and output as SPARC—for rapid prototyping.

SPARC and its variants are currently being used by multiple research groups. Moving forward, the user base is expected to grow, given the current open-source distribution, simplicity of installation and use, high accuracy, and ability to reach larger length and time scales than current state-of-the-art planewave codes. The impact thus stands to be both broad and substantial.

5. Conclusions

SPARC has now become a mature code for performing real-space Kohn–Sham DFT calculations, prompting its open-source release with this publication. Currently, it can perform pseudopotential spin-polarized and unpolarized simulations for isolated systems such as molecules and clusters as well as extended systems such as crystals, surfaces, and nanowires, in both static and dynamic settings. SPARC is not only highly accurate, but also highly competitive with established state-of-the-art planewave codes on modest computational resources, with increasing advantages as the number of processors increases. In particular, SPARC efficiently scales to thousands of processors, bringing solution times for moderate-sized systems consisting of $\mathcal{O}(100–500)$ atoms to within a few seconds, making it an attractive choice for QMD simulations in particular. Given its superior scalability, and ability to incorporate attractive features such as linear scaling methods and variety of boundary conditions, SPARC has the potential to enable a number of new and exciting applications in science and engineering that were previously beyond reach.

Declaration of competing interest

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

References

[1] Kohn W, Sham LJ. Self-consistent equations including exchange and correlation effects. Phys Rev 1965;140(4A):A1133–8.
[2] Hohenberg P, Kohn W. Inhomogeneous electron gas. Phys Rev 1964;136(3B):B864.
[3] Burke K. Perspective on density functional theory. J Chem Phys 2012;136:150901.
[4] Becke AD. Perspective: Fifty years of density-functional theory in chemical physics. J Chem Phys 2014;140:18A301.
[5] Martin R. Electronic structure: basic theory and practical methods. Cambridge University Press; 2004.
[6] Kresse G, Furthmüller J. Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set. Phys Rev B 1996;54(16):11169–86.
[7] Clark SJ, Segall MD, Pickard CJ, Hasnip PJ, Probert MI, Refson K, et al. First principles methods using CASTEP. Z Kristallogr-Cryst Mater 2005;220(5/6):567–70.

[8] Gonze X, Beuken JM, Caracas R, Detraux F, Fuchs M, Rignanese GM, et al. First-principles computation of material properties: the ABINIT software project. Comput Mater Sci 2002;25. 478–492(15).

[9] Giannozzi P, Baroni S, Bonini N, Calandra M, Car R, Cavazzoni C, et al. QUANTUM ESPRESSO: a modular and open-source software project for quantum simulations of materials. J Phys: Condens Matter 2009;21(39):395502, (19pp).

[10] Marx D, Hutter J. Ab initio molecular dynamics: Theory and implementation. In: Modern methods and algorithms of quantum chemistry, vol. 1. 2000, p. 301–449.

[11] Ismail-Beigi S, Arias TA. New algebraic formulation of density functional calculation. Comput Phys Comm 2000;128(1–2):1–45.

[12] Gygi F. Architecture of Qbox: A scalable first-principles molecular dynamics code. IBM J Res Dev 2008;52(1.2):137–44.

[13] Valiev M, Bylaska E, Govind N, Kowalski K, Straatsma T, Dam HV, et al. NWChem: A comprehensive and scalable open-source solution for large scale molecular simulations. Comput Phys Comm 2010;181(9):1477–89.

[14] Artacho E, Anglada E, Diéguez O, Gale JD, García A, Junquera J, et al. The SIESTA method: developments and applicability. J Phys: Condens Matter 2008;20(6):064208.

[15] Ono T, Heide M, Atodiresei N, Baumeister P, Tsukamoto S, Blügel S. Real-space electronic structure calculations with full-potential all-electron precision for transition metals. Phys. Rev. B 2010;82:205115.

[16] Goedecker S. Linear scaling electronic structure methods. Rev Modern Phys 1999;71(4):1085–123.

[17] Bowler DR, Miyazaki T. O(N) methods in electronic structure calculations. Rep Progr Phys 2012;75(3):036503.

[18] Aarons J, Sarwar M, Thompsett D, Skylaris C-K. Perspective: Methods for large-scale density functional calculations on metallic systems. J Chem Phys 2016;145(22):220901.

[19] Becke AD. Basis-set-free density-functional quantum chemistry. Int J Quantum Chem 1989;36(S23):599–609.

[20] Chelikowsky JR, Troullier N, Saad Y. Finite-difference-pseudopotential method: electronic structure calculations without a basis. Phys Rev Lett 1994;72(8):1240.

[21] Genovese L, Neelov A, Goedecker S, Deutsch T, Ghasemi SA, Willand A, et al. Daubechies wavelets as a basis set for density functional pseudopotential calculations. J Chem Phys 2008;129(1):014109.

[22] Seitsonen AP, Puska MJ, Nieminen RM. Real-space electronic-structure calculations: Combination of the finite-difference and conjugate-gradient methods. Phys Rev B 1995;51(20):14057.

[23] White SR, Wilkins JW, Teter MP. Finite-element method for electronic structure. Phys Rev B 1989;39(9):5819.

[24] Iwata J-I, Takahashi D, Oshiyama A, Boku T, Shiraishi K, Okada S, et al. A massively-parallel electronic-structure calculations based on real-space density functional theory. J Comput Phys 2010;229(6):2339–63.

[25] Tsuchida E, Tsukada M. Electronic-structure calculations based on the finite-element method. Phys Rev B 1995;52(8):5573.

[26] Xu Q, Suryanarayana P, Pask JE. Discrete discontinuous basis projection method for large-scale electronic structure calculations. J Chem Phys 2018;149(9):094104.

[27] Suryanarayana P, Bhattacharya K, Ortiz M. A mesh-free convex approximation scheme for Kohn–Sham density functional theory. J Comput Phys 2011;230(13):5226–38.

[28] Suryanarayana P, Gavini V, Blesgen T, Bhattacharya K, Ortiz M. Non-periodic finite-element formulation of Kohn–Sham density functional theory. J Mech Phys Solids 2010;58(2):256–80.

[29] Skylaris C-K, Haynes PD, Mostofi AA, Payne MC. Introducing ONETEP: Linear-scaling density functional simulations on parallel computers. J Chem Phys 2005;122(8):084119.

[30] Bowler DR, Choudhury R, Gillan MJ, Miyazaki T. Recent progress with large-scale ab initio calculations: the CONQUEST code. Phys Status Solidi b 2006;243(5):989–1000.

[31] Motamarri P, Das S, Rudraraju S, Ghosh K, Davydov D, Gavini V. DFT-FE — A massively parallel adaptive finite-element code for large-scale density functional theory calculations. Comput Phys Comm 2020;246:106853.

[32] Castro A, Appel H, Oliveira M, Rozzi CA, Andrade X, Lorenzen F, et al. octopus: a tool for the application of time-dependent density functional theory. Phys Status Solidi b 2006;243(11):2465–88.

[33] Briggs EL, Sullivan DJ, Bernholc J. Real-space multigrid-based approach to large-scale electronic structure calculations. Phys Rev B 1996;54:14362–75.

[34] Fattebert J-L. Finite difference schemes and block Rayleigh quotient iteration for electronic structure calculations on composite grids. J Comput Phys 1999;149(1):75–94.

[35] Shimojo F, Kalia RK, Nakano A, Vashishta P. Linear-scaling density-functional-theory calculations of electronic structure based on real-space grids: design, analysis, and scalability test of parallel algorithms. Comput Phys Comm 2001;140(3):303–14.

[36] Ghosh S, Suryanarayana P. SPARC: Accurate and efficient finite-difference formulation and parallel implementation of density functional theory: Extended systems. Comput Phys Comm 2017;216:109–25.

[37] Arias TA. Multiresolution analysis of electronic structure: semicardinal and wavelet bases. Rev Modern Phys 1999;71(1):267–311.

[38] Pask JE, Sterne PA. Finite element methods in ab initio electronic structure calculations. Modelling Simulation Mater Sci Eng 2005;13:R71–96.

[39] Lin L, Lu J, Ying L, Weinan E. Adaptive local basis set for Kohn–Sham density functional theory in a discontinuous Galerkin framework i: Total energy calculation. J Comput Phys 2012;231(4):2140–54.

[40] Beck TL. Real-space mesh techniques in density-functional theory. Rev Modern Phys 2000;72(4):1041–80.

[41] Saad Y, Chelikowsky JR, Shontz SM. Numerical methods for electronic structure calculations of materials. SIAM Rev 2010;52(1):3–54.

[42] Gygi F, Galli G. Real-space adaptive-coordinate electronic-structure calculations. Phys Rev B 1995;52(4):R2229.

[43] Hasegawa Y, Iwata J-I, Tsuji M, Takahashi D, Oshiyama A, Minami K, et al. First-principles calculations of electron states of a silicon nanowire with 100, 000 atoms on the k computer. In: Proceedings of 2011 international conference for high performance computing, networking, storage and analysis. ACM; 2011, p. 1.

[44] Osei-Kuffuor D, Fattebert J-L. Accurate and scalable O(N) algorithm for first-principles molecular-dynamics computations on large parallel computers. Phys Rev Lett 2014;112(4):046401.

[45] Suryanarayana P, Pratapa PP, Sharma A, Pask JE. SQDFT: Spectral quadrature method for large-scale parallel O(N) Kohn–Sham calculations at high temperature. Comput Phys Comm 2018;224:288–98.

[46] Ghosh S, Suryanarayana P. SPARC: Accurate and efficient finite-difference formulation and parallel implementation of density functional theory: Isolated clusters. Comput Phys Comm 2017;212:189–204.

[47] Mermin ND. Thermal properties of the inhomogeneous electron gas. Phys Rev 1965;137:A1441–3.

[48] Suryanarayana P, Phanish D. Augmented Lagrangian formulation of orbital-free density functional theory. J Comput Phys 2014;275(0):524–38.

[49] Ghosh S, Suryanarayana P. Higher-order finite-difference formulation of periodic orbital-free density functional theory. J Comput Phys 2016;307:634–52.

[50] Gropp W, Lusk E, Skjellum A. Using MPI: portable parallel programming with the message-passing interface, vol. 1. MIT Press; 1999.

[51] Sharma A, Suryanarayana P. On the calculation of the stress tensor in real-space Kohn–Sham density functional theory. J Chem Phys 2018;149(19):194104.

[52] Alfe D. Ab initio molecular dynamics, a simple algorithm for charge extrapolation. Comput Phys Comm 1999;118(1):31–3.

[53] Pratapa PP, Suryanarayana P. Restarted Pulay mixing for efficient and robust acceleration of fixed-point iterations. Chem Phys Lett 2015;635:69–74.

[54] Banerjee AS, Suryanarayana P, Pask JE. Periodic Pulay method for robust and efficient convergence acceleration of self-consistent field iterations. Chem Phys Lett 2016;647:31–5.

[55] Kumar S, Xu Q, Suryanarayana P. On preconditioning the self-consistent field iteration in real-space density functional theory. Chem Phys Lett 2020;739:136983.

[56] Zhou Y, Saad Y, Tiago ML, Chelikowsky JR. Self-consistent-field calculations using Chebyshev-filtered subspace iteration. J Comput Phys 2006;219(1):172–84.

[57] Zhou Y, Saad Y, Tiago ML, Chelikowsky JR. Parallel self-consistent-field calculations via Chebyshev-filtered subspace acceleration. Phys Rev E 2006;74(6):066704.

[58] Zhou Y, Chelikowsky JR, Saad Y. Chebyshev-filtered subspace iteration method free of sparse diagonalization for solving the Kohn–Sham equation. J Comput Phys 2014;274:770–82.

[59] Pratapa PP, Suryanarayana P, Pask JE. Anderson acceleration of the Jacobi iterative method: An efficient alternative to Krylov methods for large, sparse linear systems. J Comput Phys 2016;306:43–54.

[60] Suryanarayana P, Pratapa PP, Pask JE. Alternating Anderson–Richardson method: An efficient alternative to preconditioned Krylov methods for large, sparse linear systems. Comput Phys Comm 2019;234:278–85.

[61] Burdick WR, Saad Y, Kronik L, Vasiliev I, Jain M, Chelikowsky JR. Parallel implementation of time-dependent density functional theory. Comput Phys Comm 2003;156(1):22–42.

[62] Natan A, Benjamini A, Naveh D, Kronik L, Tiago ML, Beckman SP, et al. Real-space pseudopotential method for first principles calculations of general periodic and partially periodic systems. Phys Rev B 2008;78(7):075109.

[63] Shewchuk JR. An introduction to the conjugate gradient method without the agonizing pain. 1994.

[64] Nocedal J. Updating quasi-newton matrices with limited storage. Math Comp 1980;35:773.

[65] Bitzek E, Koskinen P, Gähler F, Moseler M, Gumbsch P. Structural relaxation made simple. Phys Rev Lett 2006;97(17):170201.

[66] Press WH. Numerical recipes 3rd edition: the art of scientific computing. Cambridge University Press; 2007.

[67] Allen MP, Tildesley DJ. Computer simulation of liquids. Oxford University Press; 1987.

[68] Swope WC, Andersen HC, Berens PH, Wilson KR. A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters. J Chem Phys 1982;76:637.

[69] Minary P, Martyna GJ, Tuckerman ME. Algorithms and novel applications based on the isokinetic ensemble. i. biophysical and path integral molecular dynamics. J Chem Phys 2003;118:2510.

[70] Perdew JP, Zunger A. Self-interaction correction to density-functional approximations for many-electron systems. Phys Rev B 1981;23:5048–79.

[71] Perdew JP, Wang Y. Accurate and simple analytic representation of the electron-gas correlation energy. Phys Rev B 1992;45(23):13244–9.

[72] Perdew JP, Burke K, Ernzerhof M. Generalized gradient approximation made simple. Phys Rev Lett 1996;77(18):3865.

[73] Hammer B, Hansen LB, Norskov JK. Improved adsorption energetics within density-functional theory using revised Perdew–Burke–Ernzerhof functionals. Phys Rev B 1999;59:7413.

[74] Perdew JP, Ruzsinszky A, Csonka GI, Vydrov OA, Scuseria GE, Constantin LA, et al. Restoring the density-gradient expansion for exchange in solids and surfaces. Phys Rev Lett 2008;100(13):136406.

[75] Hamann DR. Optimized norm-conserving Vanderbilt pseudopotentials. Phys Rev B 2013;88(8):085117.

[76] Troullier N, Martins JL. Efficient pseudopotentials for plane-wave calculations. Phys Rev B 1991;43(3):1993–2006.

[77] Schlipf M, Gygi F. Optimization algorithm for the generation of oncv pseudopotentials. Comput Phys Comm 2015;196:36–44.

[78] Xu Q, Sharma A, Comer B, Huang H, Chow E, Medford A, et al. Supporting information for SPARC: simulation package for Ab-initio real-space calculations, Mendeley Data V1. http://dx.doi.org/10.17632/mcgvnnmf78.1.

[79] Momma K, Izumi F. VESTA 3 for three-dimensional visualization of crystal, volumetric and morphology data. J Appl Crystallogr 2011;44(6):1272–6.

[80] Khoo KH, Chan T-L, Kim M, Chelikowsky JR. Ab initio molecular dynamics simulations of molten $Al_{1-x}Si_x$ alloys. Phys Rev B 2011;84:214203.

[81] Singh R, Kroll P. Structural, electronic, and magnetic properties of 13-, 55-, and 147-atom clusters of fe, co, and ni: A spin-polarized density functional study. Phys Rev B 2008;78:245404.

[82] Austin B, Bhimji W, Butler T, Deslippe J. 2014 NERSC workload analysis. http://portal.nersc.gov/project/mpccc/baustin/NERSC_2014_Workload_Analysis_v1.1.pdf.

[83] Vernon LJ. IC application performance team analysis, as part of IC knights special project. Tech. rep., LANL; 2015.

[84] Suryanarayana P. On spectral quadrature for linear-scaling density functional theory. Chem Phys Lett 2013;584:182–7.

[85] Banerjee AS, Suryanarayana P. Cyclic density functional theory: A route to the first principles simulation of bending in nanostructures. J Mech Phys Solids 2016;96:605–31.

[86] Ghosh S, Banerjee AS, Suryanarayana P. Symmetry-adapted real-space density functional theory for cylindrical geometries: Application to large group-IV nanotubes. Phys Rev B 2019;100(12):125143.

[87] Kumar S, Suryanarayana P. Bending moduli for forty-four select atomic monolayers from first principles. Nanotechnology 2020;31(43):43LT01.

[88] Sharma A, Suryanarayana P. Real-space density functional theory adapted to cyclic and helical symmetry: Application to torsional deformation of carbon nanotubes. Phys Rev B 2021 103(3):035101.

[89] Suryanarayana P, Bhattacharya K, Ortiz M. Coarse-graining Kohn–Sham density functional theory. J Mech Phys Solids 2013;61(1):38–60.

[90] Xu Q, Sharma A, Suryanarayana P. M-SPARC: Matlab-simulation package for Ab-initio real-space calculations. SoftwareX 2020;11:100423.