

Example Problem

Estimating Ionic Conductivity of Ionic Liquids: Nernst-Einstein and Einstein Formalisms

Ashutosh Kumar Verma, Amey S. Thorat, Jindal K. Shah*

School of Chemical Engineering, Oklahoma State University, Stillwater, OK 74078, United States

E-mail: jindal.shah@okstate.edu

Steps to Compute Transport Properties

Simulation Details

We employed the virtual site OPLS force field to simulate imidazolium-based ILs. Nonbonded interactions were modeled using the Lennard-Jones (LJ) 12-6 and electrostatic potential. We considered two imidazolium-based ionic liquids: 1-ethyl-3-methylimidazolium tetrafluoroborate ($[\text{C}_2\text{mim}][\text{BF}_4]$) and $[\text{C}_2\text{mim}]$ dicyanamide ($[\text{C}_2\text{mim}][\text{DCA}]$). MD simulations were conducted at 298 K using GROMACS 4.5.5 software package. Cubic boxes containing 500 ion pairs were generated using Packmol, with periodicity enforced along the three Cartesian axes. The LJ and electrostatic cut-off were fixed at 13.0 Å, and the long-range contributions to energy and pressure due to LJ interaction were handled by enabling Dispersion correction terms. Particle-Mesh Ewald summations were employed to account for long-range electrostatic interactions with a 13 Å cutoff distance. The equations of motion were integrated using the leap-frog algorithm with a time step of 1 fs. Covalent attached hydrogen bonds

were constrained using the LINCS algorithm. Different thermostats and barostats were used for the equilibration and production run; details are shown in Figure 1. First, we performed an energy minimization using the steepest descent algorithm for 5000 steps followed by a 1.5 ns annealing procedure. Starting at 0 K, the temperature was gradually raised to 298 K over 300 ps. It was then held at 298 K for 200 ps. Subsequently, the temperature was further increased to 523 K over 200 ps and maintained at 523 K for 300 ps. Finally, it was lowered back to 298 K over 200 ps. Further details are presented in Figure 1. The mean and standard deviation of the simulated conductivity were determined from independent calculations.

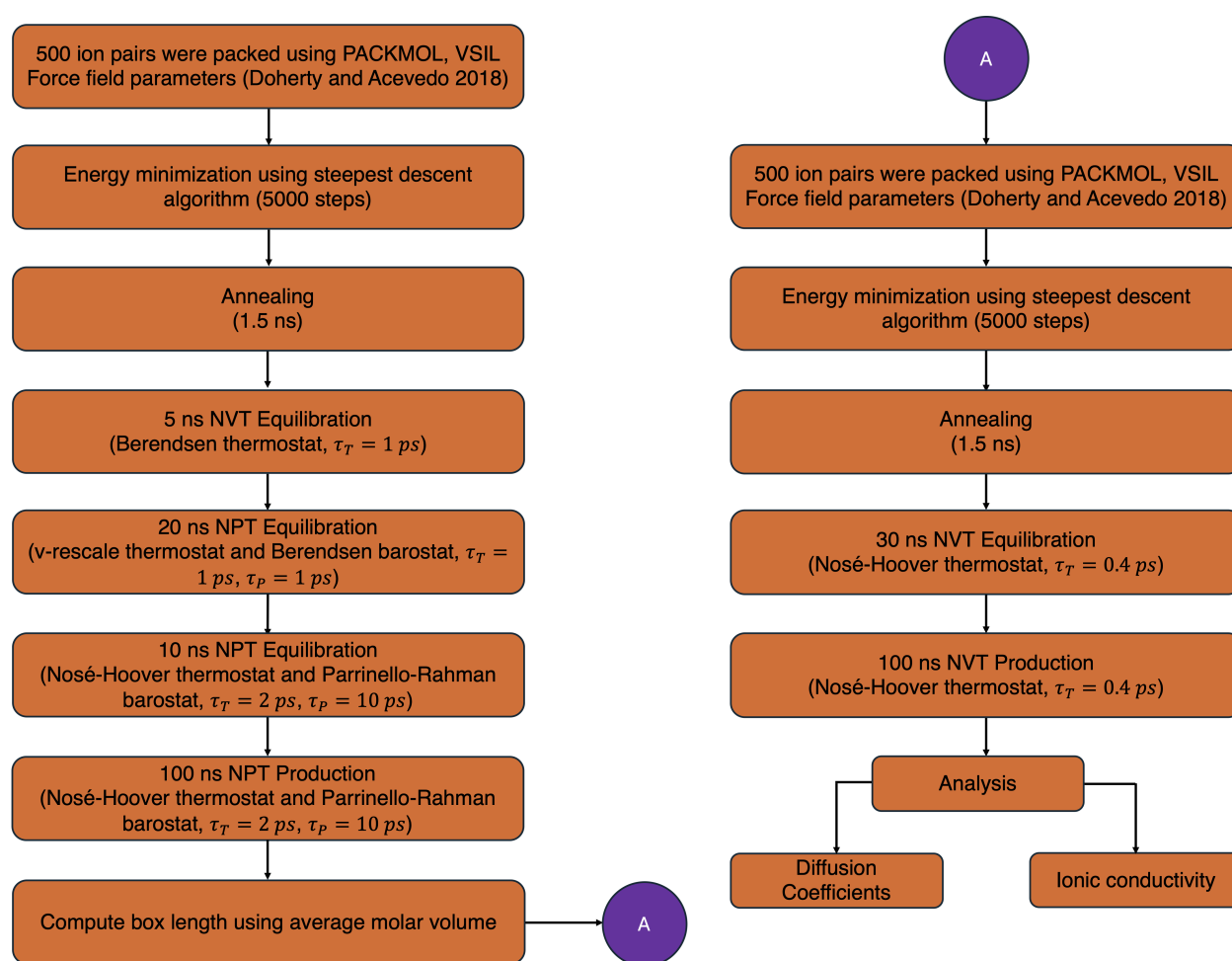


Figure 1: Flowchart depicting the computational framework, comprising a series of MD simulations and post-processing calculations.

Running MD Simulations

1. Run an initial NPT simulation for at least 40 ns (possibly longer, depending on system complexity) to ensure proper equilibration of pressure and volume.
2. Perform a subsequent NVT simulation for your desired simulation length (e.g., consider extending NVT production run beyond 100 ns if needed).

Running Shell and Python Scripts

1. Copy the necessary shell (sh) and Python (py) files to the NVT production run folder.
2. Run the shell script (sh file) first. This script will produce COM files of each ion.
3. Following successful completion of the sh script, proceed to run the Python script (py file). This script will generate CSV file containing time (ps) and product (\AA^2).

The calculation of self-diffusion coefficients, conductivity terms, Nernst-Einstein conductivity, and Einstein conductivity all require repeating the three steps outlined above for each property.

Analysis of Self-diffusion Coefficients

1. Get the MSD of both cations and anions.
2. Create separate logarithmic plots ($\ln(\text{MSD})$ vs. $\ln(t)$) for both the cation and anion MSD data (see Figures 2(b) and 2(d)).
3. In each plot, identify the region where the linear fit approaches a slope of 1 ($\beta(t) \simeq 1$ (Eq. 3). This region typically corresponds to long-time scales where diffusive motion dominates.

4. Within the identified linear region, determine the slope of the MSD vs. t plot for each ion species. Divide this slope by 6 to obtain the self-diffusion coefficient (D) for each ion.

In an isotropic fluid, the self-diffusion coefficient (D) for the ionic species of interest is obtained from the mean-squared displacement (MSD) over individual molecules (Eq. 1)

$$D_k = \frac{1}{6} \frac{d(MSD_k(t))}{dt} \quad (1)$$

The MSD, over a time interval t , is defined as

$$MSD_k(t) = \frac{1}{N_k} \left\langle \sum_{i=1}^{N_k} |r_i^c(t+t_0) - r_i^c(t)|^2 \right\rangle \quad (2)$$

Here N_k denotes the number of k^{th} species, r_i^c is the location of the center of mass of the ions, and t_0 is the different time origins. The $\langle \dots \rangle$ denotes an ensemble average over these time origins.

The self-diffusion coefficients of the ions were obtained using the MSD calculated from the production run trajectory. A logarithmic plot of the MSD versus time (\ln MSD vs. $\ln t$) was constructed. The self-diffusion coefficients were then estimated from the slope of the linear region of this plot. To determine the linear region for the fit, the MSD values were divided into three blocks spanning 0-10 ns, 10-90 ns, and 90-100 ns for the production run (You can use a Python script to find a time window where $\beta(t)$ is close to 1.). Over these blocks, we computed the non-Gaussian parameter $\beta(t)$:

$$\beta(t) = \frac{d \ln(MSD_k(t))}{d \ln(t)} \quad (3)$$

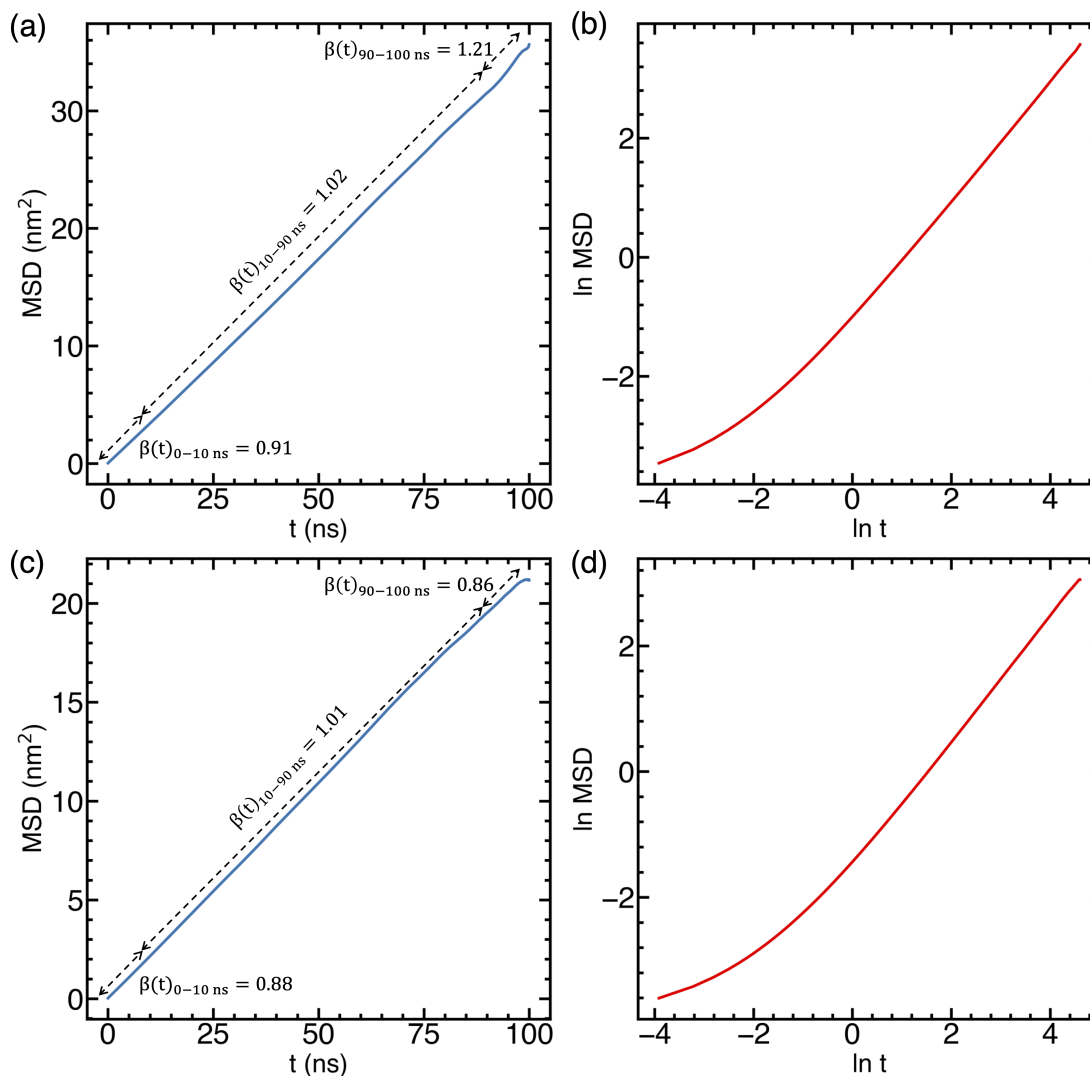


Figure 2: Mean-squared-displacement (MSD) as a function of time and ln MSD vs. ln t for [C₂mim]⁺ (top pane) and for [BF₄]⁻ (bottom pane) for pure [C₂mim][BF₄] at 298 K modeled using VSIL.

Table 1: Self-diffusion coefficients of cation and anion in pure ionic liquids at 298 K. A 10-90 ns time window was considered.

IL	D ₊ (10 ⁻¹¹ m ² /s)	D ₋ (10 ⁻¹¹ m ² /s)
[C ₂ mim][BF ₄]	5.92 ± 0.02	3.46 ± 0.29
[C ₂ mim][DCA]	3.77 ± 0.02	4.99 ± 0.13

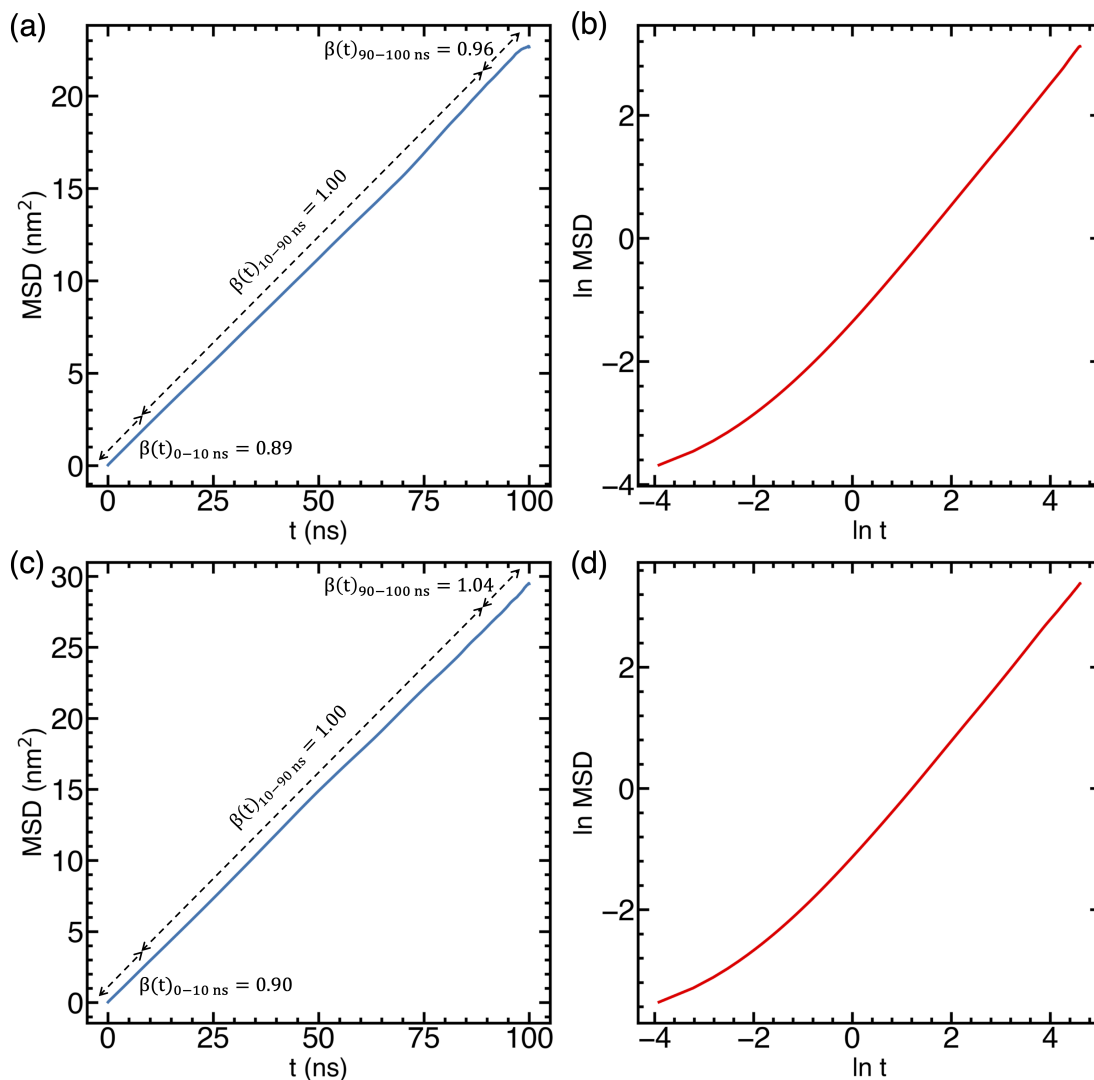


Figure 3: Mean-squared-displacement (MSD) as a function of time and $\ln MSD$ vs. $\ln t$ for $[C_2mim]^+$ (top pane) and for $[DCA]^-$ (bottom pane) for pure $[C_2mim][DCA]$ at 298 K modeled using VSIL.

Analysis of Nernst-Einstein Conductivity

The Nernst-Einstein ionic conductivity (σ_{NE}), for pure ILs, can be calculated using the self-diffusion coefficient of individual ions (Eq. 4)

$$\sigma_{NE} = \frac{e^2}{Vk_B T} (N_+ z_+^2 D_+ + N_- z_-^2 D_-) \quad (4)$$

where e represents the elementary charge, V the volume of the system, k_B denotes the Boltzmann constant while T signifies the temperature of the system. The charges on the cation and anion are given by z_+ and z_- while the respective self-diffusion coefficients are labeled as D_+ and D_- ; N_+ and N_- indicate the number of cations and anions.

Table 2: Nernst-Einstein conductivity of pure ionic liquids at 298 K. Computed using Nernst-Einstein relation derived from self-diffusion coefficients (Eq.4).

IL	σ_{NE} (S/m)
[C ₂ mim][BF ₄]	1.45 ± 0.04
[C ₂ mim][DCA]	1.36 ± 0.02

Analysis of Nernst-Einstein and Einstein Conductivity

1. Get the product of the ions with respect to time Δt .
2. Multiply the product by the factor $\frac{e^2}{6Vk_BT}$.
3. Plot the obtained values against Δt and determine the slope from a time window of initial 1 ns. The slope represents Einstein conductivity.

The Einstein conductivity (σ) for IL can be calculated using the below expression

$$\sigma = \lim_{t \rightarrow \infty} \frac{e^2}{6Vk_BT} \sum_{i,j} z_i z_j \langle [r_i^c(t+t_0) - r_i^c(t)] \cdot [r_j^c(t+t_0) - r_j^c(t)] \rangle \quad (5)$$

Table 3: Einstein conductivity of pure ionic liquids at 298 K. Computed using Einstein relation (Eq.5). A time window of initial 1 ns was used to calculate conductivity.

IL	σ (S/m)
[C ₂ mim][BF ₄]	1.29 ± 0.01
[C ₂ mim][DCA]	1.00 ± 0.12

Next, we computed Nernst-Einstein conductivity by setting $i = j$ in the Eq. 5, we get

$$\sigma_{NE} = \lim_{t \rightarrow \infty} \frac{e^2}{6Vk_BT} \sum_i z_i^2 \langle [r_i^c(t+t_0) - r_i^c(t)]^2 \rangle \quad (6)$$

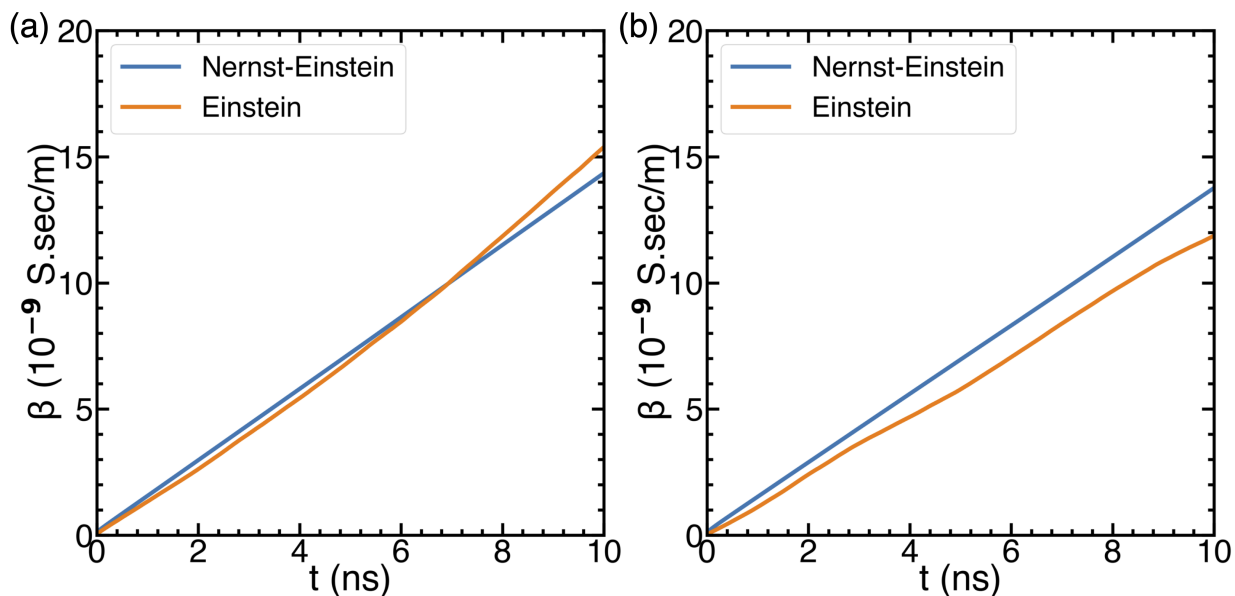


Figure 4: Collective Mean-squared-displacement ($\beta(t)$) as a function of time for (a) [C₂mim][BF₄] and (b) [C₂mim][DCA] at 298 K modeled using VSIL. A time window of 10 ns was used from a 100 ns simulation trajectory.

Table 4: Nernst-Einstein conductivity of pure ionic liquids at 298 K. Computed using Einstein relation by setting $i = j$ (Eq.6). A time window of initial 1 ns was used to calculate conductivity.

IL	σ_{NE} (S/m)
[C ₂ mim][BF ₄]	1.42 ± 0.00
[C ₂ mim][DCA]	1.38 ± 0.01

Einstein Conductivity Using Dipole of Simulation Box

1. Get the product of the ions with respect to time Δt .
2. Multiply the product by the factor $\frac{e^2}{6Vk_B T}$.
3. Plot the obtained values against Δt and determine the slope from a time window of initial 1 ns. The slope represents Einstein conductivity.

Now, we compute Einstein conductivity using the dipole moment of the box. Einstein

conductivity in terms of dipole moment is given as

$$\sigma = \lim_{t \rightarrow \infty} \frac{e^2}{6Vtk_BT} \left\langle \left[\sum_i z_i r_i^c(t + t_0) - \sum_i z_i r_i^c(t) \right] \cdot \left[\sum_i z_i r_i^c(t + t_0) - \sum_i z_i r_i^c(t) \right] \right\rangle \quad (7)$$

Note that $\sum_i z_i r_i^c$ is the dipole moment of the simulation box, and t_0 is the time origin.

Table 5: Einstein conductivity of pure ionic liquids at 298 K. Computed using dipole of the simulation box (see Eq.7). A time window of initial 1 ns was used to calculate conductivity.

IL	σ (S/m)
[C ₂ mim][BF ₄]	1.29 ± 0.01
[C ₂ mim][DCA]	1.00 ± 0.12

Error (%) in Nernst-Einstein and Einstein Conductivity

We employed two different equations for computing Nernst-Einstein and Einstein conductivity. We computed % Error in Nernst-Einstein and Einstein ionic conductivity due to the use of different equations.

$$Error \ \sigma_{NE} = \frac{\sigma_{NE_D} - \sigma_{NE_{EN}}}{\sigma_{NE_D}} \times 100 \quad (8)$$

$$Error \ \sigma = \frac{\sigma_{E_\mu} - \sigma_{E_{EN}}}{\sigma_{E_\mu}} \times 100 \quad (9)$$

where σ_{NE_D} , $\sigma_{NE_{EN}}$, σ_{E_μ} , and $\sigma_{E_{EN}}$ are Nernst-Einstein conductivity computed using Eq. 4, Nernst-Einstein conductivity computed using Eq. 6, Einstein conductivity computed using Eq. 7, and Einstein conductivity computed using Eq. 5, respectively.

Table 6: % Error in Nernst-Einstein and Einstein conductivity of [C₂mim][BF₄] and [C₂mim][DCA] at 298 K.

IL	<i>Error</i> σ_{NE} (%)	<i>Error</i> σ (%)
[C ₂ mim][BF ₄]	1.76	0.0
[C ₂ mim][DCA]	1.30	0.0

Summary

If the Nernst-Einstein approach is of interest to estimate the conductivity, it is recommended to use Eq. 4, by calculating the diffusion coefficients for the individual ionic species in the diffusion regime. Otherwise, the values obtained through Eq. 6 are also in agreement with those obtained using the diffusion coefficient values. On the other hand, if Einstein conductivity is of interest, we recommend using Eq. 7, which is much faster and computationally efficient to implement than Eq. 5.