ZENO Documentation: Version 5

September 29, 2016

1 Calculations

The ZENO code is composed of two types of calculations: exterior and interior.

1.1 Exterior calculation

The exterior calculation focuses on the computation of electrical properties including the capacitance, the electric polarizability tensor, and the intrinsic conductivity. Once the electrical properties are known, the hydrodynamic properties, including the hydrodynamic radius and the intrinsic viscosity, can be precisely estimated by invoking an electrostatic-hydrodynamic analogy as detailed in Refs. [3, 4, 5]. Other related properties are also determined.

To compute the aforementioned properties for an object requires the solution of Laplace's equation outside the object with appropriate boundary conditions. This is efficiently accomplished by using a Monte Carlo method, which involves (1) creating a launch sphere that encloses the object, (2) launching random walks from the surface of the launch sphere, and (3) determining the fate of such walks—if they hit the object or go to infinity. These walks are exterior to the object, hence the name for the calculation. Each random walk is generated using a method called Walk on Spheres. This algorithm requires generating a sphere for each step in the random walk. The center of this sphere is located at the end of the current random walk; the radius of the sphere is determined by finding the shortest distance between the center of the sphere and the object. Finally, the step in the walk is taken by randomly choosing a point on the surface of the sphere. The process is then repeated. Since the size of spheres will progressively get smaller as the object is approached, a cutoff distance, known as the skin thickness, is required. Without a cutoff distance, the algorithm would continue, at least theoretically, indefinitely. As this is reminiscent of Zeno's paradox of Tortoise and Achilles, the code is named in Zeno's honor. For more details on this method refer to Refs. [3, 8, 9].

1.2 Interior calculation

The interior calculation determines the volume and the gyration tensor for an object using a Monte Carlo method. Specifically, this calculation involves generating random points within the same launch sphere as in the exterior calculation. The location of these points can then

be used to approximate all of the relevant properties. For example, the volume of the object is estimated by the fraction of points inside the object multiplied by the the volume of the launch sphere. The interior calculation is given its name since the points in the interior of the object are essential for computing the properties.

2 Compilation of code

The code is written in `C++` and requires a compiler that supports the `C++11` standard; recent versions of `g++` have been found to work.

The only essential external libraries are the nanoflann Nearest Neighbor library (header-only, does not require compilation) and the SPRNG random number library (requires compilation). These can be obtained from the sources:

`https://github.com/jlblancoc/nanoflann/`

`http://sprng.org/`

You will need to set `NANOFLANN_DIR` and `SPRNG_DIR` at the top of the `Makefile` to point to the locations of these libraries, respectively.

You should then be able to build the executable `zeno` by simply typing `make`.

2.1 MPI support

MPI support is included, but is optional. If MPI libraries are installed on your system, you may be able to build the MPI-enabled executable `zeno-mpi` by simply typing `make mpi`. If this does not work, you will need to change `MPI_CXX`, `MPI_CXXFLAGS`, and `MPI_LDFLAGS` in the `Makefile` to values appropriate for your installation.

2.2 Modifying the code

If you modify the source code, some external utilities may be required. `Gengetopt` is required to modify command-line parameters, while `Bisonc++` and `Flexc++` are required to modify the input file format. These can be obtained from the sources:

`https://www.gnu.org/software/gengetopt/`

`https://fbb-git.github.io/bisoncpp/`

`https://fbb-git.github.io/flexcpp/`

2.3 Simple check

Once the code has been compiled, you can perform a quick self-test by typing `make check`. This will run `zeno` and, if it exists, `zeno-mpi` on various test cases and compare the output against the output from a correctly built version. Floating-point values will be allowed some tolerance to account for differences in compilers, machine precision, etc.

3 Running the code

The code is run using:

```
./zeno [OPTIONS]
```

The output can be printed to file using:

```
./zeno [OPTIONS] >& <name of output file>
```

For example, the command to run ZENO for 1e6 random walks and 1e5 interior samples on an object described in the file `obj.bod` is:

```
./zeno -i obj.bod --numwalks=1000000 --num-interior-samples=100000 >& out.txt
```

Detailed descriptions of all options are below.

3.1 Required options

These inputs are given via the command-line.

3.1.1 Input file

The input or `.bod` file must contain a list of spheres that define the shape of the object. See Sec. 4.1 for details on the content of this file. Note that it can also contain optional quantities as specified in Sec. 4.2. The input file is specified via the command-line argument

```
-i <input file name>
```

3.1.2 Exterior calculation

Either the number of exterior walks, the maximum relative standard deviation of the capacitance, or the maximum relative standard deviation of the mean electric polarizability must be specified. If one of the relative standard deviations are specified, then the calculation will continue performing walks until the relative standard deviation is less than the specified value. One of the following options is required for running the exterior calculation.

```
--numwalks=<number of walks>
--max-rsd-capacitance=<maximum relative standard deviation of capacitance>
--max-rsd-polarizability=<maximum relative standard deviation of mean electric
polarizability>
```

3.1.3 Interior calculation

Either the number of interior samples or the maximum relative standard deviation of the volume must be specified. If the relative standard deviation is specified, then the calculation

will continue performing walks until the relative standard deviation is less than the specified value. One of the following options is required for running the interior calculation.

```
--num-interior-samples=<number of interior samples>
--max-rsd-volume=<maximum relative standard deviation of volume>
```

3.2 Description of command-line options

`-h,`	`--help`	Print help and exit
`-V,`	`--version`	Print version and exit
`-i,`	`--input-file=string`	Input file name
	`--num-walks=int`	Number of exterior walks to perform
	`--num-interior-samples=int`	Number of interior samples to take
	`--max-rsd-capacitance=double`	Perform exterior walks until the relative standard deviation of the capacitance drops below this value. Relative standard deviation is defined as (Standard Deviation/Mean) × 100%
	`--max-rsd-polarizability=double`	Perform exterior walks until the relative standard deviation of the mean electric polarizability drops below this value. Relative standard deviation is defined as (Standard Deviation/Mean) × 100%
	`--max-rsd-volume=double`	Take interior samples until the relative standard deviation of the volume drops below this value. Relative standard deviation is defined as (Standard Deviation/Mean) × 100%
	`--min-num-walks=int`	Minimum number of exterior walks to perform when using max-rsd stopping conditions (default=1000)
	`--min-num-interior-samples=int`	Minimum number of interior samples to take when using max-rsd stopping conditions (default=10000)
	`--num-threads=int`	Number of threads to use (default=Number of logical cores)
	`--seed=INT`	Seed for the random number generator (default=Randomly set)
	`--surface-points-file=string`	Name of file for writing the surface points from exterior calculation
	`--interior-points-file=string`	Name of file for writing the interior sample points
	`--print-counts`	Print statistics related to counts of hit points

`--print-benchmarks`	Print detailed RAM and timing information

4 Input file

The input file, also known as a `.bod` file, contains the description of the object and some additional input parameters.

4.1 Defining the object

The object of interest must be described by a collection of spheres, which may or may not be overlapping. The shape of the object is defined in the `.bod` file. This file should contain lines of

```
SPHERE x y z r
```

where `x`, `y`, and `z` are the x, y, and z positions of the center of the sphere, respectively. `r` is the radius. For example, a `.bod` file that contains the following describes an object composed of two spheres: one of radius 2 at $x = 0$, $y = 0$, and $z = 1$ and one of radius 3 at $x = 0$, $y = 0$, and $z = -1$.

```
SPHERE 0 0 1 2
SPHERE 0 0 -1 3
```

4.2 Optional inputs

4.2.1 Launch radius

Command:	`rlaunch double`
Explanation:	Sets the radius, which is radius of the sphere from which random walks are launched. The radius must be large enough to enclose the entire object.
Default value:	The smallest radius that encloses the smallest axis-aligned bounding-box of the object.
Example:	`rlaunch 20` means that the launch radius is 20.

4.2.2 Skin thickness

Command:	`st double`
Explanation:	Sets the skin thickness. A random walker is assumed to have hit the surface of the object if the distance between the surface and the walker is less than the skin thickness.
Default value:	1e-6 times the launch radius
Example:	`st 0.01` means that the skin thickness is 0.01.

4.2.3 Units for length

Command: `hunits double string`
Explanation: Specifies the units for the length for all objects.
Options: The string can take the following values:

- `m` (meters)
- `cm` (centimeters)
- `nm` (nanometers)
- `A` (Angstroms)
- `L` (generic or unspecified length units)

Default value: 1 `L`
Example: `hunits 10 cm` means that a length of 1 for an object is equivalent to 10 cm.

4.2.4 Temperature

Command: `temp double string`
Explanation: Specifies the temperature, which is used for computing the diffusion coefficient.
Options: The string can take the following values:

- `C` (Celsius)
- `K` (Kelvin)

Default value: None
Example: `temp 20 C` means that the temperature is 20°C.

4.2.5 Mass

Command: `mass double string`
Explanation: Specify the mass of the object, which is used for computing the intrinsic viscosity in conventional units and the sedimentation coefficient.
Options: The string can take the following values:

- `Da` (Daltons)
- `kDa` (kiloDaltons)
- `g` (grams)
- `kg` (kilograms)

Default value: None
Example: `mass 2 g` means that the mass of the object is 2 grams.

4.2.6 Solvent viscosity

Command:	`viscosity double string`
Explanation:	Specify the solvent viscosity, which is used for computing the diffusion coefficient, the friction coefficient, and the sedimentation coefficient.
Options:	The string can take the following values:

- `p` (poise)
- `cp` (centipoise)

Default value:	None
Example:	`viscosity 2 cp` means that the solvent has a viscosity of 2 centipoise.

4.2.7 Buoyancy factor

Command:	`bf double`
Explanation:	Specify the buoyancy factor, which is used for computing the sedimentation coefficient.
Default value:	None
Example:	`bf 2` means that the buoyancy factor is 2.

5 Output

Depending on the quantity, the requirements for calculations are different. Some quantities require the exterior calculation, some the interior, and some both. Additionally, some quantities are direct outputs, while others are indirect—requiring only the direct outputs coupled with algebraic expressions. Finally, some special quantities require optional input such as the temperature. Each of these dependencies are listed.

5.1 Capacitance

Equation:	$C = tR$
Calculation:	Direct exterior
Explanation:	t is the fraction of random walks that hit the object as opposed to go to infinity, and R is the radius of the launch sphere.
Uncertainty:	Determined by directly estimating the variance of t and then applying propagation of uncertainties.
Units:	Length

5.2 Electric polarizability tensor

Equation:	α is expressed in Ref. [9].
Calculation:	Direct exterior
Explanation:	Several different counting variables from the exterior calculation are combined to evaluate this quantity.
Uncertainty:	Determined by directly estimating the variances of t, u, v, and w from Ref. [9] and then applying propagation of uncertainties.
Units:	Length cubed

5.3 Eigenvalues of electric polarizability tensor

Calculation:	Indirect exterior
Explanation:	The eigenvalues of the previously computed electric polarizability tensor α are determined.
Uncertainty:	Determined via propagation of uncertainties.
Units:	Length cubed

5.4 Mean electric polarizability

Equation:	$\langle\alpha\rangle = \mathrm{Tr}(\alpha)/3$
Calculation:	Indirect exterior
Explanation:	The trace of the previously computed electric polarizability tensor α is computed and then divided by three.
Uncertainty:	Determined via propagation of uncertainties.
Units:	Length cubed

5.5 Intrinsic conductivity

Equation:	$[\sigma]_\infty = \langle\alpha\rangle/V$
Explanation:	α is the electric polarizability tensor; V is the volume.
Calculation:	Indirect exterior and interior
Uncertainty:	Determined via propagation of uncertainties.
Units:	None

5.6 Volume

Equation:	$V = p\frac{4}{3}\pi R^3$
Explanation:	p is the fraction of points inside the object; R is the radius of the launch sphere.
Calculation:	Direct interior
Uncertainty:	Determined by directly estimating the variance of p and then applying propagation of uncertainties.
Units:	Length cubed

5.7 Gyration tensor

Equation:	**S** is expressed in Ref. [10]
Explanation:	Several different counting variables from the interior calculation are combined to evaluate this quantity.
Calculation:	Direct interior
Uncertainty:	Determined by directly estimating the variance of sums and the sums of products of interior sample point coordinates, and then applying propagation of uncertainties.
Units:	Length squared

5.8 Eigenvalues of gyration tensor

Calculation:	Indirect interior
Explanation:	The eigenvalues of the previously computed gyration tensor **S** are determined.
Uncertainty:	Determined via propagation of uncertainties.
Units:	Length squared

5.9 Capacitance of a sphere of the same volume

Equation:	$C_0 = (3V/(4\pi))^{1/3}$
Calculation:	Indirect interior
Explanation:	V is the volume of the object.
Uncertainty:	Determined via propagation of uncertainties.
Units:	Length

5.10 Hydrodynamic radius

Equation:	$R_h = q_{R_h} C$
Explanation:	$q_{R_h} \approx 1$, and C is the capacitance.
Calculation:	Indirect exterior
Uncertainty:	Determined via propagation of uncertainties assuming the standard deviation of q_{R_h} is 0.01.
Units:	Length

5.11 Prefactor relating average polarizability to intrinsic viscosity

Equation:	q_η varies slowly with shape and is expressed in Ref. [8].
Calculation:	Indirect exterior
Explanation:	The electric polarizability tensor plus a complicated Padé approximate is used to determine this quantity.
Uncertainty:	$0.015 q_\eta$
Units:	None

5.12 Viscometric radius

Equation:	$R_v = (3q_\eta \langle\alpha\rangle/(10\pi))^{1/3}$
Explanation:	q_η is the prefactor for the intrinsic viscosity, and $\langle\alpha\rangle$ is the mean polarizability.
Calculation:	Indirect exterior
Uncertainty:	Determined via propagation of uncertainties.
Units:	Length

5.13 Intrinsic viscosity

Equation:	$[\eta] = q_\eta [\sigma]_\infty$
Explanation:	q_η is a prefactor, and $[\sigma]_\infty$ is the intrinsic conductivity.
Calculation:	Indirect exterior and interior
Uncertainty:	Determined via propagation of uncertainties.
Units:	None

5.14 Intrinsic viscosity with mass units

Equation:	$[\eta]_m = q_\eta \langle\alpha\rangle/m$
Explanation:	q_η is the prefactor, α is the polarizability tensor, and m is the specified mass.
Calculation:	Indirect exterior
Uncertainty:	Determined via propagation of uncertainties.
Units:	Length cubed / mass
Requirements:	Specified mass.

5.15 Friction coefficient

Equation:	$f = 6\pi\eta R_h$
Explanation:	η is the solvent viscosity, and R_h is the hydrodynamic radius.
Calculation:	Indirect exterior
Uncertainty:	Determined via propagation of uncertainties.
Units:	Mass / time
Requirements:	Specified length scale and solvent viscosity.

5.16 Diffusion coefficient

Equation:	$D = k_B T/f$
Explanation:	k_B is the Boltzmann constant, T is the temperature, and f is the friction coefficient.
Calculation:	Indirect exterior
Uncertainty:	Determined via propagation of uncertainties.
Units:	Length squared / time
Requirements:	Specified length scale, solvent viscosity, and temperature.

5.17 Sedimentation coefficient

Equation:	$s = mb/f$
Explanation:	m is the mass, b is the buoyancy factor, and f is the friction coefficient.
Calculation:	Indirect exterior
Uncertainty:	Determined via propagation of uncertainties.
Units:	Time
Requirements:	Specified length scale, solvent viscosity, mass, and buoyancy factor.

6 Units

6.1 Notation

We use the following definitions:

Command	`double`	`string`
`hunits`	l	**l_unit**
`temp`	T	**T_unit**
`mass`	m	**m_unit**
`viscosity`	η	η**_unit**

6.2 Properties with length scaling

If the units of the output is length to the x power, and r is the result prior to unit conversion, then the output is

$$r \cdot l^x \ \textbf{l\_unit}^x.$$

6.3 Intrinsic viscosity with mass units

If **l_unit** = `L`, then

$$[\eta]_m = q_\eta \langle \alpha \rangle / m \cdot l^3 \ \textbf{l\_unit}^3/\textbf{m\_unit}$$

else

$$[\eta]_m = q_\eta \langle \alpha \rangle / m \cdot l^3 a_l^3 a_m \ \texttt{cm}^3/\texttt{g}.$$

6.4 Friction coefficient

If **l_unit** is different than `L`, then

$$f = 6\pi\eta R_H \cdot 10^{-2} l a_l a_\eta \ \texttt{dyne·s/cm}.$$

6.5 Diffusion coefficient

If **l_unit** is different than `L`, then

$$D = k_B(T + a_T)/(6\pi\eta R_H) \cdot 10^9 l^{-1} a_l^{-1} a_\eta^{-1} \ \texttt{cm}^2/\texttt{s}.$$

6.6 Sedimentation coefficient

If **l_unit** is different than `L`, then

$$s = mb/(6\pi\eta R_H) \cdot 10^{15} l^{-1} a_l^{-1} a_\eta^{-1} a_m^{-1} \texttt{ Sved}.$$

6.7 Conversion tables

l_unit	a_l	uncertainty
`m`	100	0
`cm`	1	0
`nm`	10^{-7}	0
`A`	10^{-8}	0

T_unit	a_T	uncertainty
`C`	273.15	0
`K`	0	0

m_unit	a_m	uncertainty
`Da`	$6.02214 \cdot 10^{23}$	10^{18}
`kDa`	$6.02214 \cdot 10^{20}$	10^{15}
`g`	1	0
`kg`	10^{-3}	0

η**_unit**	a_η	uncertainty
`cP`	1	0
`P`	100	0

k_B	uncertainty
$1.38065 \cdot 10^{-23}$	10^{-28}

7 Historical notes

The development of the current version (v5) of the code was motivated by the need to modernize the code base and to significantly speedup the computation. ZENO up to version 3.x was written in Fortran77. These versions were developed at Stevens Institute of Technology by Dr. Marc Mansfield and coworkers; they can be downloaded from `https://web.stevens.edu/zeno/`. For version 4, the code was ported to Fortran 2008. For version 5, ZENO is implemented from scratch in C++. This new version takes advantage of parallelism to deliver up to four orders of magnitude speedup compared to the Fortran versions as described in Ref. [6]. Note that the algorithms in ZENO have been extensively tested previously, e.g., Refs. [8, 9].

8 Contact information

The developers may be contacted at `zeno@nist.gov`.

9 Validation

In order to validate the current version of the code, as well as provide examples for users, five different systems were considered: (1) two touching spheres of radius 1, (2) two touching spheres: one of radius 1 and one of 1/4, (3) a torus with a minor radius of 1 and major radius of 4, (4) a "polymer" composed of 20 beads, and (5) the protein lysozyme. In the case of the torus, it is represented by a collection of 416 uniformly distributed spheres. For lysozyme, the pdb file 1LYD (see Refs. [1, 2]) was used, and a 5 Å sphere was placed at the center of each alpha carbon of each amino acid. This procedure was previously used in Ref. [7]. All `.bod` files used in testing can be found in `src/cpp/SelfTests`.

For systems composed of two spheres or a torus, the analytic values from Ref. [9] are used to define the true value, or ground truth. Since ground truths do not exist for the "polymer" and protein, the output associated with 1e11 walks and 1e11 interior samples was used as a surrogate. The ground truth is taken as the mean from three runs, and the accuracy is determined using the expanded uncertainty (see below) from those three runs. Note that the t statistic is modified ($t = 4.302653$). When differences cannot be determined due to limited accuracy, results are omitted.

For each property, the following quantities were calculated: the mean ($\bar{y} = \sum_i y_i/N$), difference between ground truth and mean ($\Delta = |\mu - \bar{y}|$), relative difference ($(\Delta/\mu) \cdot 100\,\%$), standard deviation ($s = \sum_i (y_i - \bar{y})^2/(N-1)$), standard uncertainty ($s/\sqrt{N}$), expanded uncertainty ($st/\sqrt{N}$), and relative expanded uncertainty ($st/(\mu\sqrt{N}) \cdot 100\,\%$). For the equations, the ground truth is represented by μ, the number of samples is N, and the value of observation i of a property is y_i. The expanded uncertainty was computed using a 95 % confidence interval ($t = 2.009575$). Each of the properties was found to have a normal distribution. As a result, the mean and standard deviation should define the distribution.

Results from the aforementioned tests can be found in the following tables. Each test was run a total of 50 different times for 1e6 walks and 1e6 interior samples, as well as 1e7 walks and 1e7 interior samples. Note that these two quantities should be chosen carefully as they affect the uncertainty of the calculation; choosing larger values reduces the expanded uncertainty.

Table 1: **Two touching spheres of radius 1**

Property	Ground Truth	Units	Mean	Difference	Rel. Diff.	Std. Dev.	Std. Uncert.	Expand Uncert.	Rel. Exp. Uncert.
			number of walks = 1e6; number of interior samples = 1e6						
C	1.38629	L	1.38629	1e-7	0.000 %	1.14e-3	1.6e-4	3.2e-4	0.023 %
Eigenvalue of α	22.658	L^3	22.568	9.0e-2	0.396 %	8.0e-2	1.1e-2	2.3e-2	0.101 %
Eigenvalue of α	22.658	L^3	22.745	8.7e-2	0.383 %	9.4e-2	1.3e-2	2.7e-2	0.118 %
Eigenvalue of α	60.421	L^3	60.427	6e-3	0.011 %	1.91e-1	2.7e-2	5.4e-2	0.090 %
$\langle\alpha\rangle$	35.246	L^3	35.247	8e-4	0.002 %	6.6e-2	9e-3	1.9e-2	0.054 %
R_h	1.38629	L	1.38629	1e-7	0.000 %	1.14e-3	1.6e-4	3.2e-4	0.023 %
V	8.377580	L^3	8.377812	2.32e-4	0.003 %	2.20e-2	3.11e-3	6.24e-3	0.074 %
C_0	1.25992	L	1.25993	1e-5	0.001 %	1.10e-3	1.6e-4	3.1e-4	0.025 %
Eigenvalue of **S**	0.200000	L^2	0.199307	6.93e-4	0.347 %	4.83e-4	6.8e-5	1.37e-4	0.069 %
Eigenvalue of **S**	0.200000	L^2	0.200566	5.66e-4	0.283 %	4.13e-4	5.8e-5	1.17e-4	0.059 %
Eigenvalue of **S**	1.200000	L^2	1.199827	1.73e-4	0.014 %	2.55e-3	3.60e-4	7.24e-4	0.060 %
$[\sigma]$	4.2072		4.2072	8e-6	0.000 %	1.50e-2	2.1e-3	4.3e-3	0.101 %
$[\eta]$	3.4499		3.4473	2.6e-3	0.075 %	1.23e-2	1.7e-3	3.5e-3	0.101 %
			number of walks = 1e7; number of interior samples = 1e7						
C	1.38629	L	1.38634	5e-5	0.004 %	3.5e-4	5e-5	1.0e-4	0.007 %
Eigenvalue of α	22.658	L^3	22.637	2.1e-2	0.095 %	2.7e-2	4e-3	8e-3	0.034 %
Eigenvalue of α	22.658	L^3	22.690	3.2e-2	0.143 %	2.5e-2	4e-3	7e-3	0.032 %
Eigenvalue of α	60.421	L^3	60.419	2e-3	0.004 %	7.3e-2	1.0e-2	2.1e-2	0.034 %
$\langle\alpha\rangle$	35.246	L^3	35.249	3e-3	0.007 %	2.6e-2	4e-3	7e-3	0.021 %
R_h	1.38629	L	1.38634	5e-5	0.004 %	3.5e-4	5e-5	1.0e-4	0.007 %
V	8.377580	L^3	8.376747	8.33e-4	0.010 %	6.95e-3	9.83e-4	1.98e-3	0.024 %
C_0	1.25992	L	1.25988	4e-5	0.003 %	3.5e-4	5e-5	1e-4	0.008 %
Eigenvalue of **S**	0.200000	L^2	0.199817	1.83e-4	0.091 %	1.67e-4	2.4e-5	4.7e-5	0.024 %
Eigenvalue of **S**	0.200000	L^2	0.200197	1.97e-4	0.098 %	1.65e-4	2.3e-5	4.7e-5	0.023 %
Eigenvalue of **S**	1.200000	L^2	1.200179	1.79e-4	0.015 %	7.55e-4	1.07e-4	2.15e-4	0.018 %
$[\sigma]$	4.2072		4.2079	7e-4	0.017 %	5.2e-3	7e-4	1.5e-3	0.035 %
$[\eta]$	3.4499		3.4479	2.0e-3	0.058 %	4.3e-3	6e-4	1.2e-3	0.035 %

Table 2: **Two touching spheres one of radius 1 and one of radius 1/4**

Property	Ground Truth	Units	Mean	Difference	Rel. Diff.	Std. Dev.	Std. Uncert.	Expand Uncert.	Rel. Exp. Uncert.
			number of walks = 1e6; number of interior samples = 1e6						
C	1.01992	L	1.01996	4e-5	0.004 %	8.7e-4	1.2e-4	2.5e-4	0.024 %
Eigenvalue of α	12.621	L^3	12.570	5.1e-2	0.401 %	5.3e-2	7e-3	1.5e-2	0.119 %
Eigenvalue of α	12.621	L^3	12.666	4.5e-2	0.354 %	4.9e-2	7e-3	1.4e-2	0.111 %
Eigenvalue of α	14.885	L^3	14.888	3e-3	0.020 %	5.9e-2	8e-3	1.7e-2	0.112 %
$\langle\alpha\rangle$	13.376	L^3	13.375	1e-3	0.010 %	3.1e-2	4e-3	9e-3	0.066 %
R_h	1.01992	L	1.01996	4e-5	0.004 %	8.7e-4	1.2e-4	2.5e-4	0.024 %
V	4.254240	L^3	4.251482	2.76e-3	0.065 %	1.03e-2	1.46e-3	2.92e-3	0.069 %
C_0	1.00518	L	1.00496	2.2e-4	0.022 %	8.1e-4	1.1e-4	2.3e-4	0.023 %
Eigenvalue of **S**	0.197115	L^2	0.196618	4.97e-4	0.252 %	4.63e-4	6.5e-5	1.31e-4	0.067 %
Eigenvalue of **S**	0.197115	L^2	0.197705	5.90e-4	0.299 %	3.82e-4	5.4e-5	1.09e-4	0.055 %
Eigenvalue of **S**	0.220784	L^2	0.220712	7.2e-5	0.033 %	7.14e-4	1.01e-4	2.03e-4	0.092 %
$[\sigma]$	3.1442		3.1459	1.7e-3	0.054 %	1.06e-2	1.5e-3	3.0e-3	0.096 %
			number of walks = 1e7; number of interior samples = 1e7						
C	1.01992	L	1.01999	7e-5	0.006 %	3.5e-4	5e-5	1e-4	0.010 %
Eigenvalue of α	12.621	L^3	12.606	1.5e-2	0.119 %	1.6e-2	2e-3	4e-3	0.035 %
Eigenvalue of α	12.621	L^3	12.635	1.4e-2	0.114 %	1.5e-2	2e-3	4e-3	0.035 %
Eigenvalue of α	14.885	L^3	14.882	3e-3	0.017 %	2.2e-2	3e-3	6e-3	0.041 %
$\langle\alpha\rangle$	13.376	L^3	13.375	1e-3	0.010 %	1.1e-2	2e-3	3e-3	0.024 %
R_h	1.01992	L	1.01999	7e-5	0.006 %	3.5e-4	5e-5	1e-4	0.010 %
V	4.254240	L^3	4.253419	8.21e-4	0.019 %	2.93e-3	4.15e-4	8.34e-4	0.020 %
C_0	1.00518	L	1.00512	6e-5	0.006 %	2.3e-4	3e-5	7e-5	0.007 %
Eigenvalue of **S**	0.197115	L^2	0.196960	1.55e-4	0.078 %	1.35e-4	1.9e-5	3.9e-5	0.020 %
Eigenvalue of **S**	0.197115	L^2	0.197299	1.84e-4	0.093 %	1.33e-4	1.9e-5	3.8e-5	0.019 %
Eigenvalue of **S**	0.220784	L^2	0.220771	1.3e-5	0.006 %	2.07e-4	2.9e-5	5.9e-5	0.027 %
$[\sigma]$	3.1442		3.1444	2e-4	0.008 %	3.6e-3	5e-4	1.0e-3	0.032 %

Table 3: **Torus with a minor radius of 1 and major radius of 4**

Property	Ground Truth	Units	Mean	Difference	Rel. Diff.	Std. Dev.	Std. Uncert.	Expand Uncert.	Rel. Exp. Uncert.
			number of walks = 1e6; number of interior samples = 1e6						
C	3.72768	L	3.72754	1.4e-4	0.004 %	3.93e-3	5.6e-4	1.12e-3	0.030 %
Eigenvalue of α	156.53	L^3	156.41	1.2e-1	0.078 %	9.0e-1	1.3e-1	2.6e-1	0.163 %
Eigenvalue of α	972.21	L^3	967.94	4.3e0	0.439 %	3.3e0	4.6e-1	9.3e-1	0.095 %
Eigenvalue of α	972.21	L^3	976.53	4.3e0	0.444 %	4.6e0	6.5e-1	1.3e0	0.133 %
$\langle\alpha\rangle$	700.31	L^3	700.29	2e-2	0.003 %	2.2e0	3.2e-1	6.4e-1	0.091 %
R_h	3.72768	L	3.72754	1.4e-4	0.004 %	3.93e-3	5.6e-4	1.12e-3	0.030 %
V	78.956835	L^3	78.902721	5.41e-2	0.069 %	3.11e-1	4.39e-2	8.83e-2	0.112 %
C_0	2.66134	L	2.66073	6.1e-4	0.023 %	3.49e-3	4.9e-4	9.9e-4	0.037 %
Eigenvalue of $\mathbf{S}$	0.250000	L^2	0.249765	2.35e-4	0.094 %	9.81e-4	1.39e-4	2.79e-4	0.112 %
Eigenvalue of $\mathbf{S}$	8.375000	L^2	8.345469	2.95e-2	0.353 %	1.99e-2	2.82e-3	5.67e-3	0.068 %
Eigenvalue of $\mathbf{S}$	8.375000	L^2	8.405032	3.00e-2	0.359 %	2.02e-2	2.86e-3	5.75e-3	0.069 %
$[\sigma]$	8.8696		8.8755	5.9e-3	0.066 %	4.03e-2	5.7e-3	1.14e-2	0.129 %
			number of walks = 1e7; number of interior samples = 1e7						
C	3.72768	L	3.72771	3e-5	0.001 %	1.11e-3	1.6e-4	3.2e-4	0.008 %
Eigenvalue of α	156.53	L^3	156.48	5e-2	0.035 %	2.9e-1	4e-2	8e-2	0.053 %
Eigenvalue of α	972.21	L^3	970.89	1.3e0	0.136 %	1.3e0	1.9e-1	3.8e-1	0.039 %
Eigenvalue of α	972.21	L^3	973.54	1.3e0	0.137 %	1.1e0	1.6e-1	3.2e-1	0.033 %
$\langle\alpha\rangle$	700.31	L^3	700.30	7e-3	0.001 %	6.8e-1	1e-1	1.9e-1	0.028 %
R_h	3.72768	L	3.72771	3e-5	0.001 %	1.11e-3	1.6e-4	3.2e-4	0.008 %
V	78.956835	L^3	78.913530	4.33e-2	0.055 %	1.19e-1	1.69e-2	3.40e-2	0.043 %
C_0	2.66134	L	2.66085	4.9e-4	0.018 %	1.34e-3	1.9e-4	3.8e-4	0.014 %
Eigenvalue of $\mathbf{S}$	0.250000	L^2	0.249928	7.2e-5	0.029 %	3.87e-4	5.5e-5	1.10e-4	0.044 %
Eigenvalue of $\mathbf{S}$	8.375000	L^2	8.363658	1.13e-2	0.135 %	5.68e-3	8.03e-4	1.61e-3	0.019 %
Eigenvalue of $\mathbf{S}$	8.375000	L^2	8.385071	1.01e-2	0.120 %	6.28e-3	8.88e-4	1.79e-3	0.021 %
$[\sigma]$	8.8696		8.8743	4.7e-3	0.053 %	1.61e-2	2.3e-3	4.6e-3	0.052 %

Table 4: **"Polymer"**

Property	Ground Truth	Units	Mean	Difference	Rel. Diff.	Std. Dev.	Std. Uncert.	Expand Uncert.	Rel. Exp. Uncert.
			number of walks = 1e6; number of interior samples = 1e6						
C	2.15962	L	2.15963	1e-5	0.000%	2.56e-3	3.6e-4	7.3e-4	0.034%
Eigenvalue of α	65.42	L^3	65.39	3e-2	0.049%	4.9e-1	7e-2	1.4e-1	0.211%
Eigenvalue of α	84.09	L^3	83.91	1.8e-1	0.215%	4.9e-1	7e-2	1.4e-1	0.166%
Eigenvalue of α	270.51	L^3	270.51	—	—	1.2e0	1.8e-1	3.5e-1	0.130%
$\langle\alpha\rangle$	140.004	L^3	139.936	6.8e-2	0.049%	4.64e-1	6.6e-2	1.32e-1	0.094%
R_h	2.15962	L	2.15963	1e-5	0.000%	2.56e-3	3.6e-4	7.3e-4	0.034%
V	17.102	L^3	17.095	7e-3	0.043%	7.8e-2	1.1e-2	2.2e-2	0.129%
C_0	1.59828	L	1.59805	2.3e-4	0.014%	2.42e-3	3.4e-4	6.9e-4	0.043%
Eigenvalue of $\mathbf{S}$	0.41542	L^2	0.41513	2.9e-4	0.070%	2.52e-3	3.6e-4	7.2e-4	0.172%
Eigenvalue of $\mathbf{S}$	0.64271	L^2	0.64342	7.1e-4	0.111%	3.80e-3	5.4e-4	1.08e-3	0.168%
Eigenvalue of $\mathbf{S}$	3.6002	L^2	3.6002	—	—	2.02e-2	2.9e-3	5.7e-3	0.159%
$[\sigma]$	8.1864		8.1861	3e-4	0.004%	4.88e-2	6.9e-3	1.39e-2	0.170%
$[\eta]$	6.6823		6.6817	6e-4	0.009%	3.96e-2	5.6e-3	1.13e-2	0.169%
			number of walks = 1e7; number of interior samples = 1e7						
C	2.15962	L	2.15963	1e-5	0.000%	2.56e-3	3.6e-4	7.3e-4	0.034%
Eigenvalue of α	65.42	L^3	65.39	3e-2	0.049%	4.9e-1	7e-2	1.4e-1	0.211%
Eigenvalue of α	84.09	L^3	83.91	1.8e-1	0.215%	4.9e-1	7e-2	1.4e-1	0.166%
Eigenvalue of α	270.51	L^3	270.51	—	—	1.2e0	1.8e-1	3.5e-1	0.130%
$\langle\alpha\rangle$	140.004	L^3	139.991	1.3e-2	0.009%	1.82e-1	2.6e-2	5.2e-2	0.037%
R_h	2.15962	L	2.15963	1e-5	0.000%	2.56e-3	3.6e-4	7.3e-4	0.034%
V	17.102	L^3	17.095	7e-3	0.043%	7.8e-2	1.1e-2	2.2e-2	0.129%
C_0	1.59828	L	1.59805	2.3e-4	0.014%	2.42e-3	3.4e-4	6.9e-4	0.043%
Eigenvalue of $\mathbf{S}$	0.41542	L^2	0.41513	2.9e-4	0.070%	2.52e-3	3.6e-4	7.2e-4	0.172%
Eigenvalue of $\mathbf{S}$	0.64271	L^2	0.64342	7.1e-4	0.111%	3.80e-3	5.4e-4	1.08e-3	0.168%
Eigenvalue of $\mathbf{S}$	3.6002	L^2	3.6002	—	—	2.02e-2	2.9e-3	5.7e-3	0.159%
$[\sigma]$	8.1864		8.1861	3e-4	0.004%	4.88e-2	6.9e-3	1.39e-2	0.170%
$[\eta]$	6.6823		6.6817	6e-4	0.009%	3.96e-2	5.6e-3	1.13e-2	0.169%

Table 5: **Protein**

Property	Ground Truth	Units	Mean	Difference	Rel. Diff.	Std. Dev.	Std. Uncert.	Expand Uncert.	Rel. Exp. Uncert.
			number of walks = 1e6; number of interior samples = 1e6						
C	21.4869	Å	21.4846	2.3e-3	0.011%	2.16e-2	3.1e-3	6.1e-3	0.029%
Eigenvalue of α	96681	Å^3	96641	4e1	0.041%	4.9e2	7e1	1.4e2	0.145%
Eigenvalue of α	100999	Å^3	100952	5e1	0.046%	5.5e2	8e1	1.6e2	0.156%
Eigenvalue of α	184490	Å^3	184460	3e1	0.016%	7.4e2	1.1e2	2.1e2	0.115%
$\langle\alpha\rangle$	127390	Å^3	127351	4e1	0.030%	3.2e2	4e1	9e1	0.071%
R_h	21.4869	Å	21.4846	2.3e-3	0.011%	2.16e-2	3.1e-3	6.1e-3	0.029%
V	32214	Å^3	32222	8e0	0.025%	1.0e2	1e1	3e1	0.092%
C_0	19.7387	Å	19.7403	1.6e-3	0.008%	2.12e-2	3.0e-3	6.0e-3	0.031%
Eigenvalue of **S**	58.975	Å^2	58.942	3.3e-2	0.056%	2.00e-1	2.8e-2	5.7e-2	0.097%
Eigenvalue of **S**	62.500	Å^2	62.502	2e-3	0.003%	2.40e-1	3.4e-2	6.8e-2	0.109%
Eigenvalue of **S**	184.32	Å^2	184.21	1.1e-1	0.060%	6.8e-1	1e-1	1.9e-1	0.105%
$[\sigma]$	3.9545		3.9524	2.1e-3	0.054%	1.71e-2	2.4e-3	4.9e-3	0.123%
$[\eta]$	3.2907		3.2889	1.8e-3	0.055%	1.42e-2	2.0e-3	4.0e-3	0.123%
			number of walks = 1e7; number of interior samples = 1e7						
C	21.4869	Å	21.4856	1.3e-3	0.006%	7.0e-3	1e-3	2.0e-3	0.009%
Eigenvalue of α	96681	Å^3	96659	2e1	0.023%	1.3e2	2e1	4e1	0.038%
Eigenvalue of α	100999	Å^3	101009	1e1	0.010%	1.7e2	2e1	5e1	0.049%
Eigenvalue of α	184490	Å^3	184490	—	—	2.6e2	4e1	7e1	0.040%
$\langle\alpha\rangle$	127390	Å^3	127386	4e0	0.003%	1.2e2	2e1	3e1	0.026%
R_h	21.4869	Å	21.4856	1.3e-3	0.006%	7.0e-3	1e-3	2.0e-3	0.009%
V	32214	Å^3	32218	4e0	0.012%	4e1	5e0	1e1	0.032%
C_0	19.7387	Å	19.7395	8e-4	0.004%	7.3e-3	1.0e-3	2.1e-3	0.011%
Eigenvalue of **S**	58.975	Å^2	58.981	6e-3	0.011%	7.4e-2	1.0e-2	2.1e-2	0.035%
Eigenvalue of **S**	62.500	Å^2	62.512	1.2e-2	0.018%	7.0e-2	1e-2	2.0e-2	0.032%
Eigenvalue of **S**	184.32	Å^2	184.30	2e-2	0.008%	1.8e-1	3e-2	5e-2	0.028%
$[\sigma]$	3.9545		3.9539	6e-4	0.016%	5.8e-3	8e-4	1.7e-3	0.042%
$[\eta]$	3.2907		3.2901	6e-4	0.017%	4.9e-3	7e-4	1.4e-3	0.042%

References

[1] RCSB Protein Data Bank. `www.rcsb.org`.

[2] H. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. Bourne. The Protein Data Bank. *Nucleic Acids Research*, 28:235–242, 2000.

[3] J. F. Douglas and E. J. Garboczi. Intrinsic viscosity and the polarizability of particles having a wide range of shapes. *Advances in Chemical Physics*, 91:85–153, 1995.

[4] J. F. Douglas, H.-X. Zhou, and J. B. Hubbard. Hydrodynamic friction and the capacitance of arbitrarily shaped objects. *Physical Review E*, 49(6):5319–5331, 1994.

[5] J. Hubbard and J. Douglas. Hydrodynamic friction of arbitrarily shaped Brownian particles. *Physical Review E*, 47(5):R2983–R2986, 1993.

[6] D. Juba, W. Keyrouz, M. Mascagni, and M. Brady. Acceleration and parallelization of ZENO/Walk-on-Spheres. *Procedia Computer Science*, 80:269–278, 2016. International Conference on Computational Science 2016, 6–8 June 2016, San Diego, California, USA.

[7] E.-H. Kang, M. Mansfield, and J. Douglas. Numerical path integration technique for the calculation of transport properties of proteins. *Physical Review E*, 69(3):031918, 2004.

[8] M. Mansfield and J. Douglas. Improved path integration method for estimating the intrinsic viscosity of arbitrarily shaped particles. *Physical Review E*, 78(4):046712, 2008.

[9] M. Mansfield, J. Douglas, and E. Garboczi. Intrinsic viscosity and the electrical polarizability of arbitrarily shaped objects. *Physical Review E*, 64(6):061401, 2001.

[10] D. N. Theodorou and U. W. Suter. Shape of unperturbed linear polymers: polypropylene. *Macromolecules*, 18(6):1206–1214, 1985.