npj | computational materials

Article

Published in partnership with the Shanghai Institute of Ceramics of the Chinese Academy of Sciences

https://doi.org/10.1038/s41524-026-02133-7

GEMDAT: a Python toolkit for site-resolved diffusion analysis in solid-state molecular dynamics

Check for updates

Anastasia K. Lavrinenko[1], Theodosios Famprikis[2], Victor Landgraf[1], Jouke R. Heringa[1,3], Stef Smeets[4], Victor Azizi[4], Simone Ciarella[4], Marnix Wagemaker[1] & Alexandros Vasileiadis[1]

Molecular dynamics (MD) simulations have become essential for understanding diffusion mechanisms in solid-state materials such as ionic conductors, fuel cells, and gas sensors, yet most existing studies and software tools extract only standard metrics, leaving much of the information contained in the trajectories unused. Here we introduce GEMDAT, a user-friendly Python toolkit for site-resolved diffusion analysis of MD simulations of solid-state materials (https://github.com/GEMDAT-repos/GEMDAT). Beyond mean-squared displacements, radial distribution functions, and Arrhenius-based activation energies, GEMDAT provides jump rates, attempt frequencies, site-specific activation energies, rotational diffusion, and percolation. Our tool provides access to vibrational amplitudes, site geometries, and site occupancies—quantities that are also directly comparable to experimental diffraction data. Migration sites can be defined manually or identified automatically from the trajectory. A built-in caching approach, together with rapid visualization capabilities, makes the workflow fast and interactive. We demonstrate GEMDAT on a series of case studies involving crystalline Li- and Na-ion conductors, plastic crystals, amorphous structures, and surface configurations, showing how the code extracts atomic-level structural features and connects them to macroscopic transport properties, thereby guiding the optimization and development of solid-state materials.

Mass transport plays a fundamental role in determining the performance and functionality of solid-state materials and devices used in batteries, semiconductors, fuel cells, and gas sensors[1–4]. Species diffusion through a material directly affects charge transport, reactivity, reaction rates, and structural stability. In solid-state materials, diffusion typically happens through thermally activated jumps between atomic sites[1,5,6]. Mobile species occupy these low-energy stable or metastable sites until they gain sufficient energy to jump over the barrier to neighboring sites. Mapping these sites and tracking individual jumps provides critical insights into how features, such as structural disorder, element doping, introducing vacancies, size of bottlenecks, and local symmetry, govern mass transport[7–9].

Molecular dynamics (MD) simulations become essential for exploring mass transport mechanisms across diverse material classes[6,10,11]. Classical MD enables large-scale simulations over long timescales, while ab initio MD (AIMD) provides quantum-mechanical accuracy at a higher computational cost[11,12]. More recently, machine-learned interatomic potentials (MLIPs), such as moment tensor potentials (MTP)[13], Gaussian approximation potentials (GAP)[14], and pretrained models like PET-MAD[15], MACE[16], and CHGNet[17] have bridged these approaches, enabling near-first-principles accuracy with classical-level efficiency across chemically diverse systems.

Despite these advances in simulation capabilities, the tools available for post-analysis of MD trajectories often fail to fully extract the information they contain. General-purpose MD trajectory analysis libraries such as MDAnalysis[18] and MDTraj[19] were designed primarily for biomolecular systems, with data models that organize atoms into chains, residues, and molecular topologies, and analysis modules targeting protein structure, hydrogen bonding, and membrane dynamics. Although MDAnalysis provides mean-squared-displacement calculations, it lacks support for crystallographic site identification, ionic hop detection, and the VASP trajectory formats widely used in solid-state research. MDTraj similarly provides no diffusion analysis, site-specific metrics, or support for crystallographic formats. Tools such as VMD[20] and OVITO[21] offer great visualization and basic

[1]Storage of Electrochemical Energy, Department of Radiation Science and Technology, Faculty of Applied Sciences, Delft University of Technology, Delft, The Netherlands. [2]Inorganic Chemistry Laboratory, Department of Chemistry, University of Oxford, Oxford, United Kingdom. [3]Energy Materials, Department of Radiation Science and Technology, Faculty of Applied Sciences, Delft University of Technology, Delft, The Netherlands. [4]Netherlands eScience Center, Amsterdam, The Netherlands. e-mail: a.vasileiadis@tudelft.nl

statistical metrics, but do not support site-specific diffusion analysis or jump-event detection. On the materials science side, Pymatgen[22] and its pymatgen-analysis-diffusion add-on[23] provide diffusion coefficient extraction, Arrhenius fitting, van Hove correlation functions, and site occupancy analysis within a crystal structure. However, pymatgen-analysis-diffusion does not detect discrete ionic jump events, construct site-to-site transition matrices, or compute jump-based activation energies. Its support is also largely limited to VASP-formatted trajectories, restricting applicability to classical MD simulations from other codes. The site-analysis package[24] provides four site geometries (spherical, Voronoi, polyhedral, and dynamic Voronoi) that can deform dynamically within the framework, but it focuses specifically on site-projection analysis and does not compute transport properties such as diffusion coefficients, conductivities, and activation energies or perform analysis of site geometry, rotational diffusion, and percolation.

Here, we present a generalized molecular dynamics analysis tool (GEMDAT), an open-source Python toolkit designed to fill these gaps by providing comprehensive, site-resolved diffusion analysis for solid-state materials. GEMDAT offers several features designed for flexibility and efficient performance:

- Accepts input directly from widely used MD engines, including VASP[25], LAMMPS[26], and GROMACS[27].
- Seamlessly integrates with widely used Python packages like Pymatgen[22] and Atomic Simulation Environment (ASE)[28].
- Accelerates subsequent analysis through data caching and enables intuitive visualization of results.
- Ensures correctness and reproducibility through automatic tests, supporting future development and community contributions.
- Handles both user-defined atomic sites and automatic density-based mapping of atomic positions directly from MD simulation.
- Integrates with kinisi[29] for robust error estimation.

Beyond standard tracer diffusivities and mean-squared displacements, GEMDAT provides analysis of vibrational amplitude, site occupancies, sizes, and geometry, jump rates and site-specific energy barriers, radial distribution functions, collective motion, autocorrelation functions and rotational diffusion, and jump network and percolation analysis.

We demonstrate the application of GEMDAT on several case studies of solid-state crystalline ionic conductors, plastic crystals, amorphous structures, and surface configurations—materials our research group has extensive experience with. While some GEMDAT functions benefit from a crystalline framework, which allows for applying symmetry operations and improving statistics, the code is equally capable of analyzing non-crystalline materials.

The code documentation and interactive Jupyter notebook tutorials are publicly available for the research community, and we encourage researchers to contribute to further development of GEMDAT and expand its capabilities through shared analysis workflows.

Repository: https://github.com/GEMDAT-repos/GEMDAT.git

Documentation: https://gemdat.readthedocs.io

Tutorials: https://github.com/GEMDAT-repos/gemdat-workshop-2025.git

This article is organized as follows: the "Results" section demonstrates GEMDAT's code structure, its capabilities, and application to different structural families of solid-state materials. These case studies include argyrodites ($Li_{7-x}PS_{6-x}Br_x$), sodium-ion conductors and plastic crystals ($Na_{3-x}W_xSb_{1-x}S_4$, Na_3PS_4), irreducible antifluorites ($Li_{2+x}S_{1-x}N_x$), amorphous materials (Li_xSi), and surface configurations (Li_xCu)—all representing disordered ion conductors. Some GEMDAT functions work particularly well with crystalline frameworks, where symmetry operations can be applied to improve statistics, but the code is equally capable of analyzing non-crystalline materials. The "Methods" section introduces physical principles underlying GEMDAT's calculations and simulation details for the provided case studies.

Results

Code architecture

The architecture and workflow of GEMDAT are shown in Fig. 1. GEMDAT accepts two primary inputs (shown in dashed boxes): MD trajectory files

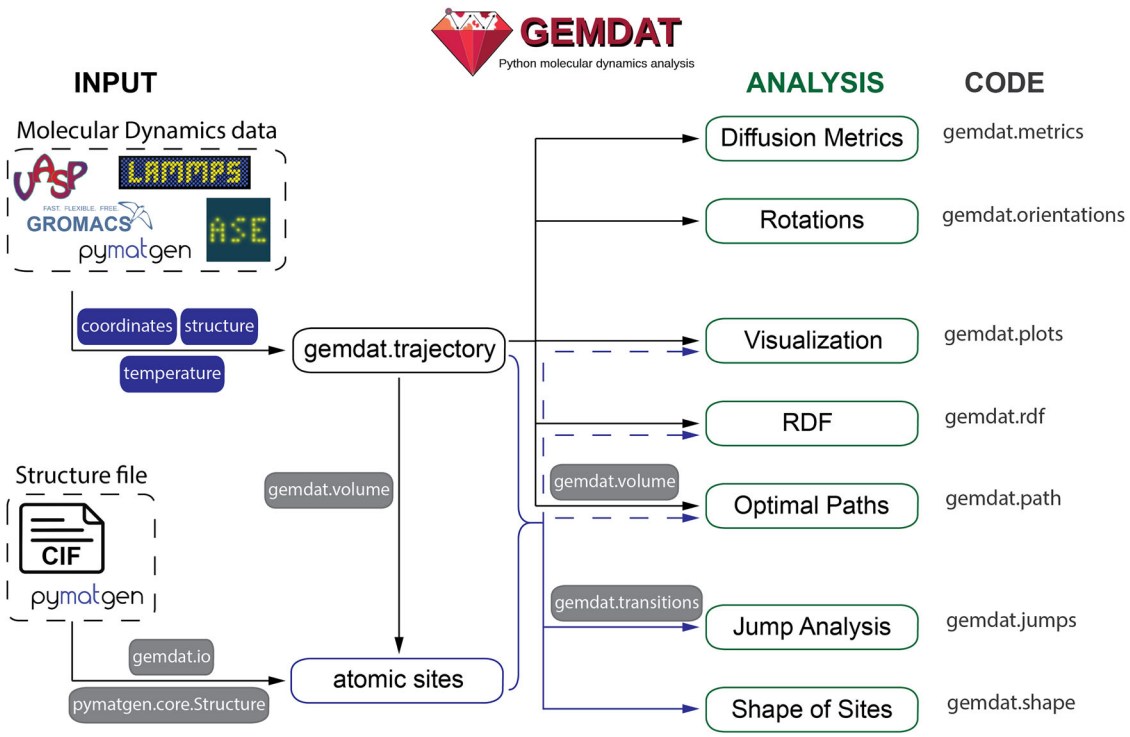


Fig. 1 | Architecture of GEMDAT code. The initial MD trajectories are obtained from atomistic simulation packages (VASP, LAMMPS, GROMACS, Pymatgen or ASE Trajectory object). Atomic sites are either loaded from a Pymatgen Structure object, read from CIF files, or extracted from density volumes of the MD trajectory. Black arrows indicate trajectory-based analysis, solid blue arrows indicate combined trajectory- and site-based analysis, dashed blue arrows indicate trajectory-based analysis where only certain functions require atomic sites. Python classes used in GEMDAT for each analysis type are shown on the right, and the gray boxes above the arrows indicate the intermediate objects required.

from atomistic simulations (VASP, LAMMPS, GROMACS) or from Python packages (ASE, Pymatgen) and a definition of atomic sites available for occupation. These sites can be derived from Pymatgen Structure objects, CIF files, or automatically extracted from high-density regions in the trajectory. Once loaded, GEMDAT creates a `Trajectory` object and saves a pickle cache file to speed up subsequent analyses (see the "Methods" subsection "Trajectory loading and analysis"). The workflow supports two types of analysis—trajectory-based and site-based—indicated by black and blue arrows in Fig. 1, respectively. The trajectory object can be filtered to isolate mobile species and analyze displacement statistics, vibrational amplitudes, and attempt frequencies. Orientational analysis of polyanion rotations provides insights into rotational dynamics (see the "Methods" subsection "Orientations and rotational analysis"). Radial distribution functions characterize bonding and coordination environment, with the option to perform site-resolved analysis when atomic sites are available (see the "Methods" subsection "Radial distribution function (RDF)"). The `Volume` class constructs 3D density grids from trajectory, allowing users to visualize diffusion pathways and calculate activation energies (see the "Methods" subsection "Volume-based analysis"). When crystallographic sites are defined (see the "Methods" subsection "Site definition"), GEMDAT identifies ion transitions between them, extracts discrete jump events, and computes jump diffusivity, jump conductivity, and site-specific (or jump) activation energies (see the "Methods" subsection "Jump analysis"). These jump activation energies and the jump network serve as input for graph percolation analysis, which reveals active sites connected to the percolating network and determines percolation activation energies (see the "Methods" subsection "From jump library to percolation analysis"). Shape analysis characterizes the site geometry and the distribution of mobile ions around atomic sites (see the "Methods" subsection "Shape analysis"). All analysis modules support visualization through both Plotly (`backend='plotly'`) for interactive exploration and Matplotlib (`backend='matplotlib'`) for static figures (Fig. 2).

Representative timings for trajectory reading, mean squared displacement, and radial distribution function calculations are compared with MDAnalysis, Pymatgen, and ASE across trajectories of varying size—from a short trajectories in Table S1, to a long VASP trajectory (104 atoms, 120,000 frames) and a large GROMACS trajectory (17,836 atoms, 10,001 frames) in Table S2—providing an estimate of the computational cost of each analysis type and of the feasibility of applying GEMDAT to the longer and larger trajectories now accessible with MLIPs. We note that GEMDAT's automatic pickle caching significantly reduces trajectory-loading times on subsequent runs, which is particularly beneficial during iterative analysis.

It is worth noting that for trajectories obtained under variable-cell conditions, several modules of GEMDAT should be applied with caution. The site-assignment procedure (see the "Methods" subsection "Site definition") relies on a static reference structure (or on density maxima extracted from a fixed-cell trajectory) and therefore assumes that the host sublattice does not undergo significant translational drift or anisotropic distortion. The voxel-based probability density (see the "Methods" subsection "Volume-based analysis") and the derived free-energy landscape are constructed on a discretised grid of fixed dimensions, and would need to be reformulated in fractional coordinates, or recomputed on a cell-averaged reference, to remain meaningful when the lattice vectors fluctuate. Likewise, the ionic conductivity computed via the Nernst–Einstein relation (Eq. (4)) requires a particle density that becomes time-dependent under variable-cell conditions and must therefore be averaged consistently over the trajectory.

Role of disorder in argyrodites

The simulation data and analysis presented in this section are reused from ref. argyrodites[8]. Here, we focus on how GEMDAT's analysis modules were applied to extract site-resolved transport properties of argyrodites and connect them to site disorder, configurational entropy, and local environments.

$Li_{7-x}PS_{6-x}Br_x$ argyrodites crystallize in the cubic $F\bar{4}3m$ space group (Fig. 3a). PS_4^{3-} tetrahedra occupy Wyckoff 4b positions, while halide ions at 4a sites and sulfide ions (not bonded to P^{5+}) at 4d positions form the anion sublattice[30]. Due to similar ionic radii, Br^- and S^{2-} ions can exchange positions and occupy both 4a and 4d sites[31]. Li^+ transport occurs through a network of tetrahedral interstitial sites, including T5 (48h), T2 (48h), and T4 (16e) positions, which form cage-like substructures around the 4d sites (Fig. 3b). Ion diffusion proceeds via three distinct jump mechanisms (Fig. 3c): short-range doublet (T5–T5) and intracage (T5–T2) jumps within individual cages, and long-range intercage (T2–T2 and T5–T4–T5) jumps between neighboring cages that are typically the rate-limiting steps for macroscopic transport[8,32].

Using GEMDAT, we extracted the lithium probability density distribution from AIMD simulations to visualize the Li^+ migration pathways across structures with varying configurational entropy and bromine content (Fig. 4). In ordered Li_6PS_5Br with S^{2-} located only at 4d sites, lithium density concentrated within isolated cages with no intercage diffusion (Fig. 4a). However, both increased configurational entropy and higher bromine content enhanced intercage migration and flattened the energy landscape for diffusion.

GEMDAT-calculated site-specific activation energies for T5–T5 doublet and T5–T4 intercage jumps showed that higher bromine occupancy at 4d sites decreases activation energy for intercage jumps but increases it for doublet jumps, causing a switch in the rate-limiting step (Fig. 5a–c). To further quantify structural changes between studied materials, we used GEMDAT shape analysis, which measured average distances between 4a/4d anion sites and surrounding lithium positions, representing cage radii (Fig. 5d–f). We found that higher bromine occupancy at 4d sites led to expansion of the 4d-centered cages, shortening of intercage distances, and an increase in doublet jump distances. Maximum conductivity occurred when 4a and 4d cages achieved similar radii (Fig. 5e), creating a uniform electrostatic force distribution.

We used GEMDAT jump analysis to characterize each jump event by its neighboring 4a and 4d anions (Fig. 6b) and to construct a library of environment-specific activation energies (Fig. 6a). Mixed S^{2-}/Br^- occupation creates low-energy pathways for intercage diffusion while raising activation energy for doublet intracage jumps. This analysis demonstrated that disorder-enhanced conductivity requires a statistically uniform distribution of favorable environments across all rate-limiting jump types, preventing high-barrier environments from becoming bottlenecks that limit percolation pathways.

Rotational analysis of plastic crystals

Na_3PS_4 has been shown to transition to a plastically crystalline (i.e., "rotor-") phase upon heating beyond 500 °C[33,34]. The rotor-phase polymorph (γ-Na_3PS_4) exhibits distinct librational and rotational mobility of the PS_4^{3-} molecular anion, along with a distinct stepwise increase in sodium ionic conductivity[33,35]. MD simulations can be used to capture and describe rotational mobility, which is of particular interest in understanding the interplay between rotational and translational dynamics in such rotor-phase electrolytes[36].

Figure 7a shows the probability density of sulfur from AIMD simulations, demonstrating the increased librational amplitude of PS_4^{3-} in γ-Na_3PS_4 compared to the low-temperature β-Na_3PS_4 polymorph (both simulated at 900 K). In β-Na_3PS_4, the sulfur atoms vibrate in well-defined, solid-like, approximately spherical energy minima ("sites"). In γ-Na_3PS_4, the probability density of sulfur is significantly more diffuse, indicating significant mobility and positional disorder, consistent with the exceedingly high displacement parameters emerging from fitting of diffraction data[33].

Another convenient way to capture the orientational behavior of molecular units is by tracking bond orientations. GEMDAT creates a trajectory of vectors (here, P–S bond vectors) and can conveniently produce plots of the orientational probability function: a function on the surface of a sphere representing the probability of the bonds pointing in a particular direction. Figure 7b presents the orientational probability functions for β- and γ-Na_3PS_4 as functions of polar angles in a rectilinear projection. For β-Na_3PS_4, four sharp maxima are observed along the <111> directions of the

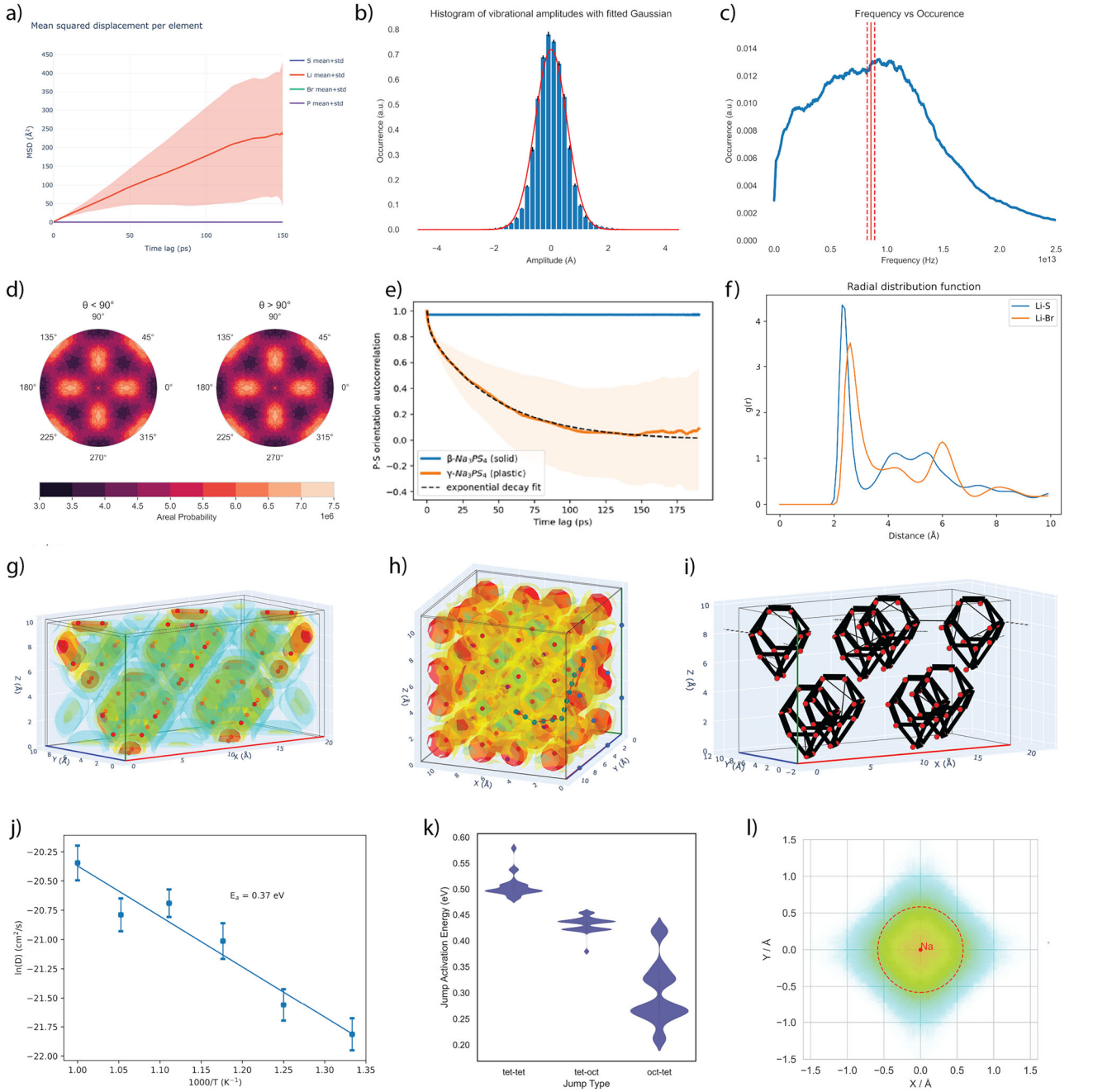


Fig. 2 | Representative visual outputs from GEMDAT analysis of AIMD trajectories. a Mean squared displacement. **b** Histogram of vibrational amplitudes. **c** Vibrational frequency spectrum with attempt frequency. **d** Probability density distribution of polyanion orientations in spherical coordinates. **e** Bond orientation self-correlation function. **f** Radial distribution functions. **g** Three-dimensional probability density distribution of diffusing atoms overlaid with crystallographic site positions. **h** Three-dimensional probability density distribution highlighting the percolating diffusion pathway. **i** Jump network visualization showing jumps that occurred between crystallographic sites. **j** Arrhenius plot of tracer diffusivity at multiple temperatures for activation energy determination. **k** Site-specific jump activation energies as a function of local coordination environment. **l** Shape analysis revealing the geometry of a crystallographic site.

cubic cell, indicative of a single, well-defined orientation for all tetrahedra, and consistent with the reported crystallographic model of β-Na_3PS_4 ($I\bar{4}3m$)[34]. For γ-Na_3PS_4, eight diffuse maxima are observed. The diffuse nature of the maxima is indicative of the large orientational mobility of the P–S bonds (and by extension the PS_4^{3-} anions). The eight maxima represent two sets of four, each set representing a tetrahedral orientation, consistent with the two equally represented orientations for PS_4^{3-} anions in the crystallographic model of γ-Na_3PS_4 (*Fddd*)[33].

The data presented in Fig. 7 is consistent with the rotor-phase description of γ-Na_3PS_4 but does not prove per se that PS_4^{3-} are dynamically reorienting (i.e., rotationally diffusing). To capture reorientations, one can determine the time-dependent self-correlation of bond orientations via the orientational self-correlation function, $\zeta(t)$, which quantifies the degree of reorientation as a function of time. GEMDAT conveniently computes ζ from bond vector trajectories, as shown in Fig. 8.

For β-Na_3PS_4, ζ quickly decays and plateaus at a value close to 1 (~0.97). This is indicative of low-amplitude libration of the P–S bonds about their equilibrium orientation, as generally expected in any solid at finite temperature. Orientational correlations are conserved in time: the orientation of a P–S bond at any given time does not significantly differ from its previous orientation. In contrast, for γ-Na_3PS_4, ζ decays to 0 over time, indicating loss of orientational correlation, consistent with large-amplitude

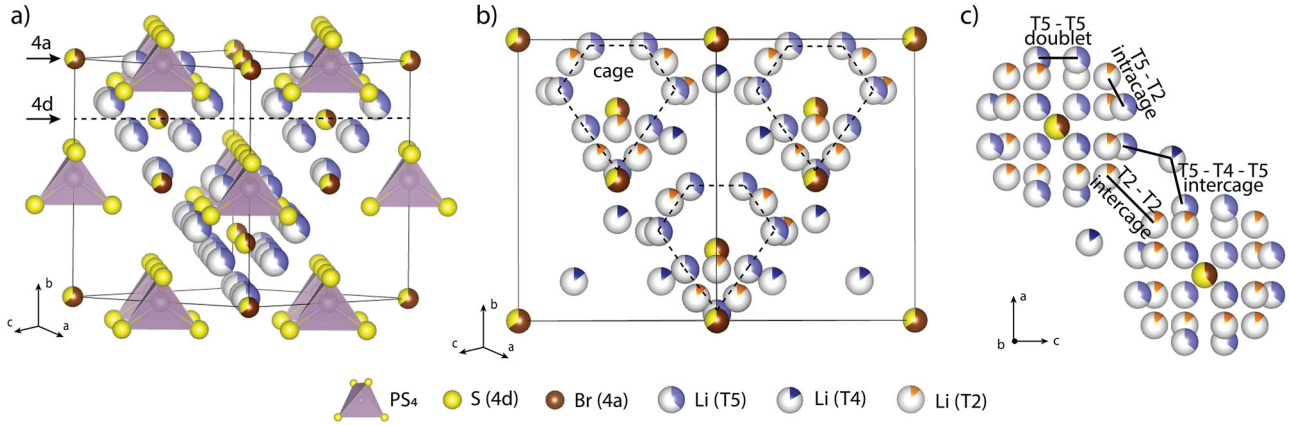


Fig. 3 | Structure of argyrodite. Br^- and S^{2-} ions are shown in brown and yellow, respectively. PS_4^{3-} tetrahedra are shown in violet. Lithium crystallographic sites T5 (Wyckoff 48h), T4 (Wyckoff 16e), and T2 (Wyckoff 48h) are shown in light blue, dark blue, and orange, respectively. **a** Crystal structure of Li_6PS_5Br argyrodite, where Br^- and S^{2-} ions (not bonded to P^{5+}) share Wyckoff 4a and Wyckoff 4d positions. **b** Lithium crystallographic sites form a cage-like structure around the Wyckoff 4d site. **c** Types of Li-ion jumps: intracage (T5–T5 doublet and T5–T2) and intercage (T5–T4–T5 and T2–T2).

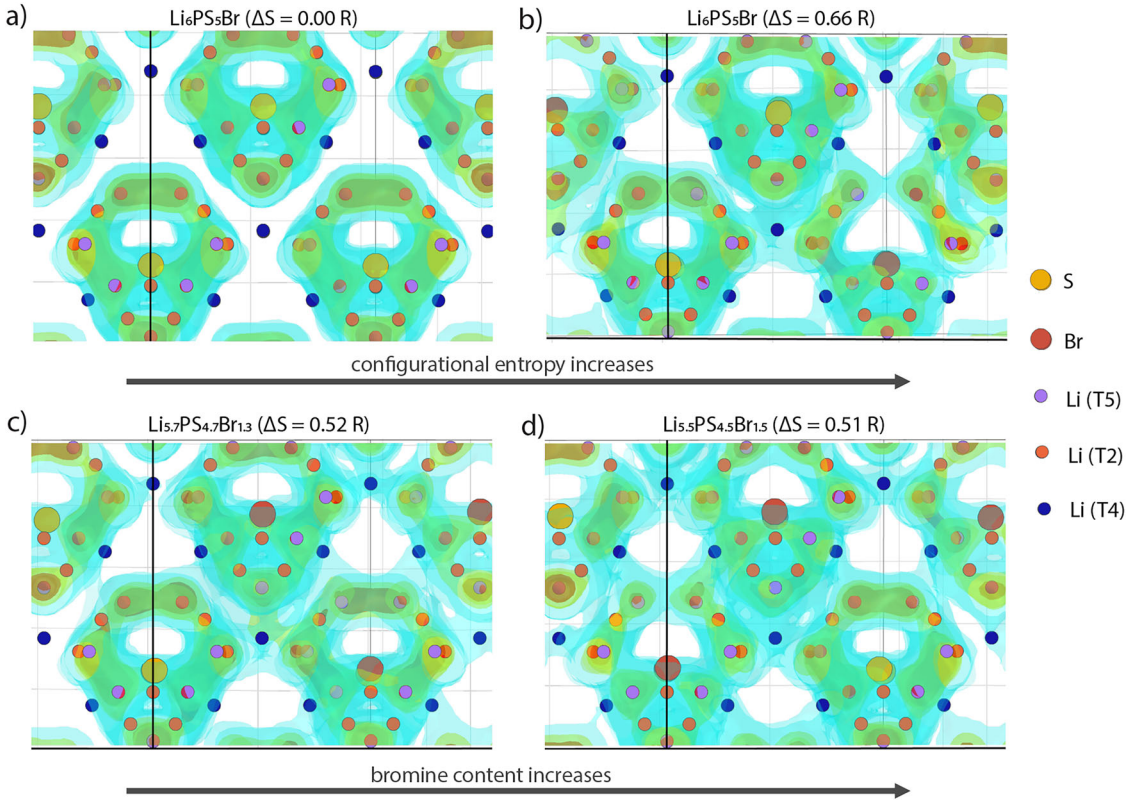


Fig. 4 | Probability density of Li^+ obtained from AIMD simulations at 650 K extracted by GEMDAT. Br^- and S^{2-} ions are shown in red and yellow, respectively. Lithium crystallographic sites T5 (Wyckoff 48h), T4 (Wyckoff 16e), and T2 (Wyckoff 48h) are shown in violet, blue, and orange, respectively. The Li probability density is represented using a discrete color scheme, increasing from cyan (low density) through yellow (intermediate) to red (high density). The top two panels **a** and **b** illustrate lithium transport as a function of increasing configurational entropy (ΔS), while the bottom two panels **c** and **d**, highlight differences in transport behavior with increasing bromine content. Adapted from ref. 8 (© 2024 Royal Society of Chemistry), licensed under CC BY 3.0. Changes were made.

molecular reorientations over time (i.e., rotor-phase behavior). The decay of ζ can be fitted to an exponential decay to quantify a characteristic time for PS_4^{3-} reorientations, which is of the order of 50 ps at 900 K, consistent with the value reported by Sau and Ikeshoji from classical MD simulations[35].

Site shape and geometry analysis of sodium-ion conductors

Partial aliovalent W^{6+} substitution in $Na_{2.875}W_{0.125}Sb_{0.875}S_4$ stabilizes the cubic β-Na_3PS_4-type structure (Fig. 9a) while introducing substantial sodium disorder through vacancy creation[37]. This structural complexity, characterized by anharmonic ionic distributions and partial site occupancies, poses significant challenges for conventional crystallographic refinement.

We used GEMDAT shape analysis to examine the sodium-ion density distribution from a 50 ps AIMD simulation at 500 K and identify the optimal crystallographic model for the $Na_{2.875}W_{0.125}Sb_{0.875}S_4$ structure. Analysis using the conventional Wyckoff 6b site assignment (0, 0, 0.5)[34,37] (Fig. 9a) captured a sodium occupancy of 0.95, which agrees well with the nominal stoichiometry. However, this assignment required a large site

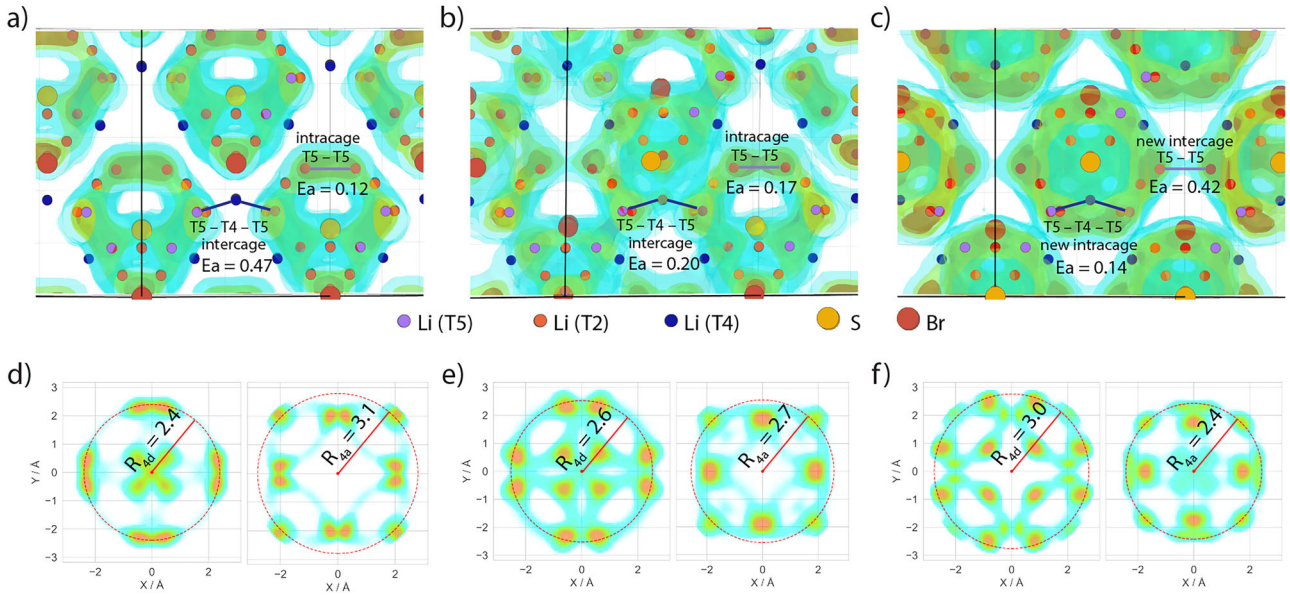


Fig. 5 | Connection of Li^+ transport and radii of Li-cages centered at 4d and 4a sites. Br^- and S^{2-} ions are shown in red and yellow, respectively. Lithium crystallographic sites T5 (Wyckoff 48h), T4 (Wyckoff 16e), and T2 (Wyckoff 48h) are shown in violet, blue, and orange, respectively. The Li probability density is represented using a discrete color scheme, increasing from cyan (low density) through yellow (intermediate) to red (high density). The top panels **a**–**c** show the probability density of Li^+ obtained from AIMD and activation energies for T5–T5 doublet and T5–T4–T5 intercage jumps. The bottom panels **d**–**f** show the radii of Li-cages centered at 4d and 4a sites. The structures presented are: **a** and **d** Li_6PS_5Br, $\Delta S = 0.00R$ (all S^{2-} are placed at 4d sites); **b** and **e** $Li_{5.5}PS_{4.5}Br_{1.5}$, $\Delta S = 0.55R$ (S^{2-} ions are equally distributed between 4d and 4a sites); **c** and **f** Li_6PS_5Br, $\Delta S = 0.00R$ (all S^{2-} are placed at 4a sites).

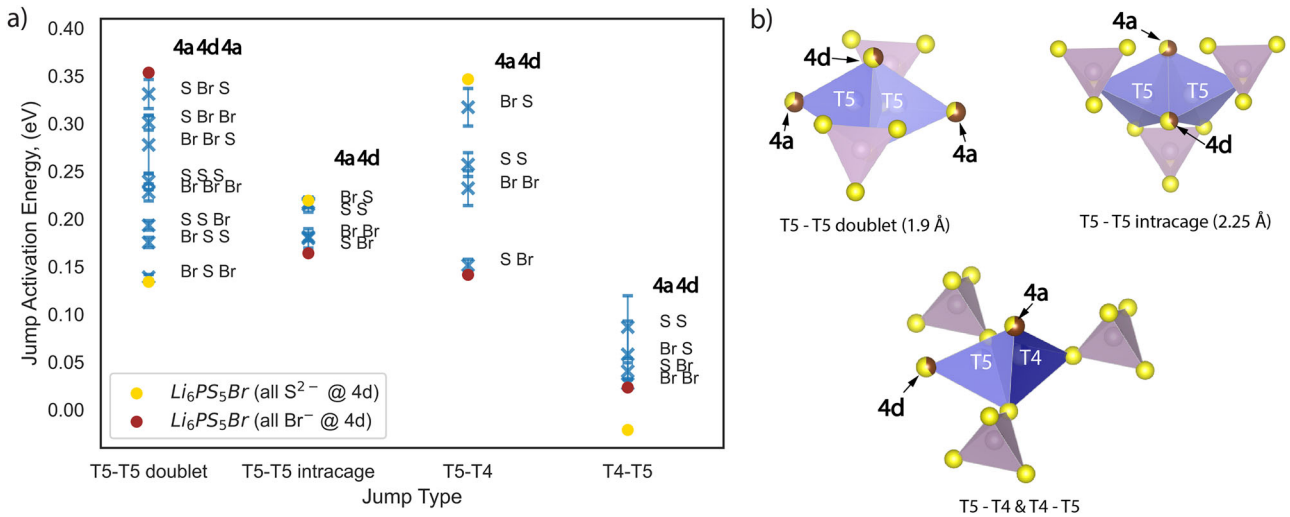


Fig. 6 | Impact of local S^{2-}/Br^- environments on Li^+ jump activation energies in Li_6PS_5Br. **a** Average jump activation energies (in blue) calculated from AIMD simulations at 650 K for all possible local anionic environments characterizing T5–T4 intercage and T5–T5 doublet jumps. Yellow and red solid dot markers indicate the jump activation energies of Li_6PS_5Br, $\Delta S = 0.00R$ (all S^{2-} are placed at 4d sites) and Li_6PS_5Br, $\Delta S = 0.00R$ (all S^{2-} are placed at 4a sites), respectively. Jump environments are labeled by anions at the 4a and 4d positions neighboring the lithium pathway. **b** Coordination of lithium T5 and T4 tetrahedral sites for each jump type. The 4a and 4d sites, which are used to label jump environments, are shown as mixed-occupancy positions (half-yellow/half-red spheres) and indicated by arrows. T5 tetrahedra (light blue) are coordinated by two S^{2-} ions (yellow), corner-shared with PS_4^{3-} tetrahedra (violet), plus two anions from the 4a and 4d sites. T4 tetrahedra (dark blue) are coordinated by three S^{2-} ions, corner-shared with PS_4^{3-} tetrahedra plus one anion from the 4a site. Adapted from ref. 8 (© 2024 Royal Society of Chemistry), licensed under CC BY 3.0. Changes were made.

radius of 1.5 Å, which exceeds the default threshold of twice the vibrational amplitude ($A = 0.51$ Å for Na^+). GEMDAT shape analysis further revealed anisotropic sodium distribution, showing a two-dimensional diamond-like shape (Fig. 9c). This finding is consistent with previous diffraction and computational studies of Na_3PS_4-type structures[37,38]. To enable direct comparison with experimental crystallographic data and extract quantitative anisotropic atomic displacement parameters (ADPs), we fitted the three-dimensional sodium coordinate distributions to Gaussian models using the scikit-learn[39] GaussianMixture module, a probabilistic framework that assumes data points are generated from a mixture of Gaussian distributions with unknown parameters (Fig. 10). For the Wyckoff 6b position, we fitted a one-component three-dimensional Gaussian mixture to extract the covariance matrix, which directly relates to the anisotropic displacement parameter tensor U. The covariance matrix was converted to anisotropic displacement parameter using the following equation[40]: $B = 8\pi^2 U$. This analysis yielded an isotropic displacement parameter $B_{iso} \approx 10.6$ Å^2 with

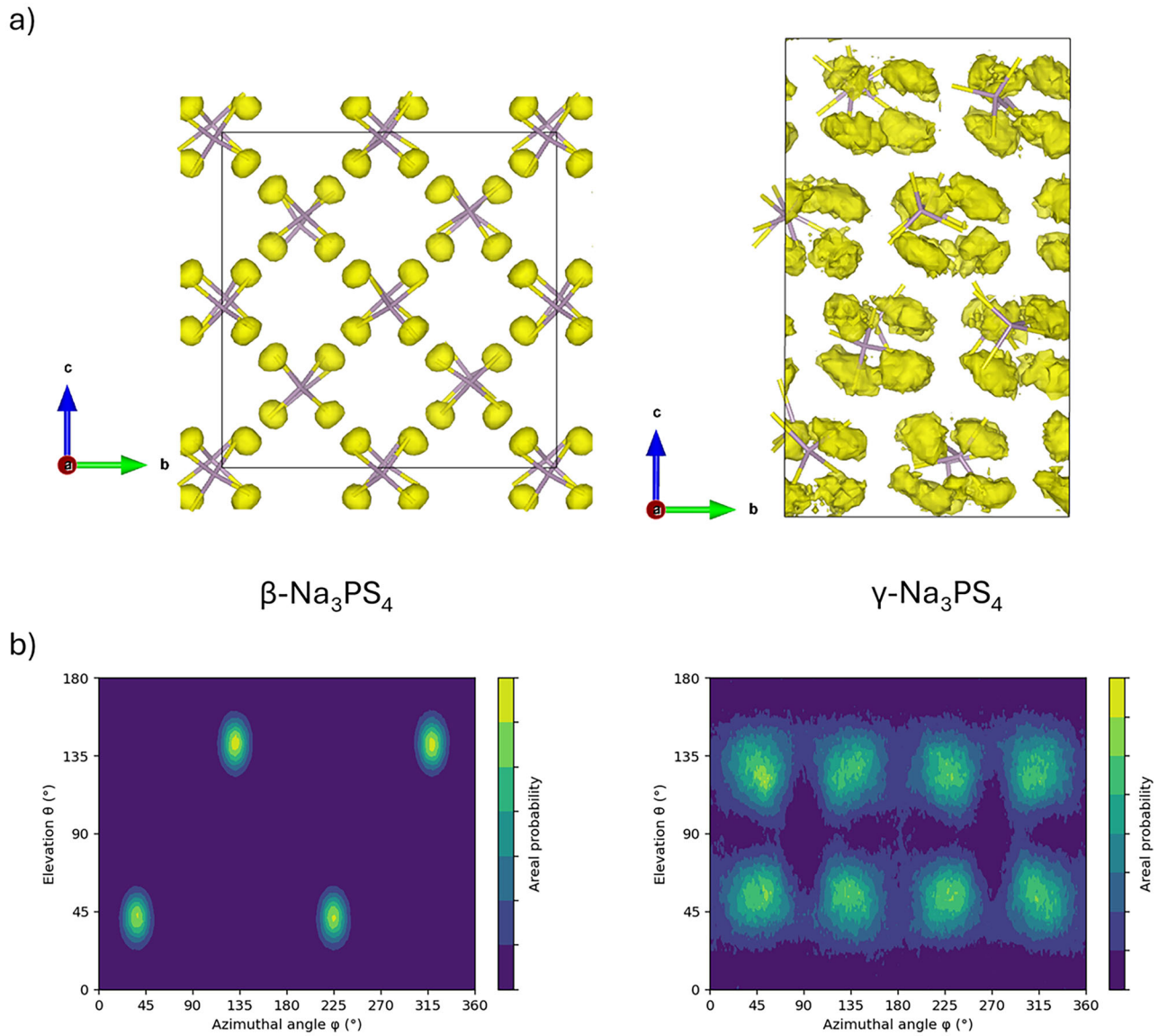


Fig. 7 | Comparison of sulfur probability density in β- and γ-Na_3PS_4 from AIMD simulations at 900 K. a Volumetric probability density in the respective simulation boxes in yellow. PS_4^{3-} tetrahedra shown as stick tetrahedra, Na^+ omitted for clarity. **b** Orientational probability density of P–S bonds in spherical coordinates.

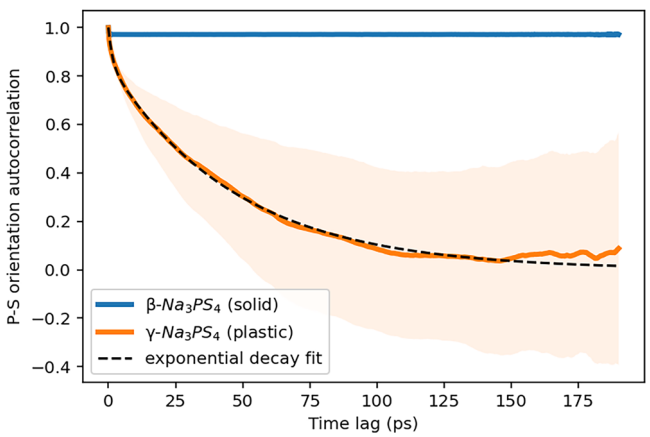


Fig. 8 | Orientational self-correlation function, $\zeta(t)$ (Eq. (8)), of P–S bonds from AIMD simulations of β- and γ-Na_3PS_4 at 900 K. The shaded region represents the standard deviation of the orientational self-correlation function across all individual P–S bonds in the supercell.

strong anisotropy, as reflected in the eigenvalues of the B tensor, showing significantly larger displacements in the *ab*-plane than in the *c*-direction (Fig. 10c). Such large displacement parameters, while capturing the observed probability density distribution, are relatively big for conventional thermal motion at 500 K, indicating either local disorder or insufficient crystallographic site definition.

As an alternative, we evaluated the Wyckoff 24f site model (Fig. 9b), in which sodium is represented by four discrete, partially occupied positions at the diamond vertices. We fitted a four-component 3D Gaussian mixture model (Fig. 10b) to extract optimal 24f site parameters from the AIMD-derived probability density distribution. The fitting results in the components with equal statistical weights, which reflect the symmetry-equivalent nature, and an isotropic displacement parameter $B_{iso} \approx 6.0$ Å^2, significantly smaller than that for the Wyckoff 6b model. The mean positions extracted from the fit—representing the centers of the four Gaussian components—directly correspond to fractional coordinates (0.406, 0, 0.5) for the Wyckoff 24f sodium sites (Fig. 9b), matching the crystallographic model reported in previous studies[34,37,38] and validating the GEMDAT analysis methodology.

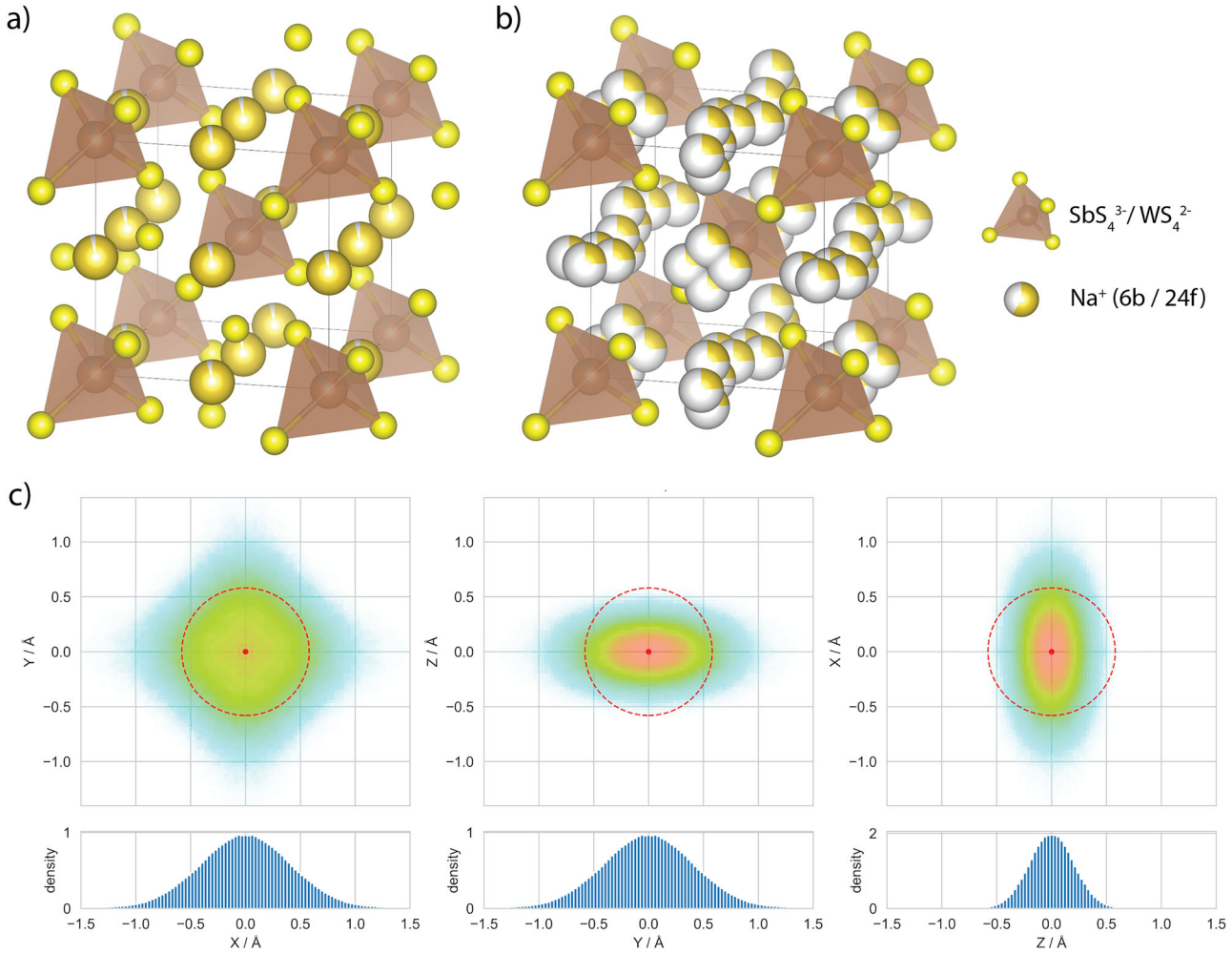


Fig. 9 | $Na_{2.875}W_{0.125}Sb_{0.875}S_4$ crystallographic site shape analysis from AIMD at 500 K. a Crystallographic model of $Na_{2.875}W_{0.125}Sb_{0.875}S_4$ with sodium (in mixed white/yellow) placed at Wyckoff 6b. **b** Crystallographic model of $Na_{2.875}W_{0.125}Sb_{0.875}S_4$ with sodium (in mixed white/yellow) placed at Wyckoff 24f. **c** Projections of sodium position during AIMD simulation with center in Wyckoff 6b site. The red dashed circle represents the average displacement of sodium ions from the center position (red dot). The Na probability density is represented using a discrete color scheme, increasing from cyan (low density) through yellow (intermediate) to red (high density).

Percolation analysis in disordered irreducible antifluorites

The simulation data and analysis presented in this section are reused from ref. 9. Here, we focus on demonstrating the GEMDAT-based workflow for extracting jump-specific activation energies and performing percolation analysis to examine Li^+ ion mobility in ordered and disordered configurations of irreducible antifluorite electrolyte Li_9S_3N.

In the anion-disordered lithium-rich antifluorite structure, Li^+ primarily occupies tetrahedral sites, with octahedral sites only partially occupied[9,41]. Using GEMDAT jump analysis, three jump pathways were identified from the AIMD trajectories: tetrahedral-to-octahedral (tet-oct) through triangular three-anion bottlenecks, octahedral-to-tetrahedral (oct-tet) through the same triangular bottlenecks, and tetrahedral-to-tetrahedral (tet-tet) jumps through linear two-anion bottlenecks (Fig. 11).

To capture all possible local jump environments, we generated multiple independent disordered configurations with a random S/N distribution while maintaining a 3:1 stoichiometry. We used GEMDAT to extract jump-specific activation energies and characterize each jump by its initial site coordination, destination site coordination, and bottleneck composition. The disordered S/N arrangement enabled 91 distinct jump types (Fig. 11f–h), compared to only 6 in the ordered phase (Fig. 11b–d). The resulting distribution of jump-specific activation energies (Fig. 12a) revealed that certain local environments created by disorder enabled low-energy pathways with activation barriers below 0.35 eV, while the minimum barrier in ordered Li_9S_3N exceeded 0.43 eV. Low-energy jumps generally occurred in the nitrogen-rich bottlenecks, which can be attributed to the smaller ionic radius and higher charge density of N^{3-}, creating wider migration channels.

Building on these GEMDAT-extracted activation energies, we developed a percolation model[9] operating on 5 × 5 × 5 supercells (1500 Li sites)—system sizes that exceed AIMD accessibility yet can be analyzed using only jump activation energies from smaller AIMD cells. The model represents the lithium sublattice as a graph where Li sites are nodes and jumps are edges. Two sites are connected only when both forward ($i \rightarrow j$) and backward ($j \rightarrow i$) jumps have activation energies below the threshold E_{cutoff}. The network is considered percolating if it spans the entire supercell.

At an energy cutoff of $E_{cutoff} = 0.4$ eV, ordered Li_9S_3N showed no percolating pathways. Lack of tet-oct and tet-tet jumps below this barrier created "inactive" sites that are not connected to the percolation network and cannot participate in long-range diffusion. At the same energy cutoff, disordered Li_9S_3N exhibited percolating networks with a significantly larger amount of lithium sites connected to it (green sites in Fig. 12b). The percolation-energy diagrams provided further quantitative insights (Fig. 12c), plotting the fraction of Li sites connected to percolation networks as a function of the energy threshold. The minimum percolation threshold was 0.34 eV for disordered compared to 0.43 eV for ordered Li_9S_3N, reflecting the availability of low-energy jumps created by nitrogen-rich environments. These simulated percolation energies aligned closely with experimental conductivity activation energies (0.39 eV disordered, 0.47 eV ordered)[9].

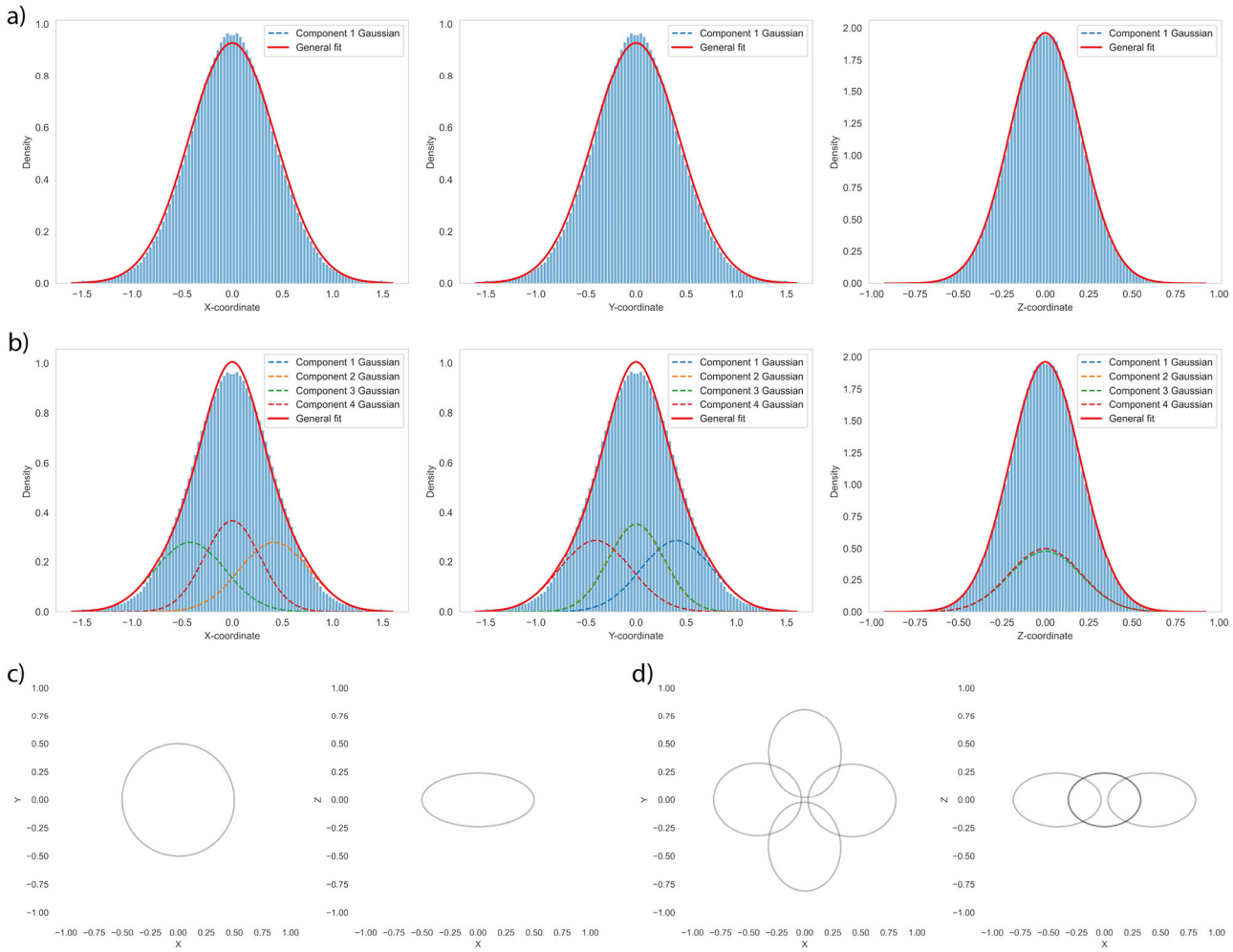


Fig. 10 | Projections of sodium coordinates distribution (in blue) with 3D Gaussian fit. General fit is shown in a solid red line, while Gaussian components are shown in dashed lines. **a** One-component 3D Gaussian to represent a crystallographic model with Wyckoff 6b position. **b** Four-component 3D Gaussian to represent a crystallographic model with Wyckoff 24f position. **c** Projected coordinate densities with anisotropy derived from a one-component Gaussian mixture. **d** Projected coordinate densities with anisotropy derived from a four-component Gaussian mixture.

Lithium diffusion in amorphous silicon

The diffusion of lithium in bulk amorphous silicon was investigated to provide an atomistic rationale for the rate-limiting kinetics governing the electrochemical lithiation of Si anodes. Although Si offers a theoretical capacity approximately ten times that of graphite, its practical performance is constrained by Li transport through the disordered host network[42,43]. Prior computational work has established that the heterogeneous void distribution in Si yields a broad spectrum of migration barriers[42], yet the relationship between local structural topology and macroscopic diffusivity remains incompletely understood, motivating the present analysis.

The diffusive behavior of the Li and Si sublattices in the amorphous $Li_{36}Si_{64}$ supercell was characterized by analyzing the MSD extracted from the AIMD trajectories using GEMDAT (Fig. 13). The tracer diffusion coefficient was determined as 1.22×10^{-10} m^2 s^{-1}. This value falls within the range reported for amorphous Li_xSi at low-to-intermediate lithiation levels from both AIMD[42] and experimental tracer diffusion measurements[44], and is several orders of magnitude larger than diffusivities measured in crystalline Si at room temperature[45], underscoring the kinetic advantage of the amorphous structure for Li transport.

Analysis of Li trajectories using GEMDAT identified preferred Li-occupancy sites within the amorphous Si matrix (Fig. 13a, b). Li atoms were found to vibrate locally about their equilibrium positions with characteristic displacements of 1.0–1.4 Å (Fig. 13c), giving rise to a number of distinct sites. The proximity of these distances suggests that adjacent preferred sites are separated by flat energy barriers, permitting frequent short-range hops while the atoms remain trapped within the same structural void of the Si network over extended timescales. This is in qualitative agreement with the picture established by previous studies[42,44], who showed that Li preferentially occupies large ring interstices in the disordered Si network and that local structural topology governs the accessibility of neighboring sites.

The mean activation energy per Li jump event, obtained from the GEMDAT transition rate analysis, is 0.197 ± 0.013 eV. This value falls within the reported range of microscopic barriers in bulk Si (~0.1–2.4 eV)[42] and reflects the relatively low-energy hops between nearby sites observed in the AIMD trajectories.

At first glance, this is much lower than the macroscopic activation energy of ~1.4 eV measured in isotopic tracer experiments[44]. However, this difference does not indicate a discrepancy but rather arises from the fundamentally different transport regimes probed by the two approaches. AIMD captures intrinsic short-range hopping between locally connected sites, which involves low activation barriers. In contrast, macroscopic diffusion depends on the formation of continuous percolation pathways across the network. This long-range transport is limited by deep trapping events associated with undercoordinated or strained Si environments, which dominate transport over experimental timescales and thus determine the effective activation energy[44].

Consistent with this picture, the simulations reveal pronounced local mobility without the emergence of long-range diffusion. Frequent short-

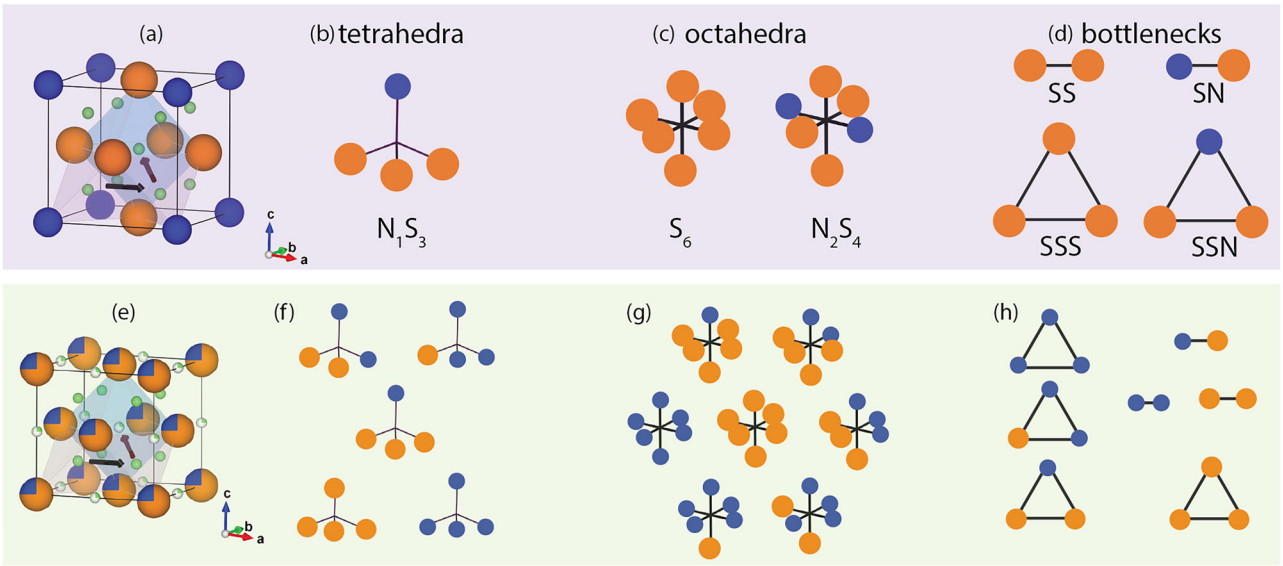


Fig. 11 | Local coordination environments and migration bottlenecks in ordered and disordered Li_9S_3N antifluorite structures. Li, S, and N atoms are shown as green, orange, and blue spheres, respectively. **a** Unit cell of ordered Li_9S_3N with lithium occupying tetrahedral sites (pink polyhedra) and octahedral sites (blue polyhedra). Black arrows indicate representative tet-tet and tet-oct jumps. **b** A tetrahedral coordination environment, which is present in ordered Li_9S_3N. **c** Octahedral coordination environments, which are present in ordered Li_9S_3N. **d** Linear and triangular jump bottlenecks, which are possible in ordered Li_9S_3N. **e** Unit cell of disordered Li_9S_3N. **f** Tetrahedral coordination environments accessible in disordered Li_9S_3N. **g** Octahedral coordination environments in disordered Li_9S_3N. **h** Linear and triangular jump bottlenecks, which are possible in disordered Li_9S_3N. Reprinted from ref. 9 (© 2025 American Chemical Society), licensed under CC BY 4.0.

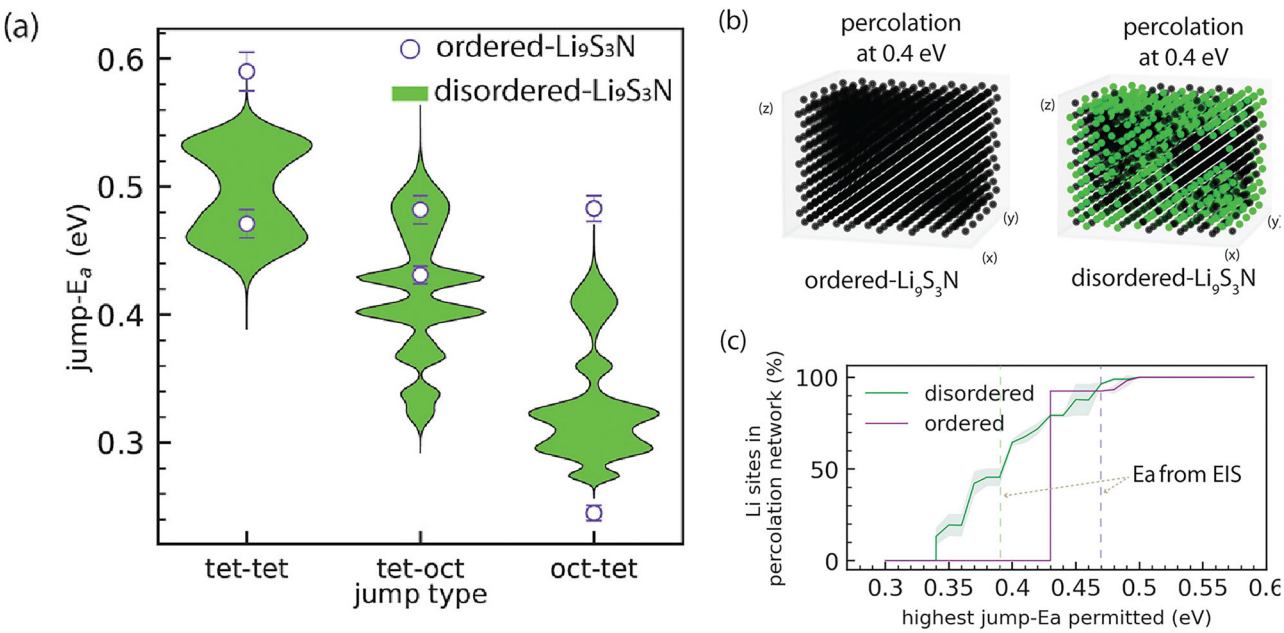


Fig. 12 | Percolation analysis in Li_9S_3N. a Distribution of jump activation energies for all observable jump types in ordered and disordered Li_9S_3N, extracted from AIMD simulations at 900 K. **b** Three-dimensional visualization of percolating networks in $5 \times 5 \times 5$ supercells for ordered Li_9S_3N and disordered Li_9S_3N. Sites highlighted in green are part of percolating networks using an energy threshold of 0.4 eV. **c** Percolation-energy diagram showing the fraction of Li sites that are connected to percolating networks for ordered and disordered Li_9S_3N. Shading is the standard deviation over 50 disordered $5 \times 5 \times 5$ supercells. Vertical dashed lines indicate the experimental activation energies. Reprinted from ref. 9 (© 2025 American Chemical Society), licensed under CC BY 4.0.

range hopping occurs between nearby sites (Fig. 13b), showing short-range connectivity (Fig. 13c), yet this motion does not translate into macroscopic transport, as evidenced by the very low MSD (Fig. 13d) and the limited network connectivity shown in Fig. 13a.

Lithium diffusion on copper surface

The mobility of lithium on copper surfaces was investigated to elucidate the mechanistic origin of the reversible Li uptake observed experimentally in Cu current collectors during electrochemical plating and stripping[46]. Although Li incorporation into Cu has been reported previously[47,48], the absence of lattice parameter changes in operando X-ray diffraction measurements[46] suggested that transport proceeds preferentially via grain boundary pathways rather than bulk lattice diffusion, further supported by the quantification of grain boundary versus lattice diffusion coefficients in Cu[49]. Understanding Li surface mobility is therefore essential for rationalizing current collector degradation and

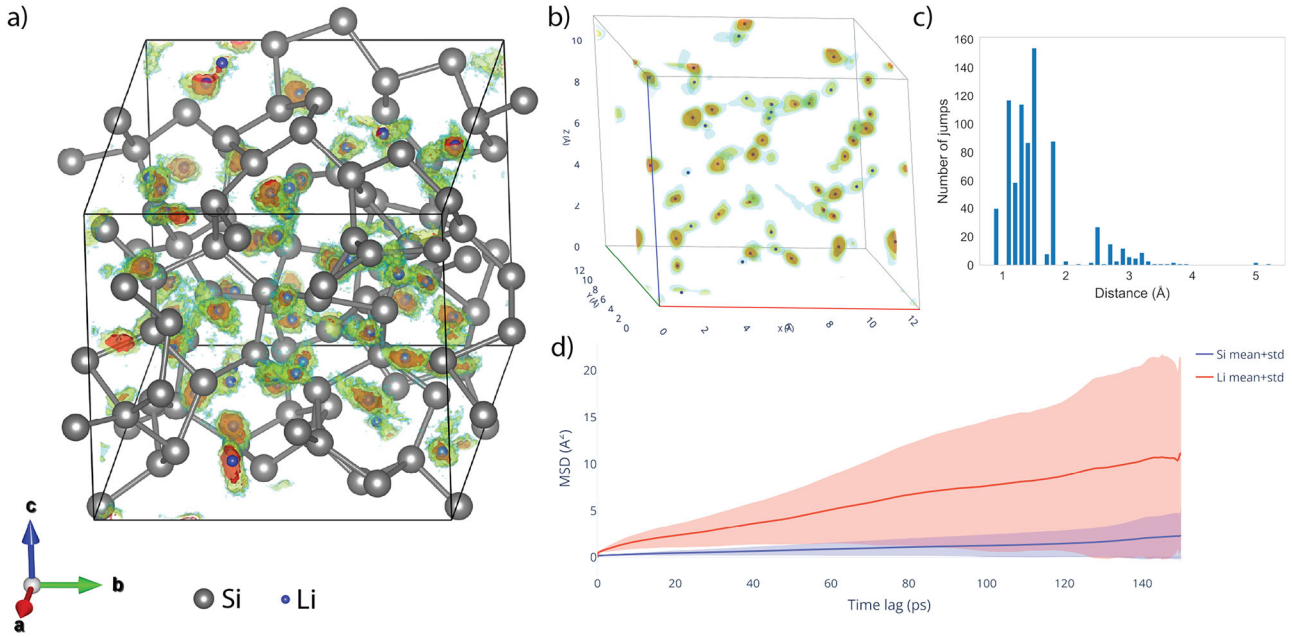


Fig. 13 | Li mobility inside the amorphous Si framework. Li positions are shown in blue, and Si atoms are shown in gray. The Li probability density is represented using a discrete color scheme, increasing from cyan (low density) through yellow (intermediate) to red (high density). **a** Li probability density resulting from a 150 ps AIMD simulation at 600 K. **b** Li density indicating the Li preferred positions extracted from AIMD using GEMDAT. **c** Histogram of Li jump distances. **d** MSD of Li (in red) and Si (in blue) atoms during AIMD simulation.

informing strategies to suppress Li trapping in next-generation Li-metal batteries[50,51].

The MD trajectory data from ref. 46. were reanalyzed using GEMDAT for four Li atoms co-adsorbed on the Cu(111) surface. In the original study, two distinct adsorption sites were identified from DFT calculations: the energetically favorable p2 positions, where Li maximizes its distance from the three nearest surface Cu atoms, and the less stable p1 positions, lying ~100 meV higher in energy[46]. Reanalysis of the transition statistics extracted from the AIMD trajectories confirmed that Li atoms reside predominantly at p2 sites, with only transient events through the p1 positions (Fig. 14). No measurable energy barrier was found for migration between adjacent p2 sites, indicating an essentially flat potential energy surface governing lateral Li transport along the Cu(111) surface. The tracer diffusion coefficient extracted from the MSD is 4.49×10^{-8} m^2 s^{-1} (Fig. 14d), confirming that Li diffusion on this surface is rapid. Taken together, these results support the interpretation that grain boundary surfaces in the Cu current collector provide facile Li transport pathways, offering a microscopic rationalization for the reversible Li uptake observed experimentally during electrochemical plating and stripping[46,49].

Discussion

The research examples presented throughout this work demonstrate how GEMDAT enables the extraction of mechanistic atomic-scale insights from MD simulations and connects them to macroscopic transport properties. Its general analysis framework can be applied across diverse solid-state material families. Beyond common metrics such as mean-squared displacement and radial distribution functions, GEMDAT offers analytical tools that were previously inaccessible or available only through custom scripts tailored to a specific material.

GEMDAT's trajectory analysis uses atomic sites to discretize ion motion into individual jump events and calculates diffusional properties, including site occupancy and geometry, jump rates, jump diffusivity, and site-specific activation energies. Analysis of the jump network and mechanism reveals how local coordination environments and bottleneck compositions control migration barriers. In addition to jump analysis, GEMDAT also investigates rotational dynamics, site geometry, and percolation pathways. Rotational analysis provides insights into the orientational disorder of polyanions and rotational diffusion. Analysis of atomic site geometry characterizes anisotropic ion distributions that define the shape of sites and allows the extraction of atomic displacement parameters that can be directly compared to experimental diffraction data. Percolation analysis identifies which sites are connected to the percolation network and examines how structural disorder creates low-energy pathways that enable macroscopic conductivity.

The demonstrated applicability of GEMDAT across structurally diverse and highly disordered lithium and sodium conductors establishes it as a versatile tool for analyzing MD in solid-state materials. Its modular and customizable architecture ensures that new analysis methods can be easily incorporated as the field advances. By providing general, reproducible workflows for extracting transport properties at the atomic level from MD trajectories, GEMDAT contributes to the understanding of the fundamental principles governing ionic transport in solid-state materials.

Methods

Trajectory loading and analysis

The `Trajectory` class in `gemdat.trajectory` loads MD simulation data from multiple packages using Pymatgen's trajectory readers as backend. The `Trajectory.from_vasprun()` method reads VASP XML files, `Trajectory.from_lammps()` reads LAMMPS trajectory files requiring structure and temperature inputs, `Trajectory.from_gromacs()` reads GROMACS trajectory and topology files requiring temperature input and supporting extraction of time-series energy data from GROMACS EDR file, and `Trajectory.from_ase_trajectory()` reads ASE trajectory file requiring temperature input. Upon first loading, GEMDAT automatically saves a pickle cache file, significantly reducing loading time for subsequent analysis. The `Trajectory` object stores a time series of atomic positions for all species, along with simulation cell lattice parameters, species information, timestep data, and simulation temperature. The underlying data is stored as NumPy arrays, enabling efficient memory access and standard Python indexing and slicing. Periodic boundary conditions are automatically handled when computing displacements and distances.

Filtering the trajectory to isolate mobile ion species is a critical preprocessing step for extracting diffusion metrics. The

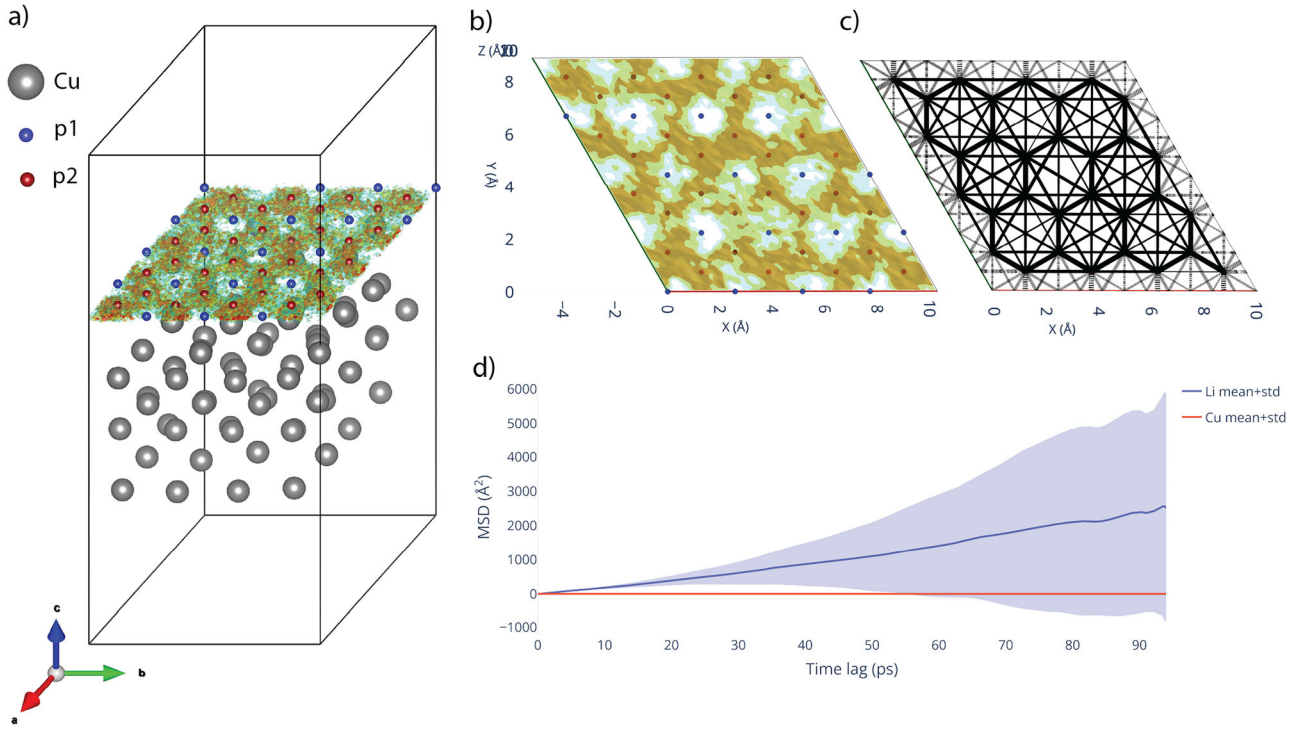


Fig. 14 | Li mobility at the Cu surface. Lithium positions p1 and p2 are shown in blue and red, respectively, while Cu atoms are shown in gray. The Li probability density is represented using a discrete color scheme, increasing from cyan (low density) through yellow (intermediate) to red (high density). **a** Li probability density of 4 diffusing Li atoms on the (111) copper surface resulting from a 94 ps AIMD simulation at 600 K. **b** Top view of the Li density indicating the Li p2 positions and the energetically less favorable p1 positions. **c** Detected transitions, represented as black lines, resulted from sampling the surface with the p2 surface positions. The thickness of the lines scales with the number of transitions. **d** MSD of atoms during AIMD simulation.

`Trajectory.filter(list_of_species)` method creates a new trajectory with only the specified species. Before identifying discrete jumps, GEMDAT analyzes continuous trajectory properties. The `TrajectoryMetrics` class in `gemdat.metrics` computes fundamental diffusion properties from atomic displacements without requiring the definition of atomic sites. The `TrajectoryMetricsStd` class provides block analysis[52,53] for error estimation. The `Trajectory` class also provides built-in plotting methods for preliminary analysis.

The `Trajectory` class computes *mean squared displacement* (MSD), which quantifies the average squared deviation of particle positions from their initial locations as a function of time[54] (Fig. 2a):

$$MSD(t) = \frac{1}{N}\sum_{i}^{N}(r_i(t) - r_i(0))^2 \tag{1}$$

where N represents the total number of diffusing particles and $r_i(t)$ denotes the position vector of the particle i at time t. This approach computes the displacement of each particle relative to its initial position at $t = 0$ providing a direct measure of particle spreading over time.

To improve statistical accuracy and reduce the influence of initial configuration effects, GEMDAT averages the direct method over multiple overlapping time windows, where all possible time origins along the trajectory are used:

$$MSD(t) = \frac{1}{N}\sum_{i}^{N}\frac{1}{M-t}\sum_{t_0=0}^{M-t-1}(r_i(t+t_0) - r_i(t_0))^2 \tag{2}$$

where M represents the total number of time steps in the simulation. Despite its computational intensity, this method is implemented using fast Fourier transform (FFT) algorithms, which makes it computationally tractable while preserving improved statistics of the overlapping sliding-window sampling[55]. The MSD plot (Fig. 2a) shows the average mean-squared displacement per atomic species in the supercell, with error bars representing the standard deviation across all individual ions of that species.

The `Trajectory` class additionally provides an interface to the *Bayesian diffusion analysis* implemented in the kinisi package[29,56]. kinisi likewise computes the MSD through overlapping sliding-window sampling but additionally estimates the variance of the observed MSD by rescaling the variance of the observed squared displacements by the number of numerically independent subtrajectories in the simulation. The kinisi `DiffusionAnalyzer` then models the resulting MSD as a multivariate normal distribution with an analytically derived covariance matrix and samples the diffusion coefficient via Bayesian regression. This approach provides an efficient estimate of the diffusion coefficient with an accurate quantification of its statistical uncertainty. A more detailed description of the methodology can be found in ref. 29.

The `TrajectoryMetrics` and `TrajectoryMetricsStd` classes compute several diffusion properties discussed below.

The *tracer diffusion coefficient* D^* quantifies the mobility of individual particles without accounting for interparticle correlations that contribute to collective transport.

From the linear regime of the MSD, the tracer diffusivity is obtained through the Einstein relation[54]:

$$D^* = \frac{MSD(t)}{2dt} \tag{3}$$

where d is the dimensionality of diffusion and MSD(t) is the total MSD at time step t.

While the GEMDAT approach of extracting the tracer diffusion coefficient from the Einstein relation, with optional block-averaging for uncertainty estimation, is well-established and computationally efficient, it

treats MSD values across different time windows as independent observations, which leads to systematically underestimated error bars[29]. To overcome this limitation, GEMDAT additionally provides the kinisi `DiffusionAnalyzer` interface introduced above. The native `TrajectoryMetrics` route is therefore well-suited to rapid inspection and screening, whereas the kinisi backend is preferable when robust uncertainty quantification on D^* is required.

The *ionic conductivity* can be estimated from tracer diffusivity using the Nernst–Einstein relation, assuming negligible inter-ionic correlations[54]:

$$\sigma = \frac{ne^2z^2}{k_B T} D^* \tag{4}$$

where n is the diffusing particle density, e the elementary electron charge, z the ionic charge, k_B Boltzmann's constant, and T the temperature in Kelvin. This approximation provides an upper bound for ionic conductivity, as correlations typically reduce the actual conductivity below this Nernst-Einstein limit.

The *amplitude of vibrations* can be extracted by monitoring the absolute displacement and segmenting it into intervals $[t_a, t_b]$ over which the discrete-time derivative maintains a constant sign. The amplitude A for that segment is[7]

$$A = r_i(t_b) - r_i(t_a) \tag{5}$$

where t_a and t_b represent time points where the displacement derivative maintains a consistent sign, indicating oscillatory rather than diffusive motion. The resulting amplitude distribution is often well described by a Gaussian function, from which the mean vibrational amplitude and its anisotropy can be extracted (Fig. 2b).

The *attempt frequency* ν^* represents the frequency at which the system attempts to make a transition[57]. The vibrational spectrum is computed through derivatives of the absolute displacement (Δr_i) per atom (i) at every time step (t)[7]:

$$\Delta r_i(t) = r_i(t) - r_i(t-1) \tag{6}$$

which is transformed to a frequency spectrum via a Fourier transformation. The frequency spectrum of each (diffusing) atom is then combined to obtain the vibrational spectrum. From the frequency spectrum, the attempt frequency can be obtained by treating each near-harmonic vibration of a diffusing ion as an "attempt" and taking the average peak frequency as the attempt frequency (Fig. 2c). Note that the presented approach determines a frequency spectrum from the vibrations of individual atoms. The obtained spectrum thus differs from a phonon spectrum, which depends on the collective vibrations of the atoms.

When simulations at multiple temperatures are available, the *activation energy* E_A can be extracted from the temperature dependence of the tracer diffusion coefficient through the Arrhenius relation[58]:

$$D(T) = D^0 e^{\left(\frac{-E_A}{k_B T}\right)} \tag{7}$$

where D^0 is the pre-exponential factor, k_B Boltzmann's constant, and T the temperature in Kelvin. This analysis is implemented in GEMDAT through the `ArrheniusFit` class in `gemdat.metrics`, which can be initiated directly from a list of trajectories at different temperatures via the `ArrheniusFit.from_trajectories()` constructor. The activation energy and pre-exponential factor are obtained from a weighted linear regression of $\ln D^*$ against $\frac{1}{T}$, with weights derived from the per-temperature uncertainties on D^* (Fig. 2j). It should be noted, however, that linearisation of the Arrhenius equation is known to introduce a statistical bias in the parameter estimates[59]. Users requiring rigorous uncertainty quantification on E_A are therefore advised to perform non-linear optimization externally.

Orientations and rotational analysis

For simulations including molecular units (e.g., tetrahedral polyanions like SO_4^{2-}, PO_4^{3-}), GEMDAT provides functionality to analyze their orientation. Given two atom types, the `gemdat.orientations` module generates a trajectory of vectors (b) from the positions (r) trajectory, representing the length and orientation of bonds of interest at every time step. The bond length distribution is readily accessible. The orientational probability function (Fig. 2d) can also be produced to examine the distribution of bond orientations throughout the trajectory.

Finally, the orientational self-correlation function, $\zeta(t)$, can be readily produced (Fig. 2e) to quantify the amplitude and rate of orientational dynamics[35]:

$$\zeta(t) = \frac{1}{\mathrm{B}} \sum_i^{\mathrm{B}} \frac{1}{M-t} \sum_{t_0=0}^{M-t-1} b_i(t_0) \cdot b_i(t_0 + t) \tag{8}$$

where B is the number of bonds and M is the total number of time steps in the simulation. The orientational self-correlation function is averaged over all bonds and time windows Δt, and is computed via an FFT, similar to the MSD (Eq. (2)).

Radial distribution function (RDF)

The `gemdat.rdf` module computes RDF $g(r)$ to characterize local coordination environments, solvation shell structure, and structural evolution (Fig. 2f). It quantifies the probability of finding a particle of type j at distance r from a particle of type i, normalized by the bulk density[60]:

$$g(r) = \frac{1}{\frac{4}{3}\pi((r+\Delta r)^3 - r^3)p} \frac{1}{M} \sum_t^M \sum_i \sum_{j \neq i} \delta\left(r - |r_i(t) - r_j(t)|\right) \tag{9}$$

The RDF is determined by calculating the distance between all species pairs $|r_i(t) - r_j(t)|$, for each time step t and binning them into a histogram. M is the total number of time steps. The histogram is then normalized, where normalization is the particle density p multiplied by the volume of the spherical shell $\frac{4}{3}\pi((r+\Delta r)^3 - r^3)$ with resolution Δr.

The `radial_distribution_between_species(specie_1, specie_2)` function provides RDFs between two species or lists of species, requiring only trajectory data. The `radial_distribution(transitions, floating_specie)` function computes site-resolved RDFs, calculating pair distributions of mobile ions around each atom type while separating mobile species by the site type they occupy. This returns a dictionary of `RDFData` objects, one for each site-pair type. This function requires a `Transitions` object created by combining the trajectory and the crystallographic sites structure (see Methods: Site Definition).

Volume-based analysis

The `Volume` class in `gemdat.volume` converts trajectories into discretized 3D grids that represent the time-averaged probability density of finding given species at each position. This probability density distribution provides a continuous view of diffusion pathways and enables automatic extraction of atomic sites. A volume object can be constructed using `volume = trajectory.to_volume(resolution = 0.1)`, where the resolution parameter controls the voxel size in Å. The probability density $P(\xi)$ is obtained directly from the MD trajectory by accumulating, for each voxel ξ, the number of times the diffusing species is found within that voxel across all trajectory frames. The resulting occupancy histogram is normalized by the total number of counts, yielding a spatially resolved probability density of the mobile ion. Note that voxels are not necessarily cubic, as their shape is determined by the lattice parameters of the simulation cell.

The `Volume` class stores the 3D density array and lattice information and provides multiple analysis capabilities. Density data is interactively visualized as isosurfaces using `volume.plot()`, revealing regions of high

occupancy. When atomic sites are provided, they are overlaid on the density map to correlate ion diffusion with site positions (Fig. 2g). Density data can also be exported in CHGCAR format for visualization in VESTA[61] or other crystallographic software.

To automatically extract site positions from local density maxima, the `Volume.to_structure()` method uses a watershed algorithm[62], with the `background_level` parameter controlling peak sensitivity—higher values reduce spurious peaks in noisy regions. The resulting peak centers define a set of site positions that can be converted into a structure object for subsequent analyses.

The `Volume` class also calculates activation energies along diffusion pathways. The time-averaged probability density $P(\xi)$ is first converted to a free energy landscape[63]:

$$A(\xi) = -k_B T \ln P(\xi) \tag{10}$$

where ξ is a voxel index, T is the temperature, and k_B is Boltzmann's constant. Voxels that are frequently visited by the mobile species correspond to low-energy positions, whereas rarely visited voxels along connecting channels correspond to transition regions. The activation energy associated with a given migration pathway is then extracted as the difference between the maximum of $A(\xi)$ along the path and the value of $A(\xi)$ at the initial site. Optimal paths between sites are found using graph-based path-finding algorithms (Fig. 2h). The 3D grid is converted into a graph, in which voxels are nodes and edges connect neighboring voxels, with weights proportional to the free energy. Dijkstra's[64] or Bellman–Ford[65,66] algorithm identifies the minimum energy path between specified sites. Multiple alternative pathways can be found by iteratively removing previously identified routes and re-running the search.

Site definition

Site-based diffusion analysis requires defining the positions between which diffusing ions jump. Sites can be determined directly from a Pymatgen Structure object, loaded from a CIF file, or extracted from trajectory density data (see the "Methods" subsection "Volume-based analysis").

The `Transitions` class in `gemdat.transitions` assigns each mobile ion to its nearest site at each time step by defining a radius around each atomic position. If no site lies within the defined radius, the ion is assigned to index -1, indicating an interstitial or unassigned position. By default, the doubled vibrational amplitude is used as the site radius. The algorithm automatically prevents double-counting when sites overlap by reducing the radius to half the inter-site distance. The `Transitions` object stores two key matrices: the occupancy matrix denoting which site ion i occupies at time t, and the transition event matrix recording all site-to-site changes. These matrices enable comprehensive occupancy statistics and jump analysis through built-in methods.

Site occupancy o_j, representing the fraction of occupied sites of type j, is calculated as[7]

$$o_j = \frac{1}{n_{\text{sites}^j}} \sum_{i}^{n_{\text{sites}^j}} \sum_{k}^{n_{\text{atoms}}} \frac{t_{\text{occupied}}}{t_{\text{total}}} \tag{11}$$

where n_{sites^j} represents the number of sites of type j, n_{atoms} is the number of diffusion atoms, t_{occupied} is the time site i is occupied by an atom k, and t_{total} is the total simulation time.

Atom locations, representing the fraction of atoms occupying sites of type j, are calculated as[7]

$$atom_locations_j = \frac{1}{n_{atoms}} \sum_{i}^{n_{\text{sites}^j}} \sum_{k}^{n_{\text{atoms}}} \frac{t_{occupied}}{t_{total}} \tag{12}$$

These quantities provide complementary information about site preferences and the distribution of diffusing species among available crystallographic positions.

Jump analysis

The `Jumps` class in `gemdat.jumps` takes a `Transitions` object (see the "Methods" subsection "Site definition") as an input and extracts discrete jump events from the transitions data (Fig. 2i). There are two key parameters that help to distinguish these jump events from thermal vibrations. The *distance threshold* defines the inner sphere radius of the site core region. A jump is only registered when an atom leaves the inner sphere of its initial site and enters the inner sphere of the destination site. The `site_inner_fraction` parameter in `Transitions` specifies what fraction of the site radius determines this inner sphere. The *time threshold* specifies the minimum residence time required at the destination site to confirm that a transition has occurred and is controlled by the `minimal_residence` parameter in `Jumps`.

For each jump, the following information is stored: initial and final site indices, jump time, and ion index. The `Jumps` class computes several jump-based diffusion properties discussed below.

Jump rates Γ_i provide direct insight into the rate-limiting process of diffusion, and can be calculated as[7]

$$\Gamma_i = \frac{J_i}{Nt} \tag{13}$$

where J_i represents the number of observed jumps of type i, N is the number of diffusing atoms, and t is the simulation time. A jump type is defined as a unique site pair (start → end) between which transitions occur.

From a random-walk theory, the *jump diffusion coefficient* D_J is obtained through the relation[7,67,68]:

$$D_J = \sum_i \frac{\Gamma_i a_i^2}{2d} \tag{14}$$

where a_i is the jump distance of the jump type i, and d is the number of diffusion dimensions, and the sum extends over all jump types contributing to macroscopic diffusion.

In comparison to tracer diffusivity (D^*), jump diffusivity typically overestimates the diffusion coefficient because it counts back-and-forth jumps that do not contribute to macroscopic displacement. The tracer single-particle correlation factor f quantifies how efficiently individual jumps contribute to macroscopic diffusion[67,69]:

$$f = \frac{D^*}{D_J} \tag{15}$$

Values $f < 1$ reflect directional back-correlation of consecutive jumps, i.e., the ion tends to partially retrace its most recent jump, so that not every jump contributes effectively to net displacement. It does not quantify collective or cross-ion effects.

When multiple temperature simulations are available, activation energies can be extracted through the Arrhenius relation (Eq. (7)). Alternatively, single-temperature estimates of *jump activation energy* employ transition state theory, serving as a quantitative measure of jump probability, where low activation energies correspond to high transition probabilities and vice versa[7,68]:

$$\Delta E_i^{\text{A}} = -k_B T \ln\left(\frac{\Gamma_i^{eff}}{\nu^*}\right) \tag{16}$$

where k_B is Boltzmann's constant, T is the temperature in Kelvin, ν^* Is the attempt frequency, and Γ_i^{eff} represents the effective jump rate corrected for

initial site occupancy o_j[7]:

$$\Gamma_i^{eff} = \frac{J_i}{tNo_j} \tag{17}$$

where t is the total simulated time, and N the number of diffusing atoms. Equation (16) provides a more accurate estimate of the jump rate, since only atoms occupying the initial site type can participate in a given transition.

Collective jumps, where multiple atoms move in a coordinated fashion, can be identified by analyzing spatiotemporal correlations in transition events. Jumps that occur within one vibrational period ($\frac{1}{\nu^*}$) and within specified spatial thresholds are classified as collective, providing insights into correlated diffusion mechanisms[7].

Jump network connectivity can also be analyzed using `jumps.to_graph()`, returning a NetworkX graph where nodes are sites and edges are jump pathways weighted by activation energy.

The `Jumps` class also provides built-in plotting methods.

From Jump Library to percolation analysis

Jump network and jump activation energies are the basis for percolation analysis, which provides insights into long-range diffusion and the identification of transport pathways in complex materials. The methodology[9] requires an initial characterization of the local structural environments surrounding each atomic site, followed by the construction of an environment-specific jump library extracted from MD simulations using GEMDAT (Fig. 2k). Characterization of the environment is based on the nearest-neighbor shell, although second- and higher-shell neighbors can also be included where relevant.

The model represents the lithium sublattice as a directed graph where diffusion atom sites are nodes and possible jumps between sites are directed edges, indicating the jump direction. The disordered local environment around sites is generated randomly and is highly dependent on the structure under investigation. Two sites are considered connected only when both forward ($i \rightarrow j$) and backward ($j \rightarrow i$) jumps exhibit randomly sampled environment-specific activation energies from their respective uncertainty distributions below a given energy threshold. A path is percolating if it spans the supercell by wrapping around the cell under periodic boundary conditions, i.e., it connects opposite sides of the supercell in at least one lattice direction, and the end point is also a valid start point. For each energy cutoff, the percolation analysis computes the fraction of sites participating in system-spanning clusters. The analysis requires statistical averaging over multiple random structure configurations to ensure convergence.

Due to high specificity to the investigating structure, percolation analysis is not included as a general GEMDAT module. However, our tutorials provide examples of the general analysis supported by GEMDAT, as well as system-specific code for the percolation model and graph analysis.

Shape analysis

The `ShapeAnalyzer` class in `gemdat.shape` characterizes the density-weighted distribution of mobile ions around atomic sites, providing descriptors of both occupancy and geometry (Fig. 2l). The `ShapeAnalyzer` is initialized from a reference atomic site structure. The `.from_structure()` constructor symmetrizes the input structure using space group operations, identifying all symmetrically equivalent site positions. This reduces the analysis to unique site types and improves statistics. At each time step, GEMDAT identifies all mobile ions within a specified radius of each site and transforms their coordinates to the site's local coordinate system. Accumulated coordinates over the trajectory build statistics.

`ShapeData` objects are visualized using `shape.plot(bins)`, creating 2D histograms projecting the displacement cloud around the site along three orthogonal directions. This reveals the time-averaged geometry of atomic distribution. To relate site geometry to crystal structure, crystallographic sites can be overlaid on plots using `shape.plot(bins, sites)`. The `ShapeAnalyzer.optimize_sites()` method refines site positions by computing the centroid of each site cluster, improving site assignments when the input structure contains slightly misplaced sites.

Simulation details

Simulation data and analysis for the "Results" subsection "*Role of disorder in argyrodites*" are reused from ref. 8, where the full simulation details can be found.

Simulation data and analysis for the "Results" subsection "*Rotational analysis of plastic crystals*"are reported here for the first time. The initial structures were obtained from published crystallographic models of (β) Na_3PS_4[34] and (γ) Na_3PS_4[33]. For (γ) Na_3PS_4, the sodium distribution on the partially occupied sites was initialized randomly. Density functional theory (DFT) calculations were performed using the Perdew–Burke–Ernzerhof (PBE) functional[70] as implemented in the Vienna Ab initio Software Package (VASP)[25]. Projector augmented wave (PAW)[71] potentials were used with cores of [Be] for Na and [Ne] for S and P. Structure optimization was carried out in a $2 \times 2 \times 2$ supercell for cubic (β) Na_3PS_4 and a $2 \times 1 \times 1$ supercell for orthorhombic (γ) Na_3PS_4 with a plane-wave energy cutoff of 338 eV. Ab initio molecular dynamics (AIMD) simulations were performed in the canonical (NVT) ensemble using a Nosé–Hoover thermostat[72,73], with a plane-wave energy cutoff of 260 eV and a Γ-point-only k-mesh. The temperature was set to 900 K and the timestep to 2 fs, for a total simulation time of 200 ps.

Simulation data and analysis for the "Results" subsection "*Site shape and geometry analysis of sodium-ion conductors*" are reported here for the first time. The initial structure was constructed from Na_3SbS_4 in the $I\bar{4}3m$ space group, taken from the Materials Project database[74]. The $Na_{2.875}W_{0.125}Sb_{0.875}S_4$ structure was obtained by replacing Sb with W at randomly chosen sites. DFT calculations were performed using the PBE functional[70] as implemented in the VASP[25]. PAW[71] potentials were used with cores of [Ne] for Na and S, [Kr] for Sb, and [Xe] for W. Structure optimization was carried out in a $2 \times 2 \times 2$ supercell with a plane-wave energy cutoff of 520 eV. AIMD simulations were performed in the canonical (NVT) ensemble using a Nosé–Hoover thermostat[72,73], with a reduced energy cutoff of 300 eV and a Γ-point-only k-mesh. The temperature was set to 500 K and the timestep to 2 fs, for a total simulation time of 50 ps.

Simulation data and analysis for the "Results" subsection "*Percolation analysis in disordered irreducible antifluorites*" are reused from ref. 9, where the full simulation details can be found.

Simulation data and analysis for the "Results" subsection "*Lithium diffusion in amorphous silicon*" are reported here for the first time. The initial $Li_{36}Si_{64}$ structure was taken from ref. 42. DFT calculations were performed using the PBE functional[70] as implemented in the VASP[25]. PAW[71] potentials were used with cores of [He] for Li and [Ne] for Si. Structure optimization was carried out with a plane-wave energy cutoff of 320 eV. AIMD simulations were performed in the canonical (NVT) ensemble using a Nosé–Hoover thermostat[72,73], with a reduced energy cutoff of 300 eV and a Γ-point-only k-mesh. The temperature was set to 600 K and the timestep to 2 fs, for a total simulation time of 150 ps.

Simulation data for the "Results" subsection "*Lithium diffusion on copper surface*" have been re-analyzed with GEMDAT. Full simulation details are available in ref. 46.

Data availability

The simulation data analyzed in the current study are not publicly available due to their large file sizes, but are available from the corresponding author upon reasonable request.

Code availability

The code is available in the open-source Python package GEMDAT https://github.com/GEMDAT-repos/GEMDAT.git and is archived on Zenodo https://doi.org/10.5281/zenodo.8401669. Illustrations and examples of use can be found in the code repository https://github.com/GEMDAT-repos/gemdat-workshop-2025.git. Documentation for the package is available https://gemdat.readthedocs.io.

Received: 19 January 2026; Accepted: 4 May 2026;
Published online: 16 May 2026

References

1. Yang, H. & Wu, N. Ionic conductivity and ion transport mechanisms of solid-state lithium-ion battery electrolytes: a review. *Energy Sci. Eng.* **10**, 1643–1671 (2022).
2. Famprikis, T., Canepa, P., Dawson, J. A., Islam, M. S. & Masquelier, C. Fundamentals of inorganic solid-state electrolytes for batteries. *Nat. Mater.* **18**, 1278–1291 (2019).
3. Xu, Y., Cai, S., Chi, B. & Tu, Z. Numerical study on improved mass and heat transfer performance in a solid oxide electrolysis cell with sine wave flow field. *Int. J. Hydrogen Energy* **137**, 806–818 (2025).
4. Baiutti, F. et al. Direct measurement of oxygen mass transport at the nanoscale. *Adv. Mater.* **33**, 2105622 (2021).
5. He, X., Zhu, Y. & Mo, Y. Origin of fast ion diffusion in super-ionic conductors. *Nat. Commun.* **8**, 15893 (2017).
6. Jun, K., Chen, Y., Wei, G., Yang, X. & Ceder, G. Diffusion mechanisms of fast lithium-ion conductors. *Nat. Rev. Mater.* **9**, 887–905 (2024).
7. de Klerk, N. J. J., van der Maas, E. & Wagemaker, M. Analysis of diffusion in solid-state electrolytes through MD simulations, improvement of the Li-ion conductivity in β-Li3PS4 as an Example. *ACS Appl. Energy Mater.* **1**, 3230–3242 (2018).
8. Lavrinenko, A. K. et al. Optimizing ionic transport in argyrodites: a unified view on the role of sulfur/halide distribution and local environments. *J. Mater. Chem. A Mater.* **12**, 26596–26611 (2024).
9. Landgraf, V. et al. Disorder-mediated ionic conductivity in irreducible solid electrolytes. *J. Am. Chem. Soc.* **147**, 18840–18852 (2025).
10. Urban, A., Seo, D.-H. & Ceder, G. Computational understanding of Li-ion batteries. *NPJ Comput. Mater.* **2**, 16002 (2016).
11. Dutra, A. C. C., Goldmann, B. A., Islam, M. S. & Dawson, J. A. Understanding solid-state battery electrolytes using atomistic modelling and machine learning. *Nat. Rev. Mater.* **10**, 566–583 (2025).
12. Chan, H. et al. Machine learning classical interatomic potentials for molecular dynamics from first-principles training data. *J. Phys. Chem. C* **123**, 6941–6957 (2019).
13. Shapeev, A. V. Moment tensor potentials: a class of systematically improvable interatomic potentials. *Multiscale Model. Simul.* **14**, 1153–1173 (2016).
14. Bartók, A. P., Payne, M. C., Kondor, R. & Csányi, G. Gaussian approximation potentials: the accuracy of quantum mechanics, without the electrons. *Phys. Rev. Lett.* **104**, 136403 (2010).
15. Mazitov, A. et al. PET-MAD as a lightweight universal interatomic potential for advanced materials modeling. *Nat. Commun.* **16**, 10653 (2025).
16. Kovács, D. P., Batatia, I., Arany, E. S. & Csányi, G. Evaluation of the MACE force field architecture: From medicinal chemistry to materials science. *J. Chem. Phys.* **159**, 044118 (2023).
17. Deng, B. et al. CHGNet as a pretrained universal neural network potential for charge-informed atomistic modelling. *Nat. Mach. Intell.* **5**, 1031–1041 (2023).
18. Gowers, R. et al. *MDAnalysis: a Python Package for the Rapid Analysis of Molecular Dynamics Simulations* 98–105 https://doi.org/10.25080/Majora-629e541a-00e (2016).
19. McGibbon, R. T. et al. MDTraj: a modern open library for the analysis of molecular dynamics trajectories. *Biophys. J.* **109**, 1528–1532 (2015).
20. Humphrey, W., Dalke, A. & Schulten, K. V. M. D. Visual molecular dynamics. *J. Mol. Graph.* **14**, 33–38 (1996).
21. Stukowski, A. Visualization and analysis of atomistic simulation data with OVITO—the Open Visualization Tool. *Model. Simul. Mater. Sci. Eng.* **18**, 015012 (2010).
22. Ong, S. P. et al. Python Materials Genomics (pymatgen): a robust, open-source Python library for materials analysis. *Comput. Mater. Sci.* **68**, 314–319 (2013).
23. Deng, Z., Zhu, Z., Chu, I.-H. & Ong, S. P. Data-driven first-principles methods for the study and design of alkali superionic conductors. *Chem. Mater.* **29**, 281–288 (2017).
24. Morgan, B. J. Site-analysis: Python module for analysing molecular dynamics simulations of ion transport. (Version 1.5.0) [Computer software] https://github.com/bjmorgan/site-analysis (2026).
25. Kresse, G. & Furthmüller, J. Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set. *Comput. Mater. Sci.* **6**, 15–50 (1996).
26. Thompson, A. P. et al. LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Comput. Phys. Commun.* **271**, 108171 (2022).
27. Berendsen, H. J. C., van der Spoel, D. & van Drunen, R. GROMACS: a message-passing parallel molecular dynamics implementation. *Comput. Phys. Commun.* **91**, 43–56 (1995).
28. Hjorth Larsen, A. et al. The atomic simulation environment—a Python library for working with atoms. *J. Phys.: Condens. Matter* **29**, 273002 (2017).
29. McCluskey, A. R., Coles, S. W. & Morgan, B. J. Accurate estimation of diffusion coefficients and their uncertainties from computer simulation. *J. Chem. Theory Comput.* **21**, 79–87 (2025).
30. Deiseroth, H. et al. Li_6PS_5X: a class of crystalline Li-rich solids with an unusually high Li^+ mobility. *Angew. Chem. Int. Ed.* **47**, 755–758 (2008).
31. Gautam, A., Al-Kutubi, H., Famprikis, T., Ganapathy, S. & Wagemaker, M. Exploring the relationship between halide substitution, structural disorder, and lithium distribution in lithium argyrodites ($Li_{6-x}PS_{5-x}Br_{1+x}$). *Chem. Mater.* **35**, 8081–8091 (2023).
32. de Klerk, N. J. J., Rosłoń, I. & Wagemaker, M. Diffusion mechanism of Li argyrodite solid electrolytes for Li-ion batteries and prediction of optimized halogen doping: the effect of li vacancies, halogens, and halogen disorder. *Chem. Mater.* **28**, 7955–7963 (2016).
33. Famprikis, T. et al. A new superionic plastic polymorph of the Na^+ conductor Na_3PS_4. *ACS Mater. Lett.* **1**, 641–646 (2019).
34. Famprikis, T. et al. Insights into the rich polymorphism of the Na^+ ion conductor Na_3PS_4 from the perspective of variable-temperature diffraction and spectroscopy. *Chem. Mater.* **33**, 5652–5667 (2021).
35. Sau, K. & Ikeshoji, T. Origin of fast ion conduction in Na_3PS_4: insight from molecular dynamics study. *J. Phys. Chem. C* **124**, 20671–20681 (2020).
36. Zhang, Z. & Nazar, L. F. Exploiting the paddle-wheel mechanism for the design of fast ion conductors. *Nat. Rev. Mater.* **7**, 389–405 (2022).
37. Nishimura, S., Hayashi, A., Sakuda, A. & Yamada, A. Vacancy-stabilized superionic state in $Na_{3-x}Sb_{1-x}W_xS_4$. *ACS Appl. Energy Mater.* **5**, 14053–14058 (2022).
38. Famprikis, T. et al. Under pressure: mechanochemical effects on structure and ion conduction in the sodium-ion solid electrolyte Na_3PS_4. *J. Am. Chem. Soc.* **142**, 18422–18436 (2020).
39. Pedregosa, F. et al. Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
40. Blessing, R. H., Guo, D. Y. & Langs, D. A. Statistical expectation value of the Debye–Waller factor and *E*(*hkl*) values for macromolecular crystals. *Acta Crystallogr. D Biol. Crystallogr.* **52**, 257–266 (1996).
41. Zhao, W. et al. Irreducible solid electrolytes: new perspectives on stabilizing high-capacity anodes in solid-state batteries. *ACS Energy Lett.* **10**, 5363–5372 (2025).
42. Tritsaris, G. A., Zhao, K., Okeke, O. U. & Kaxiras, E. Diffusion of lithium in bulk amorphous silicon: a theoretical study. *J. Phys. Chem. C* **116**, 22212–22216 (2012).
43. McDowell, M. T., Lee, S. W., Nix, W. D. & Cui, Y. 25th Anniversary Article: understanding the lithiation of silicon and other alloying anodes for lithium-ion batteries. *Adv. Mater.* **25**, 4966–4985 (2013).

44. Strauß, F., Dörrer, L., Bruns, M. & Schmidt, H. Lithium tracer diffusion in amorphous Li_xSi for low Li concentrations. *J. Phys. Chem. C* **122**, 6508–6513 (2018).
45. Hatchard, T. D. & Dahn, J. R. In situ XRD and electrochemical study of the reaction of lithium with amorphous silicon. *J. Electrochem. Soc.* **151**, A838 (2004).
46. Lv, S. et al. Operando monitoring the lithium spatial distribution of lithium metal anodes. *Nat. Commun.* **9**, 2152 (2018).
47. Rehnlund, D. et al. Lithium trapping in alloy forming electrodes and current collectors for lithium based batteries. *Energy Environ. Sci.* **10**, 1350–1357 (2017).
48. Rehnlund, D., Wang, Z. & Nyholm, L. Lithium-diffusion induced capacity losses in lithium-based batteries. *Adv. Mater.* **34**, 2108827 (2022).
49. Rupp, R., Caerts, B., Vantomme, A., Fransaer, J. & Vlad, A. Lithium Diffusion in Copper. *J. Phys. Chem. Lett.* **10**, 5206–5210 (2019).
50. Huang, Y., Chen, H. & Nyholm, L. Influence of lithium diffusion into copper current collectors on lithium electrodeposition in anode-free lithium-metal batteries. *Small* **19**, 2306829 (2023).
51. Kartamyshev, A. I., Poletaev, D. O., Boev, A. O. & Aksyonov, D. A. Weak segregation and accelerated diffusion of Li at twin boundaries in Cu from DFT: Implications for current collectors in Li-ion batteries. *Comput. Mater. Sci.* **230**, 112517 (2023).
52. Frenkel, D., Smit, B. & Ratner, M. A. Understanding molecular simulation: from algorithms to applications. *Phys. Today* **50**, 66–66 (1997).
53. Flyvbjerg, H. & Petersen, H. G. Error estimates on averages of correlated data. *J. Chem. Phys.* **91**, 461–466 (1989).
54. Zhu, Z., Deng, Z., Chu, I.-H., Radhakrishnan, B. & Ping Ong, S. Ab initio molecular dynamics studies of fast ion conductors. In: Shin, D., Saal, J. (eds) *Computational Materials System Design* 147–168 (Springer International Publishing, 2018).
55. Calandrini, V., Pellegrini, E., Calligari, P., Hinsen, K. & Kneller, G. R. nMoldyn - Interfacing spectroscopic experiments, molecular dynamics simulations and models for time correlation functions. *Éc. thémat. Soc. Fr. Neutron.* **12**, 201–232 (2011).
56. McCluskey, A. R., Squires, A. G., Dunn, J., Coles, S. W. & Morgan, B. J. kinisi: Bayesian analysis of mass transport from molecular dynamics simulations. *J. Open Source Softw.* **9**, 5984 (2024).
57. Koettgen, J., Zacherle, T., Grieshammer, S. & Martin, M. Ab initio calculation of the attempt frequency of oxygen diffusion in pure and samarium doped ceria. *Phys. Chem. Chem. Phys.* **19**, 9957–9973 (2017).
58. Mehrer, H. Dependence of Diffusion on Temperature and Pressure. in *Diffusion in Solids: Fundamentals, Methods, Materials, Diffusion-Controlled Processes* 127–149 https://doi.org/10.1007/978-3-540-71488-0_8 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2007).
59. McCluskey, A. R. Is there still a place for linearization in the chemistry curriculum? *J. Chem. Educ.* **100**, 4174–4176 (2023).
60. Levine, B. G., Stone, J. E. & Kohlmeyer, A. Fast analysis of molecular dynamics trajectories with graphics processing units—radial distribution function histogramming. *J. Comput. Phys.* **230**, 3556–3569 (2011).
61. Momma, K. & Izumi, F. *VESTA*3 for three-dimensional visualization of crystal, volumetric and morphology data. *J. Appl. Crystallogr.* **44**, 1272–1276 (2011).
62. Roerdink, J. B. T. M. & Meijster, A. The watershed transform: definitions, algorithms and parallelization strategies. *Fundam. Inform.* **41**, 187–228 (2000).
63. Darve, E. & Pohorille, A. Calculating free energies using average force. *J. Chem. Phys.* **115**, 9169–9183 (2001).
64. Dijkstra, E. W. A note on two problems in connexion with graphs. *Numer. Math.* **1**, 269–271 (1959).
65. Bellman, R. On a routing problem. *Q. Appl. Math.* **16**, 87–90 (1958).
66. Ford, L. R. *Network Flow Theory* (RAND Corporation, 1956).
67. Mehrer, H. Random Walk Theory and Atomic Jump Process. in *Diffusion in Solids: Fundamentals, Methods, Materials, Diffusion-Controlled Processes* 55–67 https://doi.org/10.1007/978-3-540-71488-0_4 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2007).
68. Sau, K. et al. Unlocking the secrets of ideal fast ion conductors for all-solid-state batteries. *Commun. Mater.* **5**, 122 (2024).
69. Vargas-Barbosa, N. M. & Roling, B. Dynamic ion correlations in solid and liquid electrolytes: how do they affect charge and mass transport? *ChemElectroChem* **7**, 367–385 (2020).
70. Perdew, J. P., Burke, K. & Ernzerhof, M. Generalized gradient approximation made simple. *Phys. Rev. Lett.* **77**, 3865–3868 (1996).
71. Nosé, S. A unified formulation of the constant temperature molecular dynamics methods. *J. Chem. Phys.* **81**, 511–519 (1984).
72. Hoover, W. G. Canonical dynamics: equilibrium phase-space distributions. *Phys. Rev. A* **31**, 1695–1697 (1985).
73. Jain, A. et al. Commentary: The Materials Project: a materials genome approach to accelerating materials innovation. *APL Mater.* **1**, 011002 (2013).
74. Blöchl, P. E. Projector augmented-wave method. *Phys. Rev. B* **50**, 17953–17979 (1994).

Acknowledgements

A.V. acknowledges financial support from the eScience Centre under the NLESC. OEC.2022.013. Grant. A.K.L. and M.W. acknowledge the financial support from the 'BatteryNL—Next Generation Batteries based on Understanding Materials Interfaces' project (with project number NWA.1389.20.089) of the NWA research program 'Research on Routes by Consortia (ORC)' funded by the Dutch Research Council (NWO). A.K.L., T.F., V.L., and A.V. acknowledge the use of computational resources of the DelftBlue supercomputer, provided by Delft High Performance Computing Centre (https://www.tudelft.nl/dhpc). A.K.L. and A.V. acknowledge the Dutch national e-infrastructure with the support of the SURF Cooperative using grant nos. EINF-8992, EINF-8989, EINF-15589, and EINF-17290. T.F. acknowledges support from the European Union's HORIZON EUROPE program in the form of a Marie Skłodowska-Curie individual postdoctoral fellowship (project no. 101066486) and the Royal Society in the form of a Newton International Fellowship (NIF\R1\231784). We thank Dr. Benjamin Morgan for insightful discussions and valuable suggestions, and for proposing the interface with the kinisi software package.

Author contributions

A.K.L. conceived the project, performed MD simulations and analysis, contributed to code development and documentation, carried out software testing, developed tutorials, and wrote the manuscript. T.F. conceived the project, performed MD simulations and analysis, and contributed to code development, software testing, tutorial development, and manuscript writing. V.L. performed MD simulations and analysis and contributed to code development and documentation. J.R.H. contributed to code development and documentation and conducted software testing. S.S., V.A., and S.C. contributed to code development and documentation. M.W. conceived and co-supervised the project. A.V. led the project, conceived the study, supervised all stages of its development, and contributed to software testing, documentation, and manuscript writing. All authors reviewed the final manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41524-026-02133-7.

Correspondence and requests for materials should be addressed to Alexandros Vasileiadis.

Reprints and permissions information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

© The Author(s) 2026