Supporting Information for:

OCTP: A Tool for On-the-fly Calculation of Transport Properties of Fluids with the Order-n Algorithm in LAMMPS

Seyed Hossein Jamali,[†] Ludger Wolff,[‡] Tim M. Becker,[†] Mariette de Groen,[†] Mahinder Ramdin,[†] Remco Hartkamp,[†] André Bardow,[‡] Thijs J. H. Vlugt,[†] and Othonas A. Moultos[*,†]

[†]Engineering Thermodynamics, Process & Energy Department, Faculty of Mechanical, Maritime and Materials Engineering, Delft University of Technology, Leeghwaterstraat 39, 2628CB Delft, The Netherlands

[‡]Institute of Technical Thermodynamics, RWTH Aachen University, 52056 Aachen, Germany

E-mail: o.moultos@tudelft.nl

Theoretical Background

The general form of the Einstein relations for computing transport properties of a fluid in equilibrium is:[1]

$$\gamma = \left\langle \left(A\left(t\right) - A\left(0\right)\right)^2 \right\rangle / 2t \tag{S1}$$

where γ is the corresponding transport property, $\left(A\left(t\right) - A\left(0\right)\right)^2$ is the mean-squared displacement (MSD) of the time integral of the dynamical variable ($\dot{A}$), and t is time. The angle brackets $\langle\ldots\rangle$ denote an ensemble average. In the following subsections, the Einstein relations for the self-diffusion coefficient, Maxwell-Stefan (MS) diffusion coefficient, shear viscosity, bulk viscosity, and thermal conductivity are provided, along with a brief description of the order-n algorithm for sampling system properties. Simulations for computing the bulk viscosity and thermal conductivity of a fluid are preferentially performed in the microcanonical (NVE) ensemble.[1,2] The procedure for equilibrating the NVE ensemble from an NPT ensemble with specified temperature and pressure is shown in Figure S1.
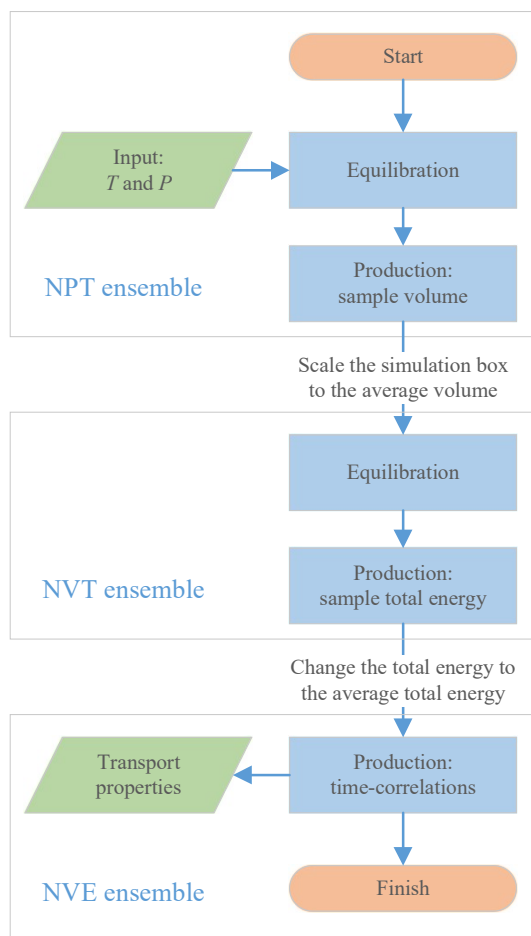
Figure S1: Flowchart of the MD simulations for computing transport properties of a fluid. Each simulation starts with a simulation in the NPT ensemble, from which the average volume of the system is sampled. Then, the simulation box is scaled to this average volume which serves as the input for an equilibration run in the NVT ensemble. Consequently a production NVT run is performed from which the average energy of the system is sampled. This is used to scale the kinetic energy of the last configuration of the NVT run and is used as input for a simulation in the NVE ensemble in which all properties are calculated.

Diffusivity

Self diffusivity

The self-diffusion coefficient of species i $(D_{i,\text{self}})$ in a isotropic three-dimensional system can be defined as the mean-squared displacement of all molecules of species i:[1,3–7]

$$
\begin{aligned}
D_{i,\text{self}} &= \lim_{t \to \infty} \frac{1}{2t} \frac{1}{3N_i} \left\langle \sum_{j=1}^{N_i} \left(\int_0^t \boldsymbol{v}_{j,i}\left(t'\right) \mathrm{d}t' \right) \right\rangle \\
&= \lim_{t \to \infty} \frac{1}{2t} \frac{1}{3N_i} \left\langle \sum_{j=1}^{N_i} \left(\boldsymbol{r}_{j,i}\left(t\right) - \boldsymbol{r}_{j,i}\left(0\right) \right)^2 \right\rangle
\end{aligned}
\tag{S2}
$$

where t is the correlation time, N_i is the number of molecules of species i. $\boldsymbol{v}_{j,i}$ and $\boldsymbol{r}_{j,i}$ are the velocity and position of j-th molecule of species i, respectively. The factor of 3 is for averaging over x, y, and z dimensions.[8] Note that the OCTP plugin is also able to compute the diffusivities for the three different directions individually. This is required for non-isotropic fluids, e.g. liquid crystals.

Maxwell-Stefan diffusivity

The Maxwell-Stefan (MS) diffusion coefficients $(Đ_{\text{MS}})$ of a binary or multicomponent mixture describe the motion of the constituent molecules due to the gradient in chemical potentials of constituent species.[7,9,10] MS diffusivities in a three-dimensional system can be obtained from the Onsager coefficients (Λ_{ij}), computed from the crosscorrelation of the displacement of the molecules of species i and j:[3,5–7,11]

$$
\Lambda_{ij} = \lim_{t \to \infty} \frac{1}{2t} \frac{1}{3N} \left\langle \left(\sum_{k=1}^{N_i} \left(\boldsymbol{r}_{k,i}\left(t\right) - \boldsymbol{r}_{k,i}\left(0\right) \right) \right) \cdot \left(\sum_{l=1}^{N_j} \left(\boldsymbol{r}_{l,j}\left(t\right) - \boldsymbol{r}_{l,j}\left(0\right) \right) \right) \right\rangle
\tag{S3}
$$

where N_i and N_j are the number of molecules of species i and j, respectively and N is the total number of molecules in the mixture. $\boldsymbol{r}_{l,j}$ is the position of the l-th molecule of species i. The relations for computing MS diffusion coefficients from Onsager coefficients for binary,

ternary, and quaternary mixtures are listed in the articles by Krishna and van Baten,[3] and Liu et al.[4,7,12,13] For a binary mixture with mole fractions of x_1 and x_2, a single MS diffusion coefficient can be defined ($Đ_{12,\text{MS}} = Đ_{21,\text{MS}} = Đ_{\text{MS}}$):[3]

$$Đ_{\text{MS}} = \frac{x_2}{x_1}\Lambda_{11} + \frac{x_1}{x_2}\Lambda_{22} - 2\Lambda_{12} \tag{S4}$$

Fick diffusivity

The Fick diffusion coefficient (D_{Fick}) describes the diffusion of molecules in a multicomponent mixture as a result of the gradient in the concentration of constituent species.[7,14] D_{Fick} and $Đ_{\text{MS}}$ are related via the so-called thermodynamic factor (Γ). For a binary mixture, the following algebraic relation holds:[9]

$$D_{\text{Fick}} = \Gamma Đ_{\text{MS}} \tag{S5}$$

where Γ for a multicomponent mixture is defined as:[7,9,10,15]

$$\Gamma_{ij} = \delta_{ij} + \frac{\partial \ln \gamma_i}{\partial \ln x_j}\bigg|_{T,p,\Sigma} \tag{S6}$$

in which γ_i is the activity coefficient of species i and δ_{ij} is the Kronecker delta. The symbol Σ indicates that the partial differentiation of $\ln\gamma_i$ with respect to mole fraction x_j is carried out at constant mole fraction of all other components except the n-th one, so that $\sum_{i=1}^{n} x_i = 1$ during the differentiation.[15] There are different methods for computing the thermodynamic factors such as using equations of state,[3,16] the permuted Widom test particle insertion method,[17,18] and Kirkwood-Buff integrals.[7,19–23] The last method has the advantage that the required parameters are directly accessible from MD simulations. Analytic expressions for Γ_{ij} for various activity coefficient models are derived by Taylor and Kooijman.[15]

Viscosity

Shear viscosity

The shear viscosity (η) is the resistance of a fluid to flow.[24] η can be computed from the time integral over the autocorrelation function of the off-diagonal components of the pressure tensor ($P_{\alpha\beta,\alpha\neq\beta}$):[1,2,25,26]

$$\eta_{\alpha\beta} = \lim_{t\to\infty} \frac{1}{2t}\frac{V}{k_B T}\left\langle\left(\int_0^t P_{\alpha\beta}\left(t'\right)\mathrm{d}t'\right)^2\right\rangle \tag{S7}$$

where V is the volume of the system. The components of the pressure tensor are composed of an ideal and a virial term. The first part is due to the total kinetic energy of particles and the second is constructed from intra- and intermolecular interactions.[1,27,28] In isotropic systems (rotational invariance), the shear viscosities computed from any of the three off-diagonal components of the pressure tensor (P_{xy}, P_{xz}, and P_{yz}) are equal. In isotropic systems, the shear viscosity can also be computed from all components of the traceless pressure tensor ($P_{\alpha\beta}^{os}$):[25,26]

$$\eta = \lim_{t\to\infty} \frac{1}{10\cdot 2t}\frac{V}{k_B T}\left\langle\sum_{\alpha\beta}\left(\int_0^t P_{\alpha\beta}^{os}\left(t'\right)\mathrm{d}t'\right)^2\right\rangle \tag{S8}$$

where[26]

$$P_{\alpha\beta}^{os} = \frac{P_{\alpha\beta} + P_{\beta\alpha}}{2} - \delta_{\alpha\beta}\left(\frac{1}{3}\sum_k P_{kk}\right) \tag{S9}$$

where $\delta_{\alpha\beta}$ is the Kronecker delta. The last term, i.e. one-third of the invariant trace of the pressure tensor,[29] equals the instantaneous kinetic pressure of the system (p). The contribution of the diagonal components of the pressure tensor to the shear viscosity in Equation (S8) is $4/3$. Therefore, the contribution of all 9 components of the traceless pressure tensor results in the factor 10 in the denominator of Equation (S8).

Bulk viscosity

Bulk viscosity is a mysterious quantity - it is a transport coefficient, a property of the continuum description of a flow - that points to the molecular world. It is related to the equilibration of the energy of intramolecular degrees of freedom (rotations, vibrations) with translational energy. For CO_2, the bulk viscosity can be a thousand times larger than the shear viscosity. The reason for such large bulk viscosity values lies in the fact that many molecular collisions are needed to equilibrate the vibrational energy. Clearly, the value of the bulk viscosity depends on how fast flow phenomena and energy transfer occur, and hence the value of the bulk viscosity is frequency dependent. In Molecular Dynamics simulations, the frequency dependent bulk viscocity can be computed from the van Hove correlation function.[30] At zero frequency, the bulk viscosity can be computed from the fluctuations in kinetic pressure (δp):[1]

$$\delta p\left(t\right) = p\left(t\right) - \langle p \rangle \tag{S10}$$

where (p) is the instantaneous pressure and $(\langle p \rangle)$ the ensemble-averaged pressure. Accordingly, the Einstein relation for the bulk viscosity is:[1,31]

$$
\begin{aligned}
\eta_{\mathrm{b}} &= \lim_{t\to\infty} \frac{1}{2t}\frac{V}{k_{\mathrm{B}}T} \left\langle \left(\int_0^t \delta p\left(t'\right) \mathrm{d}t'\right)^2 \right\rangle \\
&= \lim_{t\to\infty} \frac{1}{2t}\frac{V}{k_{\mathrm{B}}T} \left\langle \left(\int_0^t \left(p\left(t'\right) - \langle p \rangle\right) \mathrm{d}t'\right)^2 \right\rangle \\
&= \lim_{t\to\infty} \frac{1}{2t}\frac{V}{k_{\mathrm{B}}T} \left\langle \left(\int_0^t p\left(t'\right) \mathrm{d}t' - \langle p \rangle t\right)^2 \right\rangle \\
&= \lim_{t\to\infty} \frac{1}{2t}\frac{V}{k_{\mathrm{B}}T} \left\langle \left(\int_0^t p\left(t'\right) \mathrm{d}t\right)^2 - 2\langle p \rangle t \left(\int_0^t p\left(t'\right) \mathrm{d}t\right) + \left(\langle p \rangle t\right)^2 \right\rangle
\end{aligned} \tag{S11}
$$

Thermal conductivity

The thermal conductivity (λ) describes the rate of heat conduction in a fluid as a result of the temperature gradient in the system.[24] λ can be computed from the components of the

energy current/heat flux (J_α):[1]

$$\lambda_\mathrm{T} = \lim_{t \to \infty} \frac{1}{2t} \frac{V}{k_\mathrm{B} T^2} \left\langle \left(\int_0^t J_\alpha \left(t' \right) \mathrm{d}t' \right)^2 \right\rangle \tag{S12}$$

The total heat flux consists of two parts: the kinetic heat flux ($\boldsymbol{J}_\mathrm{kinetic}$) and the potential heat flux ($\boldsymbol{J}_\mathrm{potential}$). The total heat flux is computed from:[32]

$$\boldsymbol{J} = \boldsymbol{J}_\mathrm{kinetic} + \boldsymbol{J}_\mathrm{potential} = \frac{1}{2} \sum_{k=1}^{N_t} \boldsymbol{v}_k \left[m\boldsymbol{v}_k^2 + \sum_{j=1, j \neq k}^{N_t} \left(\phi_{jk} + \boldsymbol{r}_{jk} \cdot \boldsymbol{f}_{jk} \right) \right] \tag{S13}$$

where N_t is the total number of atoms in the system. $\boldsymbol{v}_k$ is the velocity vector of atom i. ϕ_{jk}, $\boldsymbol{r}_{jk}$, and $\boldsymbol{f}_{jk}$ are the interaction potential, distance, and force between the two atoms j and k. It is important to note here that Equation (S13) is valid for two-body interaction potentials. For a detailed discussion on thermal conductivity computations in EMD the reader is referred to the work of Kinaci et al.[33]

Order-n algorithm

The order-n algorithm samples time-correlation functions or MSDs at different sampling frequencies.[8,34] Several blocks (buffers) for each sampling frequency are created. For every simulation timestep, it is examined if a buffer has to be updated based on the different sampling frequencies. The oldest element within the buffer is used as the origin to compute the time-correlation function/MSD. The computed quantity is added to an array which will be used to obtain the ensemble-averaged MSD. The oldest element of the buffer is discarded and all other elements are shifted one step to create a space for the newest system property and this procedure continues. Details of the original order-n algorithm and the improved algorithm which is used in this study can be found in the work of Dubbeldam et al.[34]

Installation of the OCTP Plugin in LAMMPS

For installing a new class in LAMMPS, the C++ files (.cpp) and the corresponding header files (.h) should be included in the source directory (/src). As explained in the online manual,[35] by recompiling LAMMPS the new classes and the new commands are available in the newly-made executable file. In the following bash script snippet, the procedure for compiling LAMMPS with the OCTP plugin is shown. In this example LAMMPS files are stored in the home directory ($\sim$), thus the source code is located at $\sim$/`LAMMPS/src/` . OCTP is downloaded in the source directory of LAMMPS as a subdirectory by using the `git clone` command.

```
1 cd ~/LAMMPS/src    #changing the directory to the src directory of LAMMPS
2 git clone https://github.com/omoultosEthTuDelft/OCTP.git #downloading OCTP
3 cp ./OCTP/*.cpp .  #copying all C++ files to the source directory
4 cp ./OCTP/*.h .    #copying all header files to the source directory
5 make mpi           #recompiling LAMMPS to create lmp_mpi executable file
```

The OCTP Plugin Commands

The flowchart in Figure S2 shows how OCTP is invoked in LAMMPS input file to calculate the tranport properties. In the following subsections, the OCTP commands are described in detail following the format of LAMMPS manual.
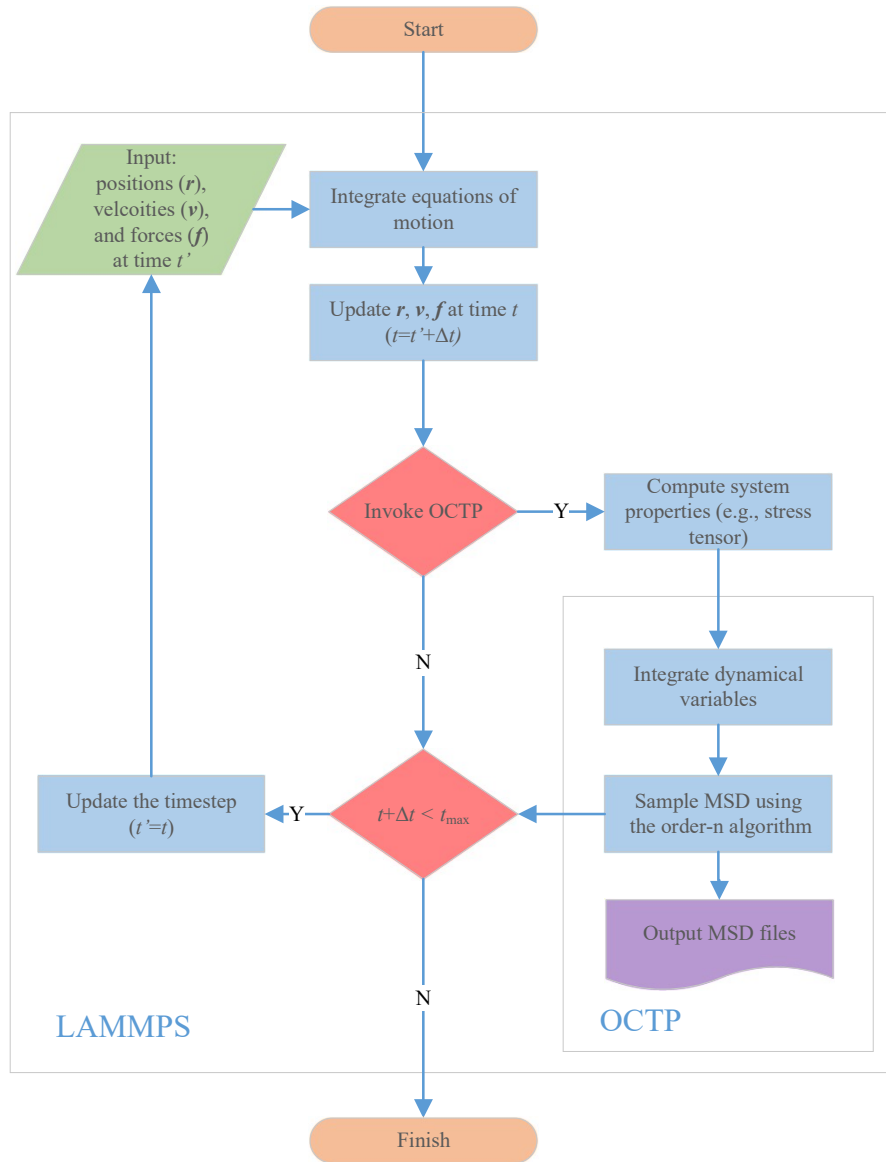
Figure S2: Flowchart showing the use of OCTP in MD simulations with LAMMPS. The sampling frequency specified for the OCTP plugin determines how frequently this plugin is invoked. The system properties are calculated by using the relevant compute commands in LAMMPS. These properties are then integrated and sampled using the order-n algorithm.

compute position command

Syntax

```
1  compute ID group-ID position
```

- ID, group-ID are described in compute command in LAMMPS[35]

- position = style name of this compute command

- no optional keyword

Example

```
1  compute c1 all position
```

Description

This command gathers the upwrapped positions, the atom IDs, and the group masks of all atoms and passes this data as a global vector with size $5N$, where N is the total number of atoms in the system. This vector can be used in the command `fix ordern` to compute the self-diffusion and Onsager coefficients. This command is developed based on the `compute msd` command.[35] This command should be used with the group-ID *all*.

fix ordern command

Syntax

```
1  fix ID group-ID ordern style Nevery Nfreq value keyword values ...
```

- ID, group-ID are described in fix command in LAMMPS[35]

- ordern = style name of this fix command

- style = *diffusivity*, *viscosity*, or *thermalconductivity*

- Nevery = sample system properties every this timestep

- Nfreq = write output data files every this timestep

- value = c_ID, global vector calculated by a compute with **ID**

- keyword values = optional arguments for fix ordern (see Table S1)

Table S1: Optional arguments for the new command `fix ordern`.

Argument	Description	Default
file	name of the output file(s)	diffusion (selfdiffusivity.dat and onsagercoefficient.dat) viscosity (viscosity.dat) thermal conductivity (thermalconductivity.dat)
format	C-style string format to write data	%g
title	text added to the top of output files	-
start	start sampling after this number of steps	0
Dxyz	print x, y, z components of diffusivities if yes	no (applicable only for the style *diffusivity*)
TCconvective	print convective portion of the thermal conductivity if yes for more information see the manual[35] for `compute heat/flux`	no (applicable only for the style *thermalconductivity*)
nb	number of blocks for the order-n algorithm	10
nbe	number of block (buffer) elements for the order-n algorithm	10

Examples

```
1  fix 1 all ordern diffusivity 1000 100000 c_pos nb 10 nbe 20 title "
      diffuison"
2  fix 2 all ordern viscosity 5 100000 c_P title viscosity start 10000
3  fix 3 all ordern thermalconductivity 5 100000 c_HF file thermcond.dat
```

Description

This command computes the MSD of dynamic variables using the order-n algorithm. Three *style*s can be defined: *diffusivity*, *viscosity*, and *thermalconductivity*. The corresponding system properties for each *style* are sampled every *Nevery* timesteps. The ensemble-averaged MSD is output to a file every *Nfreq* timesteps as a function of time. The system properties required for these calculations are provided by a compute command.[35] *value* specifies the ID of this compute command (c_ID). This compute should provide a global compute vector. This fix does not return any values. It only outputs in data files the computed MSD as a function of time. The corresponding transport properties can be calculated by linear regression at timescales where MSD is a linear function of time. This fix command can be restarted. All required data are automatically written to restart files generated by LAMMPS.

In this fix command, dynamical variables of the system are integrated prior to sampling the MSD. For both *viscosity* and *thermalconductivity* styles, Simpson's rule is used to integrate the corresponding dynamical variables. This means that the MSD is only available at every $2 \times Nevery$. Therefore, *Nfreq* should be a multiple of $2 \times Nevery$. To improve the accuracy of integration, it is suggested that *Nevery* should be between 1 and 10 timesteps for *viscosity* and *thermalconductivity*. The sampling frequency when the style *diffusivity* is used can be every hundreds to thousands timesteps. This is because the integration over the positions has been already carried out at each timestep by the MD integration scheme.

The style *diffusivity* is used to compute MSDs for the calculation of self-diffusion and Onsager coefficients. `fix ordern` accepts the global vector provided by the command `compute position`. This compute command provides the positions of all atoms in the system along with their atom IDs and group masks. To distinguish molecules of different species, an atom type of each species should be specified via the `group` command.[35] The only restriction for using groups is that no atom should belong to two different groups at the same time. If n groups are specified, this command outputs MSDs corresponding to n self-diffusion coefficients and $n(n + 1)/2$ Onsager coefficients. Sample output files for self-diffusion and

Onsager coefficients are shown in Tables S2 and S3, respectively. The MSDs reported in the files are defined as:

$$\text{MSD}_{D_{i,\text{self}}} = \frac{1}{6} \left\langle \sum_{j=1}^{N_i} \left(\boldsymbol{r}_{j,i}\left(t\right) - \boldsymbol{r}_{j,i}\left(0\right) \right)^2 \right\rangle \tag{S14}$$

$$\text{MSD}_{\Lambda_{ij}} = \frac{1}{6} \left\langle \left(\sum_{k=1}^{N_i} \left(\boldsymbol{r}_{k,i}\left(t\right) - \boldsymbol{r}_{k,i}\left(0\right) \right) \right) \cdot \left(\sum_{l=1}^{N_j} \left(\boldsymbol{r}_{l,j}\left(t\right) - \boldsymbol{r}_{l,j}\left(0\right) \right) \right) \right\rangle \tag{S15}$$

Note that the number of molecules of species i (N_i) and the total number of molecules (N) (see Equations (S2) and (S3)) are not included in Equations (S14) and (S15). Thus, the obtained MSDs must be divided by N_i and N, respectively. The reported MSDs in the output files have already been divided by 6 (or 2 for the case of diffusion coefficients in directions x, y, and z). The computed diffusion and Onsager coefficients from the provided MSDs have units of distance^2·time^{-1}. For more information on the units available in LAMMPS, the reader is referred to the manual.[35] If the command `units real`, the units of computed diffusion coefficients are in Å^2·fs^{-1}.

Table S2: The first output file produced from the OCTP plugin when *diffusivity* style is chosen. Note that a second output file, corresponding to Osanger coefficients, is created from this style. The reported MSDs can be used to compute the self-diffusion coefficients of an equimolar binary mixture of water-methanol at 298 K and 1 atm. The system contains 500 molecules totally. The indices 1 and 2 represent water and methanol, respectively. The OCTP commands corresponding to this example are provided in Figure S3.

# NOTE: MSDs should be divided by the number of molecules of species i (N_i).		
# NOTE: MSDs have been divided by the factor 6 (or 2, i.e., MSD_x, MSD_y, and MSD_z).		
#Time	MSD__1	MSD__2
1000	78.2073	83.7855
2000	122.834	128.634
3000	163.089	168.577
4000	202.485	207.747
5000	241.402	246.248
6000	279.998	284.295
7000	318.248	321.778
8000	356.302	358.75
9000	394.247	395.392
10000	432.408	432.168
10000	434.872	434.545
20000	813.997	795.878
30000	1187.37	1152.93
40000	1563.5	1509.9
50000	1931.28	1860.65
60000	2300.2	2209.15
70000	2667.63	2554.07
80000	3033.98	2895.75
90000	3404.92	3236.87
100000	3766.9	3572.67
100000	3829.88	3572.68
200000	7424.72	7097
300000	11154.2	10665.4
400000	14869	14095.5
500000	18674.2	17713
600000	22438.8	21123.5
700000	26269.8	24485.7
800000	29761.8	27713.5
900000	33305.7	31037.5
1.00×10^6	36577.7	34378
1.00×10^6	36577.7	34378

Table S3: The second output file produced from the OCTP plugin when *diffusivity* style is chosen. The reported MSDs can be used to compute the Onsager coefficients of an equimolar binary mixture of water-methanol at 298 K and 1 atm. The system contains 500 molecules. The indices 1 and 2 represent water and methanol, respectively. The OCTP commands corresponding to this example are provided in Figure S3.

NOTE: MSDs should be divided by the total number of molecules (N).
NOTE: MSDs have been divided by 6 (or 2, i.e., MSD_x, MSD_y, and MSD_z).

#Time	MSD__1-1	MSD__1-2	MSD__2-2
1000	78.6612	-36.1723	28.4585
2000	122.458	-59.356	43.306
3000	165.841	-82.108	57.7855
4000	215.892	-107.9207	73.1907
5000	260.7	-131.9022	87.6468
6000	304.815	-155.0308	101.111
7000	352.803	-180.0117	114.696
8000	398.088	-204.0983	128.654
9000	445.89	-229.7967	143.198
10000	493.083	-255.4383	157.433
10000	491.533	-261.38	162.495
20000	954.192	-535.2833	327.2
30000	1413.77	-794.2267	472.338
40000	1883.75	-1051.605	615.543
50000	2324.05	-1294.4333	748.723
60000	2678.15	-1505.21	872.827
70000	3127.52	-1761.4667	1021.45
80000	3650.48	-2062.1833	1194.92
90000	4167.9	-2346.8167	1349.84
100000	4646.98	-2611.1	1498.82
100000	4605.6	-2593.9667	1500.61
200000	8312.03	-4613.3667	2603.97
300000	9677.52	-5367.6333	3015.1
400000	9937.93	-5435.9	3013.63
500000	14731.1	-8184.1667	4596.8
600000	20837	-11591.8	6468.67
700000	23588.2	-13261.9333	7484.8
800000	20415.5	-11528.3667	6536.53
900000	19109.2	-10913.0667	6257.53
1.00×10^6	11490.2	-6267.15	3422.1
1.00×10^6	11490.2	-6267.15	3422.1

When the style *viscosity* is chosen, the MSDs of the components of the pressure tensor are computed. This style accepts the global vector calculated by the command `compute pressure`,[35] which are the components of the pressure tensor. The output file consists of 10 columns. The first column contains the correlation time, columns 2 to 7 contain the MSD computed from the three diagonal and the three off-diagonal components of the pressure tensor, column 8 is the average MSD computed from the three off-diagonal components and all six components of the pressure tensor, and the last column contains the MSD corresponding to the bulk viscosity. A sample output file is shown in Table S4. The MSDs computed when style *viscosity* is chosen are defined as:

$$\text{MSD}_\eta = \frac{1}{2} \frac{V}{k_B} \left\langle \left(\int_0^t P_{\alpha\beta}(t') \, \mathrm{d}t' \right)^2 \right\rangle \tag{S16}$$

$$\text{MSD}_{\eta_\mathrm{b}} = \frac{1}{2} \frac{V}{k_\mathrm{B}} \left\langle \left(\int_0^t \delta p(t') \, \mathrm{d}t' \right)^2 \right\rangle \tag{S17}$$

Note that the reported MSDs must be divided by T as in Equations (S7) and (S11). The reported MSDs in the output file have already been divided by the factor 2 (or 10 for the viscosity obtained from all components of the stress tensor). The computed viscosities have units of pressure·time. If the command `units real` is used, the units of computed viscosities are in atm·fs. To convert these units to mPa·s (cP), the computed viscosities should be multiplied by $1.01325 \cdot 10^{-7}$.

Table S4: Output file obtained when the *viscosity* style is chosen. These MSDs can be used to compute the shear and bulk viscosities of an equimolar mixture of water-methanol at 298 K and 1 atm. The system contains 500 molecules. The MSDs of columns 2 to 4 and 5 to 7 are computed from the diagonal and off-diagonal components of the pressure tensor, respectively. Columns 8 and 9 are the average MSDs from the off-diagonal and all components of the pressure tensor, respectively. Column 10 contains the MSDs computed from the fluctuation of the kinetic pressure to calculate the bulk viscosity. The OCTP commands corresponding to this example are provided in Figure S3.

#NOTE: MSDs should be divided by the temperature.
#NOTE: MSDs have been divided by 2 or 10 (i.e., the viscosity computed from all components).

#Time	MSD_xx	MSD_yy	MSD_zz	MSD_xy	MSD_xz	MSD_yz	MSD_off	MSD_all	MSD_bulkvisc
10	1.02×10^9	1.01×10^9	1.01×10^9	1.02×10^9	1.02×10^9	1.04×10^9	1.03×10^9	1.02×10^9	1.15×10^9
20	3.26×10^9	3.22×10^9	3.21×10^9	3.28×10^9	3.26×10^9	3.31×10^9	3.29×10^9	3.26×10^9	4.28×10^9
30	6.77×10^9	6.68×10^9	6.67×10^9	6.82×10^9	6.78×10^9	6.89×10^9	6.83×10^9	6.78×10^9	9.22×10^9
40	1.10×10^{10}	1.09×10^{10}	1.09×10^{10}	1.12×10^{10}	1.11×10^{10}	1.13×10^{10}	1.12×10^{10}	1.11×10^{10}	1.56×10^{10}
50	1.57×10^{10}	1.55×10^{10}	1.54×10^{10}	1.59×10^{10}	1.58×10^{10}	1.60×10^{10}	1.59×10^{10}	1.58×10^{10}	2.32×10^{10}
60	2.09×10^{10}	2.06×10^{10}	2.05×10^{10}	2.12×10^{10}	2.10×10^{10}	2.14×10^{10}	2.12×10^{10}	2.10×10^{10}	3.19×10^{10}
70	2.61×10^{10}	2.56×10^{10}	2.55×10^{10}	2.65×10^{10}	2.62×10^{10}	2.67×10^{10}	2.65×10^{10}	2.62×10^{10}	4.12×10^{10}
80	3.16×10^{10}	3.10×10^{10}	3.08×10^{10}	3.21×10^{10}	3.17×10^{10}	3.24×10^{10}	3.21×10^{10}	3.17×10^{10}	5.13×10^{10}
90	3.71×10^{10}	3.64×10^{10}	3.62×10^{10}	3.77×10^{10}	3.73×10^{10}	3.82×10^{10}	3.77×10^{10}	3.73×10^{10}	6.18×10^{10}
100	4.27×10^{10}	4.20×10^{10}	4.16×10^{10}	4.36×10^{10}	4.30×10^{10}	4.41×10^{10}	4.36×10^{10}	4.30×10^{10}	7.26×10^{10}
100	4.28×10^{10}	4.17×10^{10}	4.14×10^{10}	4.36×10^{10}	4.30×10^{10}	4.40×10^{10}	4.35×10^{10}	4.29×10^{10}	7.28×10^{10}
200	1.11×10^{11}	1.09×10^{11}	1.08×10^{11}	1.13×10^{11}	1.12×10^{11}	1.17×10^{11}	1.14×10^{11}	1.12×10^{11}	1.88×10^{11}
300	1.99×10^{11}	1.94×10^{11}	1.92×10^{11}	2.02×10^{11}	2.01×10^{11}	2.10×10^{11}	2.04×10^{11}	2.01×10^{11}	3.14×10^{11}
400	3.02×10^{11}	2.93×10^{11}	2.89×10^{11}	3.07×10^{11}	3.06×10^{11}	3.20×10^{11}	3.11×10^{11}	3.04×10^{11}	4.58×10^{11}
500	4.17×10^{11}	4.02×10^{11}	3.97×10^{11}	4.24×10^{11}	4.23×10^{11}	4.44×10^{11}	4.30×10^{11}	4.21×10^{11}	6.19×10^{11}
600	5.44×10^{11}	5.20×10^{11}	5.16×10^{11}	5.53×10^{11}	5.52×10^{11}	5.82×10^{11}	5.62×10^{11}	5.48×10^{11}	7.94×10^{11}
700	6.82×10^{11}	6.46×10^{11}	6.45×10^{11}	6.93×10^{11}	6.91×10^{11}	7.31×10^{11}	7.05×10^{11}	6.86×10^{11}	9.83×10^{11}
800	8.29×10^{11}	7.81×10^{11}	7.83×10^{11}	8.41×10^{11}	8.39×10^{11}	8.90×10^{11}	8.57×10^{11}	8.33×10^{11}	1.19×10^{12}
900	9.86×10^{11}	9.22×10^{11}	9.30×10^{11}	9.99×10^{11}	9.95×10^{11}	1.06×10^{12}	1.02×10^{12}	9.89×10^{11}	1.40×10^{12}
1000	1.15×10^{12}	1.07×10^{12}	1.08×10^{12}	1.17×10^{12}	1.16×10^{12}	1.24×10^{12}	1.19×10^{12}	1.15×10^{12}	1.63×10^{12}
1000	1.16×10^{12}	1.08×10^{12}	1.09×10^{12}	1.21×10^{12}	1.18×10^{12}	1.19×10^{12}	1.19×10^{12}	1.16×10^{12}	1.65×10^{12}
2000	3.23×10^{12}	2.81×10^{12}	2.94×10^{12}	3.20×10^{12}	3.13×10^{12}	3.34×10^{12}	3.22×10^{12}	3.13×10^{12}	4.35×10^{12}
3000	5.75×10^{12}	4.78×10^{12}	5.18×10^{12}	5.57×10^{12}	5.31×10^{12}	6.04×10^{12}	5.64×10^{12}	5.48×10^{12}	7.55×10^{12}
4000	8.52×10^{12}	6.88×10^{12}	7.65×10^{12}	8.15×10^{12}	7.50×10^{12}	9.07×10^{12}	8.24×10^{12}	8.02×10^{12}	1.10×10^{13}
5000	1.15×10^{13}	9.19×10^{12}	1.02×10^{13}	1.08×10^{13}	9.60×10^{12}	1.22×10^{13}	1.09×10^{13}	1.07×10^{13}	1.47×10^{13}
6000	1.47×10^{13}	1.17×10^{13}	1.29×10^{13}	1.34×10^{13}	1.18×10^{13}	1.55×10^{13}	1.36×10^{13}	1.34×10^{13}	1.88×10^{13}
7000	1.80×10^{13}	1.43×10^{13}	1.55×10^{13}	1.61×10^{13}	1.39×10^{13}	1.88×10^{13}	1.63×10^{13}	1.61×10^{13}	2.31×10^{13}
8000	2.13×10^{13}	1.70×10^{13}	1.80×10^{13}	1.89×10^{13}	1.61×10^{13}	2.22×10^{13}	1.91×10^{13}	1.89×10^{13}	2.76×10^{13}
9000	2.46×10^{13}	1.98×10^{13}	2.06×10^{13}	2.17×10^{13}	1.85×10^{13}	2.55×10^{13}	2.19×10^{13}	2.18×10^{13}	3.24×10^{13}
10000	2.81×10^{13}	2.27×10^{13}	2.31×10^{13}	2.47×10^{13}	2.08×10^{13}	2.88×10^{13}	2.48×10^{13}	2.47×10^{13}	3.71×10^{13}
10000	3.03×10^{13}	2.27×10^{13}	2.49×10^{13}	2.44×10^{13}	2.03×10^{13}	3.00×10^{13}	2.49×10^{13}	2.53×10^{13}	3.75×10^{13}
20000	6.70×10^{13}	5.02×10^{13}	5.21×10^{13}	5.91×10^{13}	4.22×10^{13}	6.31×10^{13}	5.48×10^{13}	5.55×10^{13}	8.77×10^{13}
30000	1.01×10^{14}	7.99×10^{13}	8.30×10^{13}	9.46×10^{13}	6.23×10^{13}	9.84×10^{13}	8.51×10^{13}	8.63×10^{13}	1.42×10^{14}
40000	1.46×10^{14}	1.19×10^{14}	1.27×10^{14}	1.28×10^{14}	8.78×10^{13}	1.17×10^{14}	1.11×10^{14}	1.19×10^{14}	1.93×10^{14}
50000	1.90×10^{14}	1.55×10^{14}	1.71×10^{14}	1.63×10^{14}	1.12×10^{14}	1.24×10^{14}	1.33×10^{14}	1.49×10^{14}	2.41×10^{14}
60000	2.34×10^{14}	1.99×10^{14}	2.22×10^{14}	1.88×10^{14}	1.38×10^{14}	1.35×10^{14}	1.54×10^{14}	1.80×10^{14}	2.91×10^{14}
70000	2.64×10^{14}	2.38×10^{14}	2.72×10^{14}	2.09×10^{14}	1.58×10^{14}	1.52×10^{14}	1.73×10^{14}	2.07×10^{14}	3.37×10^{14}
80000	2.95×10^{14}	2.71×10^{14}	3.25×10^{14}	2.26×10^{14}	1.76×10^{14}	1.75×10^{14}	1.92×10^{14}	2.34×10^{14}	3.72×10^{14}
90000	3.15×10^{14}	3.04×10^{14}	3.81×10^{14}	2.33×10^{14}	1.98×10^{14}	2.01×10^{14}	2.11×10^{14}	2.60×10^{14}	3.91×10^{14}
100000	3.33×10^{14}	3.43×10^{14}	4.37×10^{14}	2.42×10^{14}	2.23×10^{14}	2.30×10^{14}	2.32×10^{14}	2.87×10^{14}	4.00×10^{14}
100000	5.52×10^{14}	4.58×10^{14}	6.27×10^{14}	3.55×10^{14}	1.28×10^{14}	2.15×10^{14}	2.33×10^{14}	3.58×10^{14}	3.12×10^{14}
200000	7.13×10^{14}	5.80×10^{14}	1.37×10^{15}	3.39×10^{14}	3.96×10^{14}	6.15×10^{14}	4.50×10^{14}	6.25×10^{14}	1.83×10^{14}
300000	7.61×10^{14}	7.07×10^{14}	1.79×10^{15}	6.31×10^{14}	8.20×10^{14}	9.05×10^{14}	7.85×10^{14}	9.06×10^{14}	3.97×10^{14}
400000	1.29×10^{15}	1.27×10^{15}	2.51×10^{15}	8.63×10^{14}	1.30×10^{15}	1.34×10^{15}	1.17×10^{15}	1.38×10^{15}	3.72×10^{14}
500000	1.24×10^{15}	1.72×10^{15}	3.41×10^{15}	1.18×10^{15}	1.82×10^{15}	2.06×10^{15}	1.69×10^{15}	1.86×10^{15}	5.20×10^{14}
600000	8.60×10^{14}	2.09×10^{15}	4.14×10^{15}	1.64×10^{15}	2.59×10^{15}	2.93×10^{15}	2.38×10^{15}	2.38×10^{15}	3.87×10^{14}
700000	9.70×10^{14}	3.24×10^{15}	4.46×10^{15}	1.96×10^{15}	3.21×10^{15}	3.79×10^{15}	2.99×10^{15}	2.95×10^{15}	3.24×10^{14}
800000	5.66×10^{14}	4.30×10^{15}	4.56×10^{15}	2.77×10^{15}	4.02×10^{15}	4.48×10^{15}	3.76×10^{15}	3.51×10^{15}	3.34×10^{14}
900000	1.03×10^{14}	4.24×10^{15}	3.55×10^{15}	3.23×10^{15}	4.95×10^{15}	6.52×10^{15}	4.90×10^{15}	3.99×10^{15}	3.30×10^{14}
1.00×10^6	2.38×10^{15}	8.29×10^{15}	1.79×10^{15}	3.54×10^{15}	6.41×10^{15}	7.82×10^{15}	5.92×10^{15}	5.21×10^{15}	7.59×10^8
1.00×10^6	2.38×10^{15}	8.29×10^{15}	1.79×10^{15}	3.54×10^{15}	6.41×10^{15}	7.82×10^{15}	5.92×10^{15}	5.21×10^{15}	7.59×10^8

The style *thermalconductivity* is used to compute MSDs of the energy current/heat flux, from which the thermal conductivity can be calculated. This style accepts the global vector calculated by `compute heat/flux`.[35] The output file (see an example in Table S5) reports the MSD computed from the three components of the heat flux (x, y, and z), followed by the average of all components as a function of time. The MSDs computed when *thermalconductivity* style is chosen is defined as:

$$\text{MSD}_{\lambda_\text{T}} = \frac{1}{2}\frac{V}{k_\text{B}}\left\langle\left(\int_0^t J_\alpha\left(t'\right)\text{d}t'\right)^2\right\rangle \tag{S18}$$

Note that the reported MSDs must be divided by T^2 as in Equation (S12). The MSDs reported in the output file have already been divided by 2. The computed thermal conductivities have units of energy·distance^{-1}·temperature^{-1}·time^{-1}. If the command `units real` is used, the units of computed thermal conductivities are kcal·mol^{-1}·Å^{-1}·K^{-1}·fs^{-1}. To convert these units to W·m^{-1}·K^{-1}, the computed properties should be multiplied by $6.9477 \cdot 10^4$.

Table S5: Output file obtained when the *thermalconductivity* style is chosen. These MSDs can be used to compute the thermal conductivity of an equimolar mixture of water-methanol at 298 K and 1 atm. The system contains 500 molecules. The MSDs reported in columns 2 to 4 are calculated from the components of the heat flux vector. The average from the three components is listed in column 5. The OCTP commands corresponding to this example are provided in Figure S3.

#Time	MSD_x	MSD_y	MSD_z	MSD_all
# NOTE: MSDs should be divided by (temperature^2).				
# NOTE: MSDs have been divided by 2.				
10	6.14796	6.04591	6.05702	6.08363
20	16.9439	16.6303	16.696	16.7567
30	25.7431	25.2557	25.4648	25.4878
40	30.798	30.2201	30.5988	30.539
50	34.8567	34.1091	34.6857	34.5505
60	41.6606	40.5192	41.4167	41.1989
70	50.3229	48.5958	49.9849	49.6346
80	59.4713	57.1289	59.1014	58.5672
90	67.689	64.8516	67.2842	66.6083
100	74.974	71.7629	74.5068	73.7479
100	75.3937	71.9785	73.9582	73.7768
200	148.499	140.433	147.295	145.409
300	216.157	204.382	211.053	210.531
400	279.572	263.181	268.949	270.567
500	340.476	321.524	324.151	328.717
600	396.735	376.955	379.506	384.399
700	452.05	430.876	433.366	438.764
800	506.239	488.334	481.969	492.181
900	558.893	546.774	530.8	545.489
1000	610.095	604.337	579.614	598.015
1000	612.992	591.925	555.888	586.935
2000	1111.81	1118.97	989.736	1073.51
3000	1595.06	1629.51	1440.87	1555.15
4000	2105.55	2138.1	1834.11	2025.92
5000	2626.31	2591.98	2235.56	2484.62
6000	3055.28	3022.96	2663.39	2913.88
7000	3535.53	3432.9	3093.31	3353.91
8000	4006.1	3822.18	3519.3	3782.52
9000	4460.53	4237.4	3945.58	4214.5
10000	4945.63	4674.69	4402.65	4674.32
10000	5281.72	4627.21	4494.12	4801.02
20000	10046.9	9352.87	10196.5	9865.4
30000	14362.6	13602	16625.1	14863.2
40000	17830.9	17715.1	23617.1	19721
50000	20345.1	23228.6	31729.8	25101.2
60000	23638.3	28144	40613.2	30798.5
70000	25649.1	33091.1	47831.1	35523.8
80000	27750.3	37172.7	52676.2	39199.7
90000	31189.6	43184.8	58610.2	44328.2
100000	34153.9	47337.5	64462.9	48651.4
100000	33917.8	39154.2	76568.8	49880.3
200000	71830.1	66682.6	150588	96367
300000	89178.9	109313	262698	153730
400000	97787.4	130251	425249	217762
500000	82688.3	198854	664211	315251
600000	102158	206431	835281	381290
700000	149280	246085	990271	461879
800000	250730	364304	957791	524275
900000	486497	440610	975287	634131
1.00×10^6	490899	448789	1.29×10^6	744712
1.00×10^6	490899	448789	1.29×10^6	744712

A list of available optional arguments for fix orden is presented in Table S1. The keyword *file* followed by one string (for style *viscosity* and *thermalconductivity*) or two strings (for style *diffusivity*) defines the name(s) of the output file(s). The keyword *format* followed by a string containing a C-style string format defines the precision of the data written in data files. The keyword *title* followed by a string provides the possibility to add an optional text to the header of the output files. The keyword *start* specifies for how many timesteps the sampling is postponed (i.e., the time at which the sampling starts). The keyword *Dxyz* followed by either yes/no enables/disables the functionality to write the x, y, and z components of self-diffusivities and Onsager coefficients in the output files, respectively. The keyword *TCconductive* followed by either yes/no enables/disables the functionality to write the x, y, and z components as well as the average of the convective portion of the thermal conductivity in the output file. For more information, the reader is referred to the LAMMPS manual for the command `compute heat/flux`. The keywords *nb* and *nbe* specify the number of blocks and the number of elements for each block. The default value for both nb and nbe is 10. This means that fix orden can sample MSD for timescales ranging from $1 \times Nevery$ to $10^{10} \times Nevery$ timesteps.

compute rdf/ext command

Syntax

```
1 compute ID group-ID rdf/ext keyword values
```

- ID, group-ID are described in compute command in LAMMPS[35]

- rdf/ext = style name of this compute command

- keyword = *Nbin, Nfreq, file*

Example

```
1 compute c2 all rdf/ext Nbin 2000 Nwrite 100 file rdf.dat
2 fix fc2 all ave/time 1 1 1000 c_c2
```

Description

This command computes the radial distribution function (RDF; g_{ij}) for all pairs of atom types specified by the command `group`.[35] This command should be used with the group-ID *all*. For a system consisting of n groups, $n(n+1)/2$ RDFs can be computed. The computed RDFs are reported in histogram form. The number of bins can be specified via the keyword *Nbin*. The default number of bins is 1000. RDFs computed in MD simulations depend on the system-size. In `compute rdf/ext` the finite-size RDFs are corrected according to the work of van der Vegt and co-workers.[36,37] For each pair of groups, both the finite-size and corrected RDF histograms are output. The command `fix ave/time`[35] is used to invoke `compute rdf/ext`. In the example, `fix ave/time` is invoked with a sampling rate of 1000 timesteps. The keyword *Nwrite*, followed by a number, is used to specify the writing frequency in the output file. The default value for *Nwrite* is 1000. The name of the output file can be modified by the keyword *file*. `compute rdf` is developed based on the `compute msd` and `compute rdf` commands.[35] `compute rdf/ext` has the following two advantages over `compute rdf`: (1) it computes RDFs beyond the cutoff radius; and (2) corrects RDFs for the finite-size effects.

Sample script

A sample script of the LAMMPS input file using the OCTP commands is shown in Figure S3 for the calculation of diffusion coefficients, viscosities, and thermal conductivities of an equimolar water-methanol mixture.

At lines 1-5, the computation of MSDs related to the diffusion coefficients is specified by using the style *diffusivity*. For the binary water-methanol mixture, two groups are defined at lines 2 and 3 to tag all oxygen atoms of water (i.e., type 5) and all oxygen atoms of methanol (i.e., type 4) with the corresponding group IDs of 1 and 2. These IDs are used in the code to identify that the system is binary and, thus compute two MSDs (one for each component) and three Onsager coefficients. At line 4, the command `compute position` is used to obtain the position of all atoms in the simulation box, which is needed as an input for `fix ordern` (line 5). The sampling frequency of positions is specified by the variable ${Ndiff}, while the

```
 1  # computing diffusion coefficients
 2  group 1 type 5        # The Oxygen of the water
 3  group 2 type 4        # The Oxygen of the methanol
 4  compute positions all position
 5  fix fix1 all ordern diffusivity ${Ndiff} ${Nwrite} c_positions
 6
 7  # computing viscosity and average pressure
 8  compute T all temp
 9  compute P all pressure T
10  fix fix2 all ordern viscosity ${Nvis} ${Nwrite} c_P
11
12  # computing thermal conductivity
13  compute KE all ke/atom
14  compute PE all pe/atom
15  compute ST all stress/atom NULL virial
16  compute heatflux all heat/flux KE PE ST
17  fix fix3 all ordern thermalconductivity ${Ntherm} ${Nwrite} c_heatflux
```

Figure S3: Sample script for computing transport properties in LAMMPS using the OCTP plugin. The MSDs related to diffusivities, viscosities, and thermal conductivities are computed using the `fix ordern` command. The variables ${Nvis}, ${Ntherm}, and ${Ndiff} specify the sampling frequency of system properties, while output data are written every ${Nwrite} timesteps. The required system properties for `fix ordern` are provided by the commands `compute position` , `compute pressure`,[35] and `compute heat/flux`.[35] In this script, no optional arguments are used.

output frequency in data files is specified by the variable ${Nwrite}.

At lines 7-10 and 12-16, the OCTP commands for computing MSDs related to the viscosities and thermal conductivities are specified, respectively. The corresponding compute commands are `compute pressure` (line 9) and `compute heat/flux` (line 16). These two commands require the instantaneous temperature of the system (line 8), and per-atom kinetic energy (line 13), potential energy (line 14), and pressure tensor (line 15), respectively.

Finite-size Effects of Transport Properties of a Lennard-Jones Fluid Close to the Critical Point

OCTP is used to investigate the finite-size effects of transport properties of a Lennard-Jones (LJ) fluid. The LJ size ($\sigma = 1$) and energy ($\epsilon = 1$) parameters, along with the mass $= 1$ are in reduced units.[1] To ensure the smooth integration in Equation (S17) the shifted-force 12-6 LJ potential with a cutoff radius (r_c) of 2.5σ is used:[1]

$$
U_{\text{LJ,shifted-force}}\left(r_{ij}\right) = \begin{cases} U_{\text{LJ}}\left(r_{ij}\right) - U_{\text{LJ}}\left(r_c\right) - \left(r_{ij} - r_c\right)\left(\frac{\mathrm{d}U_{\text{LJ}}}{\mathrm{d}r_{ij}}\right)_{r_{ij}=r_c} & r \le r_c \\ 0 & r > r_c \end{cases} \tag{S19}
$$

where r_{ij} is the distance between two particles i and j.[38] The temperature (T) and density (ρ) of MD simulations are 1.00 and 0.325, respectively, which is close to the critical point ($T_c = 0.937$ and $\rho_c = 0.320$).[39] MD simulations were performed in the NVE ensemble according to the procedure shown in Figure S1. Four system sizes were considered: 500, 1000, 2000, and 4000 LJ particles. At least 5 independent simulations were performed for each system size to compute 95% confidence intervals.

The finite-size transport properties are shown in Figure S4. These properties are: (panel a) the self-diffusivity, (panel b) the shear viscosity, (panel c) bulk viscosity, and (panel d) thermal conductivity. Except for shear viscosity, all transport properties show a strong system-size dependency. In agreement with our recent work,[40] the analytic correction proposed by Yeh and Hummer[41,42] does not hold for finite-size effects of self-diffusivities close to the critical point. Bulk viscosity shows the maximum finite-size effect, for which the value in the thermodynamic limit is almost three times the value computed from a system of 500 LJ particles.
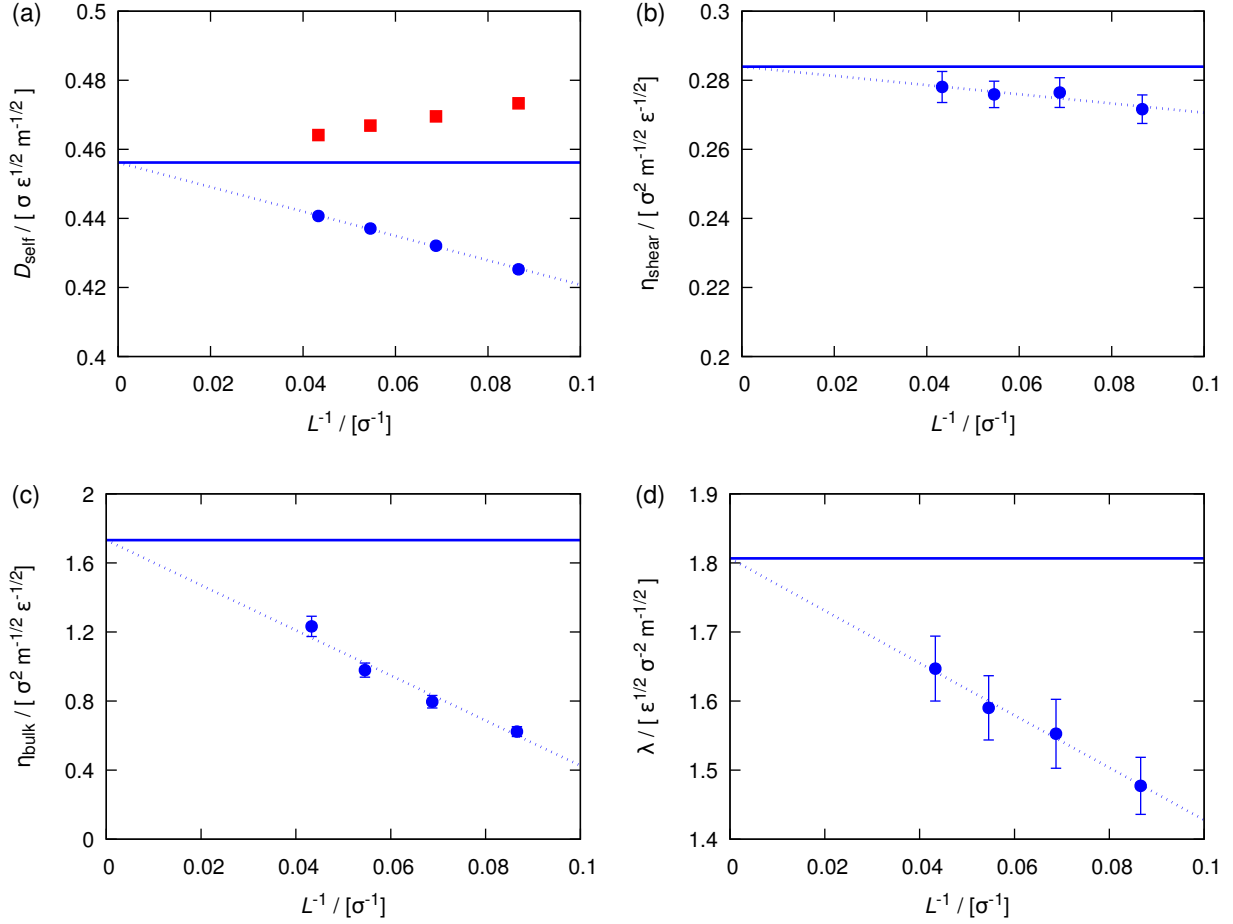
Figure S4: (a) Self-diffusion coefficient, (b) shear viscosity, (c) bulk viscosity, and (d) thermal conductivity of a LJ fluid close to the critical point as a function of the simulation box length (L). Blue circles are the computed finite-size transport properties. The dashed lines indicate extrapolation to the thermodynamic limit, and the solid lines show the values of the transport properties at the thermodynamic limit. Error bars correspond to 95% confidence intervals. Red squares are the corrected self-diffusivities using the analytic relation proposed by Yeh and Hummer.[41] Properties are reported in reduced units[1] with σ, ϵ, and mass equal to 1. The shifted-force potential at a cutoff radius of 2.5σ is used. The critical temperature and density of this LJ fluid is 0.937 and 0.320, respectively.[39] The temperature and density of the system are 1.00 and 0.325, respectively.

Details of Water-Methanol Mixture Simulations

MD simulations were performed to compute the self-diffusivities, Maxwell-Stefan diffusion coefficients, shear viscosity, and thermal conductivity of a water-methanol mixture ($x_{\mathrm{methanol}} = 0.5$) at 298 K and 1 atm. To investigate the computational requirements and scaling behaviour of the OCTP plugin, 6 system sizes of 250, 500, 1000, 2000, 4000, and 8000 molecules were considered. The SPC/E water model[43] and the TraPPE-UA force field[44] for methanol are used. Non-bonded interactions are truncated at a cutoff radius of 10 Å. Analytic tail corrections are included for the calculation of energy and pressure. The Lorentz-Berthelot mixing rules are used for the interactions of unlike atoms.[1] Long-range electrostatic interactions are considered using the particle-particle particle-mesh (PPPM) method with a relative precision of 10^{-6}.[1,35] All simulations are performed according to the procedure shown in Figure S1. The length of the production runs is 1 ns. Dynamical variables are sampled every 1000, 5, and 5 timesteps, for diffusion coefficients, viscosities, and thermal conductivity, respectively. The computed MSDs for Onsager coefficients, shear viscosity, and thermal conductivity are shown in Figure S5. The MS diffusion coefficients are computed from the Onsager coefficients (see Equation (S4)). The transport coefficients are obtained by linear regression at timescales where the slope of MSD is equal to 1 in the log-log plots.[1,34] As shown in Figure S5, at large timescales ($t > 100$ ps), the small number of samples leads to scattered MSDs. The observed jumps at 10 ps and 100 ps are due to the different number of samples at a correlation time in two different blocks.

All input files for LAMMPS to simulate the water-methanol system are provided as a separate Supporting Information file.
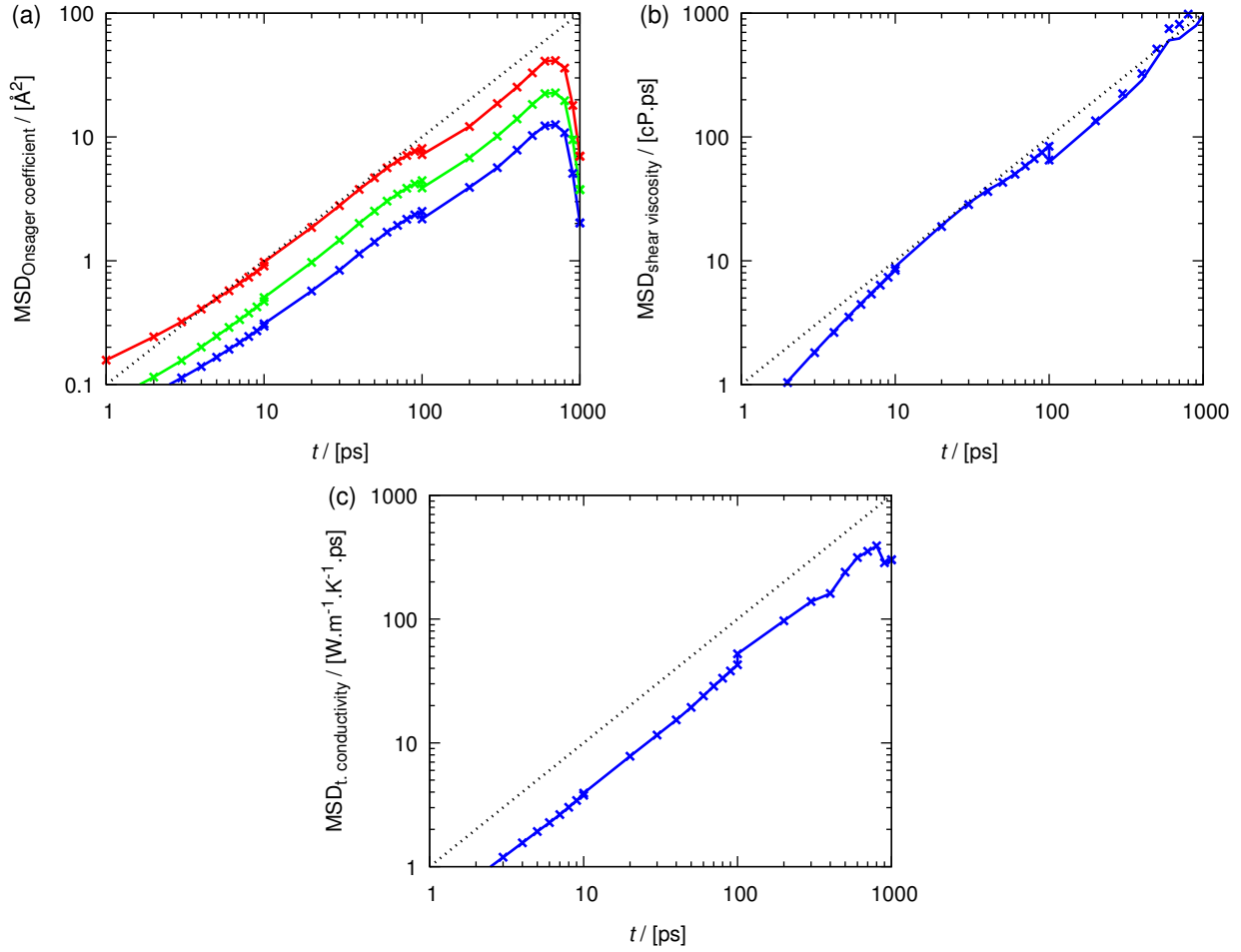
Figure S5: Computed MSDs for (a) Onsager coefficients (blue line: $\Lambda_{\text{water-water}}$, red line: $\Lambda_{\text{methanol-methanol}}$, and green line: $-\Lambda_{\text{water-methanol}}$), (b) shear viscosity, and (c) thermal conductivity for a water-methanol mixture ($x_{\text{methanol}} = 0.5$). Solid lines are computed from the OCTP plugin, while crosses represent the MSDs obtained by postprocessing trajectory files using the order-n algorithm. The dashed lines represent the slope of 1 in these log-log plots.

References

(1) Allen, M. P.; Tildesley, D. J. *Computer Simulation of Liquids*, 2nd ed.; Oxford University Press: Croydon, 2017.

(2) Zwanzig, R. Time-Correlation Functions and Transport Coefficients in Statistical Mechanics. *Annu. Rev. Phys. Chem.* **1965**, *16*, 67–102.

(3) Krishna, R.; van Baten, J. M. The Darken Relation for Multicomponent Diffusion in Liquid Mixtures of Linear Alkanes: An Investigation Using Molecular Dynamics (MD) Simulations. *Ind. Eng. Chem. Res.* **2005**, *44*, 6939–6947.

(4) Liu, X.; Vlugt, T. J. H.; Bardow, A. Maxwell-Stefan Diffusivities in Liquid Mixtures: Using Molecular Fynamics for Testing Model Predictions. *Fluid Phase Equilib.* **2011**, *301*, 110–117.

(5) Liu, X.; Schnell, S. K.; Simon, J.-M.; Bedeaux, D.; Kjelstrup, S.; Bardow, A.; Vlugt, T. J. H. Fick Diffusion Coefficients of Liquid Mixtures Directly Obtained From Equilibrium Molecular Dynamics. *J. Phys. Chem. B* **2011**, *115*, 12921–12929.

(6) Liu, X.; Vlugt, T. J. H.; Bardow, A. Maxwell-Stefan Diffusivities in Binary Mixtures of Ionic Liquids with Dimethyl Sulfoxide (DMSO) and H_2O. *J. Phys. Chem. B* **2011**, *115*, 8506–8517.

(7) Liu, X.; Schnell, S. K.; Simon, J.-M.; Krüger, P.; Bedeaux, D.; Kjelstrup, S.; Bardow, A.; Vlugt, T. J. H. Diffusion Coefficients from Molecular Dynamics Simulations in Binary and Ternary Mixtures. *Int. J. Thermophys.* **2013**, *34*, 1169–1196.

(8) Frenkel, D.; Smit, B. *Understanding Molecular Simulation: From Algorithms to Applications*, 2nd ed.; Academic Press: London, 2002.

(9) Taylor, R.; Krishna, R. *Multicomponent Mass Transfer*; John Wiley & Sons: New York, 1993.

(10) Krishna, R.; van Baten, J. M. Describing Diffusion in Fluid Mixtures at Elevated Pressures by Combining the Maxwell-Stefan Formulation with an Equation of State. *Chem. Eng. Sci.* **2016**, *153*, 174–187.

(11) Liu, X.; Bardow, A.; Vlugt, T. J. H. Multicomponent Maxwell-Stefan Diffusivities at Infinite Dilution. *Ind. Eng. Chem. Res.* **2011**, *50*, 4776–4782.

(12) Liu, X.; Vlugt, T. J. H.; Bardow, A. Predictive Darken Equation for Maxwell-Stefan Diffusivities in Multicomponent Mixtures. *Ind. Eng. Chem. Res.* **2011**, *50*, 10350–10358.

(13) Liu, H.; Maginn, E. A Molecular Dynamics Investigation of the Structural and Dynamic Properties of the Ionic Liquid 1-n-Butyl-3-methylimidazolium Bis(Trifluoromethanesulfonyl)imide. *J. Chem. Phys.* **2011**, *135*, 124507.

(14) Keffer, D. J.; Gao, C. Y.; Edwards, B. J. On the Relationship between Fickian Diffusivities at the Continuum and Molecular Levels. *J. Phys. Chem. B* **2005**, *109*, 5279–5288.

(15) Taylor, R.; Kooijman, H. A. Composition Derivatives of Activity Coefficient Models (For the Estimation of Thermodynamic Factors in Diffusion). *Chem. Eng. Commun.* **1991**, *102*, 87–106.

(16) Krishna, R.; Wesselingh, J. The Maxwell-Stefan Approach to Mass Transfer. *Chem. Eng. Sci.* **1997**, *52*, 861–911.

(17) Balaji, S. P.; Schnell, S. K.; McGarrity, E. S.; Vlugt, T. J. H. A Direct Method for Calculating Thermodynamic Factors for Liquid Mixtures Using the Permuted Widom Test Particle Insertion Method. *Mol. Phys.* **2013**, *111*, 287–296.

(18) Balaji, S. P.; Schnell, S. K.; Vlugt, T. J. H. Calculating Thermodynamic Factors of Ternary and Multicomponent Mixtures Using the Permuted Widom Test Particle insertion Method. *Theor. Chem. Acc.* **2013**, *132*, 1333.

(19) Zhou, Y.; Miller, G. H. Green-Kubo Formulas for Mutual Diffusion Coefficients in Multicomponent Systems. *J. Phys. Chem.* **1996**, *100*, 5516–5524.

(20) Ben-Naim, A. *Molecular Theory of Solutions*; Oxford University Press: Oxford, 2006.

(21) Krüger, P.; Schnell, S. K.; Bedeaux, D.; Kjelstrup, S.; Vlugt, T. J. H.; Simon, J.-M. Kirkwood-Buff Integrals for Finite Volumes. *J. Phys. Chem. Lett.* **2013**, *4*, 235–238.

(22) Krüger, P.; Vlugt, T. J. H. Size and Shape Dependence of Finite-volume Kirkwood-Buff Integrals. *Phys. Rev. E* **2018**, *97*, 051301.

(23) Dawass, N.; Krüger, P.; Schnell, S. K.; Simon, J.-M.; Vlugt, T. J. H. Kirkwood-Buff Integrals from Molecular Simulation. *Fluid Phase Equilib.* **2019**, *486*, 21–36.

(24) Poling, B. E.; Prausnitz, J. M.; O'Connel, J. P. *The Properties of Gases and Liquids*, 5th ed.; McGraw-Hill: Singapore, 2001.

(25) Mondello, M.; Grest, G. S. Viscosity Calculations of n-Alkanes by Equilibrium Molecular Dynamics. *J. Chem. Phys.* **1997**, *106*, 9327.

(26) Tenney, C. M.; Maginn, E. J. Limitations and Recommendations for the Calculation of Shear Viscosity using Reverse Nonequilibrium Molecular Dynamics. *J. Chem. Phys.* **2010**, *132*, 014103.

(27) Heyes, D. M. Pressure Tensor of Partial-Charge and Point-Dipole Lattices with Bulk and Surface Geometries. *Phys. Rev. B* **1994**, *49*, 755–764.

(28) Thompson, A. P.; Plimpton, S. J.; Mattson, W. General Formulation of Pressure and Stress Tensor for Arbitrary Many-Body Interaction Potentials Under periodic Boundary Conditions. *J. Chem. Phys.* **2009**, *131*, 154107.

(29) Meador, W. E.; Miner, G. A.; Townsend, L. W. Bulk viscosity as a relaxation parameter: Fact or fiction? *Phys. Fluids* **1996**, *8*, 258–261.

(30) Evans, D. J.; Morriss, G. *Statistical Mechanics of Nonequilibrium Liquids*, 2nd ed.; Cambridge University Press: Cambridge, 2008.

(31) Meier, K.; Laesecke, A.; Kabelac, S. Transport coefficients of the Lennard-Jones model fluid. III. Bulk viscosity. *J. Chem. Phys.* **2005**, *122*, 014513.

(32) Vogelsang, R.; Hoheisel, C.; Ciccotti, G. Thermal conductivity of the LennardJones liquid by molecular dynamics calculations. *J. Chem. Phys.* **1987**, *86*, 6371–6375.

(33) Kinaci, A.; Haskins, J. B.; Cagin, T. On calculation of thermal conductivity from Einstein relation in equilibrium molecular dynamics. *J. Chem. Phys.* **2012**, *137*.

(34) Dubbeldam, D.; Ford, D. C.; Ellis, D. E.; Snurr, R. Q. A New Perspective on the Order-n Algorithm for Computing Correlation Functions. *Mol. Simul.* **2009**, *35*, 1084–1097.

(35) Sandia National Laboratories, LAMMPS Documentation. 2018; `https://lammps.sandia.gov/doc/Manual.html`.

(36) Ganguly, P.; van der Vegt, N. F. A. Convergence of Sampling Kirkwood-Buff Integrals of Aqueous Solutions with Molecular Dynamics Simulations. *J. Chem. Theory Comput.* **2013**, *9*, 1347–1355.

(37) Milzetti, J.; Nayar, D.; van der Vegt, N. F. A. Convergence of Kirkwood-Buff Integrals of Ideal and Nonideal Aqueous Solutions Using Molecular Dynamics Simulations. *J. Phys. Chem. B* **2018**, *122*, 5515–5526.

(38) Allen, M. P.; Tildesley, D. J. *Computer Simulation of Liquids*; Oxford university press: Bristol, 1989.

(39) Errington, J. R.; Debenedetti, P. G.; Torquato, S. Quantification of order in the Lennard-Jones system. *J. Chem. Phys.* **2003**, *118*, 2256–2263.

(40) Jamali, S. H.; Hartkamp, R.; Bardas, C.; Söhl, J.; Vlugt, T. J. H.; Moultos, O. A. Shear Viscosity Computed from the Finite-Size Effects of Self-Diffusivity in Equilibrium Molecular Dynamics. *J. Chem. Theory Comput.* **2018**, *14*, 5959–5968.

(41) Yeh, I.-C.; Hummer, G. System-Size Dependence of Diffusion Coefficients and Viscosities from Molecular Dynamics Simulations with Periodic Boundary Conditions. *J. Phys. Chem. B* **2004**, *108*, 15873–15879.

(42) Dünweg, B.; Kremer, K. Molecular Dynamics Simulation of a Polymer Chain in Solution. *J. Chem. Phys.* **1993**, *99*, 6983–6997.

(43) Berendsen, H. J. C.; Grigera, J. R.; Straatsma, T. P. The Missing Term in Effective Pair Potentials. *J. Phys. Chem.* **1987**, *91*, 6269–6271.

(44) Chen, B.; Potoff, J. J.; Siepmann, J. I. Monte Carlo Calculations for Alcohols and Their Mixtures with Alkanes. Transferable Potentials for Phase Equilibria. 5. United-Atom Description of Primary, Secondary, and Tertiary Alcohols. *J. Phys. Chem. B* **2001**, *105*, 3093–3104.