Electronic Supplementary Material (ESI) for Physical Chemistry Chemical Physics.

Supporting Information

Reconciling *MA'AT* and Molecular Dynamics Models of Linkage Conformation in Oligosaccharides

Reagan Meredith,[1] Robert J. Woods,[2] Ian Carmichael,[3] and Anthony S. Serianni[1]*

[1]Department of Chemistry and Biochemistry, and [3]Radiation Laboratory, University of Notre Dame, Notre Dame, IN 46556-5670 USA; [2]Complex Carbohydrate Research Center, University of Georgia, Athens, GA 30602

Table of Contents

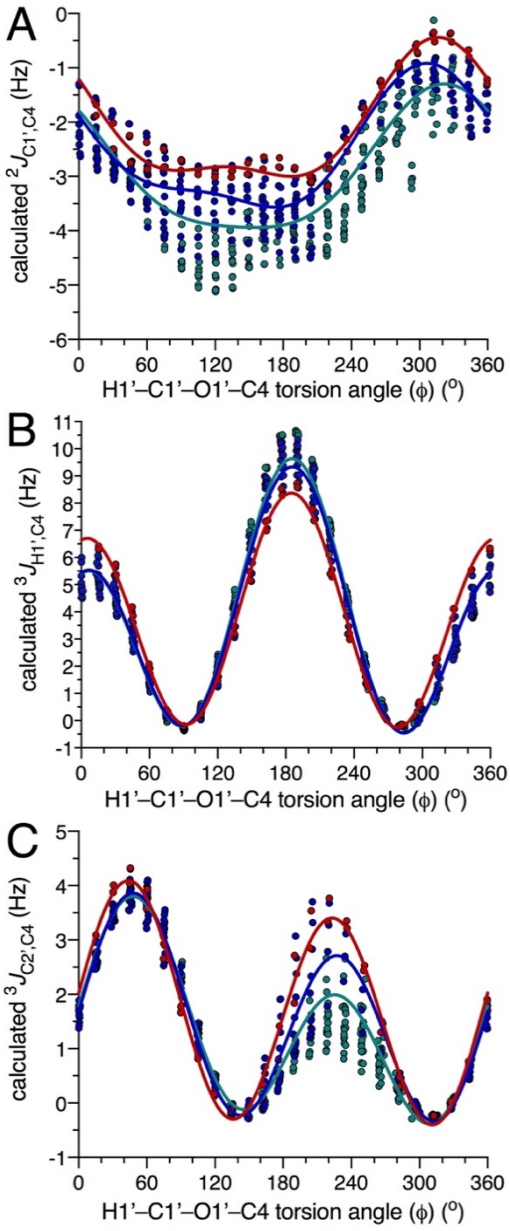


Figure S1. Plots of calculated trans-*O*-glycoside *J*-couplings in **2** as a function of the *phi* (ϕ) torsion angle, defined as H1'–C1'–O1'–C4. (A) $^2J_{C1',C4}$. (B) $^3J_{H1',C4}$. (C) $^3J_{C2',C4}$. Green points, full data set. Blue points, trimmed data set. Red points, ψ-constrained data set. The colored solid lines represent the functions obtained from parameterization of the corresponding data sets.

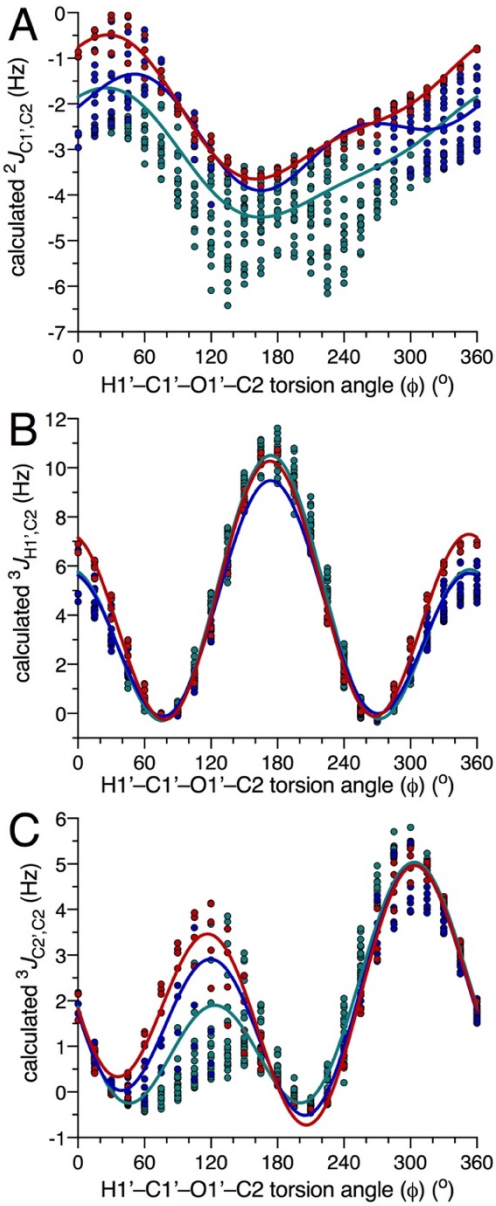


Figure S2. Plots of calculated trans-*O*-glycoside *J*-couplings in **3** as a function of the *phi* (ϕ) torsion angle, defined as H1'–C1'–O1'–C2. (A) $^2J_{C1',C2}$. (B) $^3J_{H1',C2}$. (C) $^3J_{C2',C2}$. Green points, full data set. Blue points, trimmed data set. Red points, ψ-constrained data set. The colored solid lines represent the functions obtained from parameterization of the corresponding data sets.

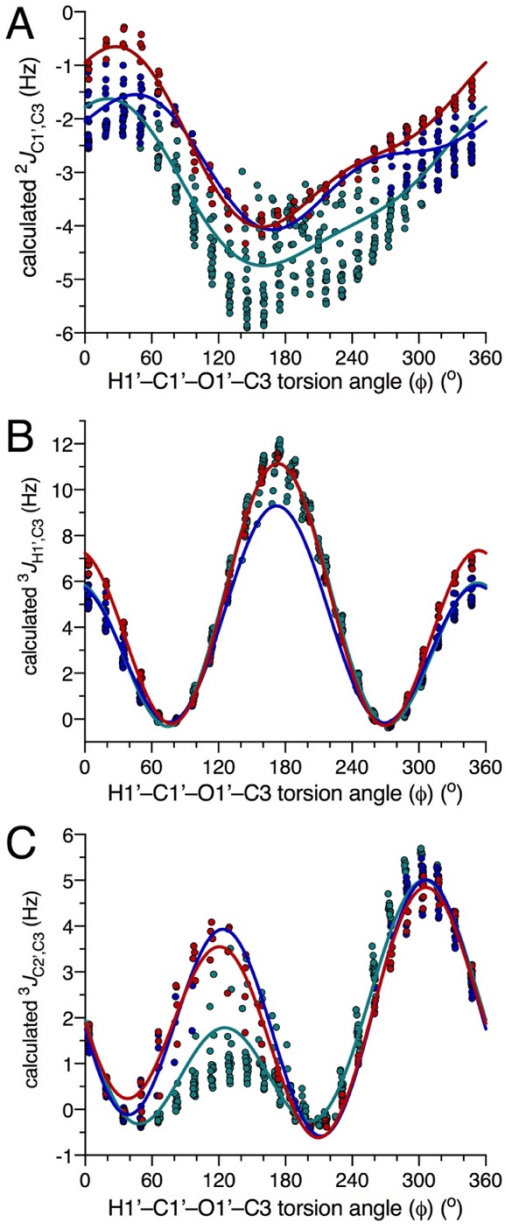


Figure S3. Plots of calculated trans-*O*-glycoside *J*-couplings in **4** as a function of the *phi* (ϕ) torsion angle, defined as H1'–C1'–O1'–C3. (A) $^2J_{C1',C3}$. (B) $^3J_{H1',C3}$. (C) $^3J_{C2',C3}$. Green points, full data set. Blue points, trimmed data set. Red points, ψ-constrained data set. The colored solid lines represent the functions obtained from parameterization of the corresponding data sets.

ϕ-Dependent *J*-Coupling Equations for **2**–**4** Using the Full, Trimmed and Constrained DFT Datasets for Parameterization

βGlc-(1→4)-βGlcOCH$_3$ (**2**) – <u>full</u>

$^2J_{C1',C4}$ (Hz) = $-2.90 + 1.05\cos\phi - 0.79\sin\phi - 0.28\sin 2\phi$
rms 0.61 Hz [1]

$^3J_{C2',C4}$ (Hz) = $1.35 + 0.52\cos\phi + 0.75\sin\phi - 0.10\cos 2\phi + 1.54\sin 2\phi$
rms 0.39 Hz [2]

$^3J_{H1',C4}$ (Hz) = $3.69 - 2.07\cos\phi - 0.10\sin\phi + 3.79\cos 2\phi + 0.82\sin 2\phi$
rms 0.50 Hz [3]

βGlc-(1→4)-βGlcOCH$_3$ (**2**) – <u>trimmed</u>

$^2J_{C1',C4}$ (Hz) = $-2.53 + 0.84\cos\phi - 0.90\sin\phi - 0.19\cos 2\phi - 0.36\sin 2\phi$
rms 0.45 Hz [4]

$^3J_{C2',C4}$ (Hz) = $1.51 + 0.34\cos\phi + 0.46\sin\phi - 0.17\cos 2\phi + 1.76\sin 2\phi$
rms 0.35 Hz [5]

$^3J_{H1',C4}$ (Hz) = $3.62 - 1.90\cos\phi - 0.10\sin\phi + 3.72\cos 2\phi + 0.83\sin 2\phi$
rms 0.54 Hz [6]

αMan-(1→2)-αManOCH$_3$ (**3**) – <u>full</u>

$^2J_{C1',C2}$ (Hz) = $-3.15 + 1.29\cos\phi + 0.21\sin\phi + 0.30\sin 2\phi$
rms 0.88 Hz [7]

$^3J_{C2',C2}$ (Hz) = $1.70 + 0.86\cos\phi - 1.31\sin\phi - 0.70\cos 2\phi - 1.62\sin 2\phi$
rms 0.68 Hz [8]

$^3J_{H1',C2}$ (Hz) = $4.04 - 2.31\cos\phi + 0.21\sin\phi + 4.04\cos 2\phi - 0.90\sin 2\phi$
rms 0.63 Hz [9]

αMan-(1→2)-αManOCH$_3$ (**3**) – <u>trimmed</u>

$^2J_{C1',C2}$ (Hz) = $-2.58 + 0.88\cos\phi + 0.22\sin\phi - 0.36\cos 2\phi + 0.45\sin 2\phi$
rms 0.68 Hz [10]

$^3J_{C2',C2}$ (Hz) = $1.88 + 0.78\cos\phi - 0.73\sin\phi - 0.92\cos 2\phi - 1.84\sin 2\phi$
rms 0.53 Hz [11]

$^3J_{H1',C2}$ (Hz) = $3.82 - 1.88\cos\phi + 0.15\sin\phi + 3.67\cos 2\phi - 0.84\sin 2\phi$
rms 0.68 Hz [12]

αMan-(1→3)-βManOCH$_3$ (**4**) – <u>full</u>

$^2J_{C1',C3}$ (Hz) = $-3.34 + 1.43\cos\phi + 0.7\sin\phi + 0.12\cos 2\phi + 0.33\sin 2\phi$
rms 0.70 Hz [13]

$^3J_{C2',C3}$ (Hz) = $1.64 + 0.90\cos\phi - 1.34\sin\phi - 0.60\cos 2\phi - 1.64\sin 2\phi$
rms 0.59 Hz [14]

$^3J_{H1',C3}$ (Hz) = 4.23 – 2.59 cos ϕ + 0.26 sin ϕ + 4.18 cos 2ϕ – 1.02 sin 2ϕ
rms 0.49 Hz [15]

αMan-(1$\rightarrow$3)-βManOCH$_3$ (**4**) – <u>trimmed</u>
$^2J_{C1',C3}$ (Hz) = –2.75 + 0.99 cos ϕ + 0.7 sin ϕ - 0.28 cos 2ϕ + 0.35 sin 2ϕ
rms 0.50 Hz [16]

$^3J_{C2',C3}$ (Hz) = 2.07 + 0.50 cos ϕ – 0.31 sin ϕ – 0.81 cos 2ϕ – 2.26 sin 2ϕ
rms 0.30 Hz [17]
$^3J_{H1',C3}$ (Hz) = 3.75 – 1.72 cos ϕ + 0.26 sin ϕ + 3.66 cos 2ϕ – 1.02 sin 2ϕ
rms 0.57 Hz [18]

βGlc-(1$\rightarrow$4)-βGlcOCH$_3$ (**2**) – <u>constrained</u>
$^2J_{C1',C4}$ (Hz) = –2.10 + 0.89 cos ϕ – 0.80 sin ϕ – 0.46 sin 2ϕ
rms 0.18 Hz [19]

$^3J_{C2',C4}$ (Hz) = 1.70 + 0.21 cos ϕ + 0.27 sin ϕ + 0.11 cos 2ϕ + 2.04 sin 2ϕ
rms 0.21 Hz [20]

$^3J_{H1',C4}$ (Hz) = 3.68 – 0.83 cos ϕ + 3.80 cos 2ϕ + 0.68 sin 2ϕ
rms 0.21Hz [21]

αMan-(1$\rightarrow$2)-αManOCH$_3$ (**3**) – <u>constrained</u>
$^2J_{C1',C2}$ (Hz) = –2.16 + 1.39 cos ϕ + 0.20 sin ϕ + 0.41 sin 2ϕ
rms 0.20 Hz [22]
$^3J_{C2',C2}$ (Hz) = 2.02 + 0.84 cos ϕ – 0.38 sin ϕ – 1.05 cos 2ϕ – 1.91 sin 2ϕ
rms 0.41 Hz [23]

$^3J_{H1',C2}$ (Hz) = 4.33 – 1.49 cos ϕ + 0.12 sin ϕ + 4.32 cos 2ϕ – 1.09 sin 2ϕ
rms 0.37 Hz [24]

αMan-(1$\rightarrow$3)-βManOCH$_3$ (**4**) – <u>constrained</u>
$^2J_{C1',C3}$ (Hz) = –2.41 + 1.46 cos ϕ + 0.18 sin ϕ + 0.47 sin 2ϕ
rms 0.20 Hz [25]

$^3J_{C2',C3}$ (Hz) = 2.01 + 0.72 cos ϕ – 0.30 sin ϕ – 0.85 cos 2ϕ – 2.00 sin 2ϕ
rms 0.37 Hz [26]

$^3J_{H1',C3}$ (Hz) = 4.55 – 1.88 cos ϕ + 0.26 sin ϕ + 4.57 cos 2ϕ – 1.08 sin 2ϕ
rms 0.27 Hz [27]

Molecular Dynamics Simulations of Disaccharides **2**–**4**

Initial structures of **2**–**4** were built using the Carbohydrate Builder module available at the GLYCAM website (http://www.glycam.org).[1] The GLYCAM06[2] (version j) force field was employed in all simulations. Structures **2**–**4** were solvated with TIP3P[3] water using a 12Å buffer in a cubic box, using the LEaP module in the AMBER14 software package.[4] Energy minimizations for solvated **2**–**4** were performed separately under constant volume (500 steps steepest descent, followed by 24500 steps of conjugate-gradient minimization). Each system was subsequently heated to 300K over a period of 50 ps, followed by equilibration at 300K for a further 0.5 ns using the nPT condition, with the Berendsen thermostat[5] for temperature control. All covalent bonds involving hydrogen atoms were constrained using the SHAKE algorithm,[6] allowing a simulation time step of 2 fs throughout the simulation. After equilibration, production simulations were carried out with the GPU implementation[7] of the PMEMD.MPI module, and trajectory frames were collected every 1 ps for a total of 1 μs. One to four non-bonded interactions were not scaled[8] and a non-bonded cut-off of 8 Å was applied to van der Waals interactions, with long-range electrostatics treated with the particle mesh Ewald approximation. Output from each MD simulation was imported into *Prism* for visualization.

References

(1) Complex Carbohydrate Research Center (CRCC), University of Georgia. http://www.glycam.org

(2) Kirschner, K. N., Yongye, A. B., Tschampel, S. M., González-Outeiriño, J., Daniels, C. R., Foley, B. L., and Woods, R. J. (**2008**) GLYCAM06: A Generalizable Biomolecular Force Field. Carbohydrates, *J. Comput. Chem. 29*, 622–655.

(3) Jorgensen, W. L., Chandrasekhar, J., Madura, J. D., Impey, R. W., and Klein, M. L. (**1983**) Comparison of Simple Potential Functions for Simulating Liquid Water, *J. Chem. Phys. 79*, 926–935.

(4) Case, D. A., Babin, V., Berryman, J. T., Betz, R. M., Cai, Q., Cerutti, D. S., Cheatham, T. E. I., Darden, T. A., Duke, R. E., Gohlke, H. *et al*. (**2014**) AMBER 14, University of California, San Francisco.

(5) Berendsen, H. J. C., Postma, J. P. M., van Gunsteren, W. F., DiNola, A., and Haak, J. R. (**1984**) Molecular Dynamics With Coupling to an External Bath, *J. Chem. Phys. 81*, 3684–3690.

(6) van Gunsteren, W. F., and Berendsen, H. J. C. (**1977**) Algorithms for Macromolecular Dynamics and Constraint Dynamics, *Mol. Phys. 34*, 1311–1327.

(7) Götz, A. W., Williamson, M. J., Xu, D., Poole, D., Le Grand, S., and Walker, R. C. (**2012**) Routine Microsecond Molecular Dynamics Simulations with AMBER on GPUs. 1. Generalized Born, *J. Chem. Theory Comput. 8*, 1542–1555.

(8) Kirschner, K. N., and Woods, R. J. (**2001**) Solvent Interactions Determine Carbohydrate Conformation, *Proc. Natl. Acad. Sci. U. S. A. 98*, 10541–10545.

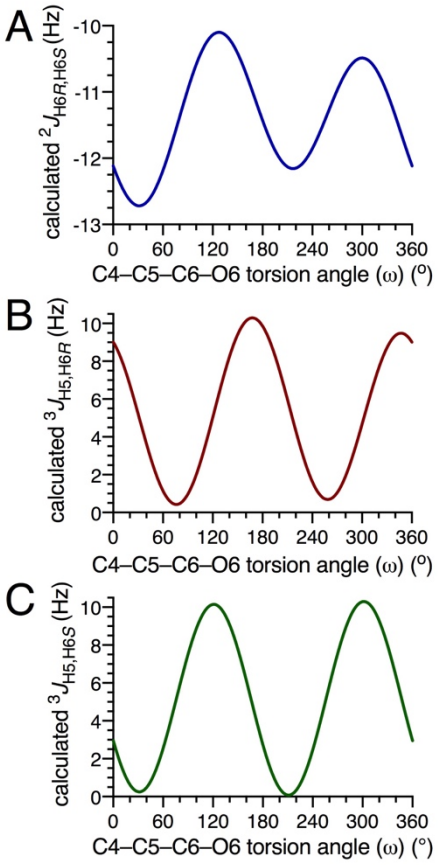
DFT-Derived Hydroxymethyl Equations for
Methyl β-D-Glucopyranoside (**6**)

$^{2}J_{H6R,H6S}$ (Hz) = −11.37 − 0.34 cos θ + 0.01 sin θ − 0.41 cos 2θ − 0.99 sin 2θ
RMSD = 1.29 Hz*

$^{3}J_{H5,H6R}$ (Hz) = 5.22 − 0.42 cos θ − 0.04 sin θ + 4.21 cos 2θ − 2.01 sin 2θ
RMSD = 0.58 Hz

$^{3}J_{H5,H6S}$ (Hz) = 5.19 + 0.11 cos θ − 0.02 sin θ − 2.35 cos 2θ − 4.45 sin 2θ
RMSD = 0.45 Hz

*The large error associated with this equation is caused by the secondary dependence of the $^{2}J_{HH}$ value on C6–O6 bond rotation. When this rotation is highly constrained as in **5**, the error drops to ~0.4 Hz.



Plots of the equations shown above for $^{2}J_{H6R/H6S}$ (A), $^{3}J_{H5,H6R}$ (B), and $^{3}J_{H5,H6S}$ (C).

Brief description of *MA'AT* analysis

The dependence of NMR *J*-couplings on molecular torsion angles is well understood, making it possible to obtain a function of $J(\theta)$ that yields the dependence of the torsion angle on the *J*-coupling. Furthermore, since NMR provides information that is population-averaged, rather than conformations of single molecules, experimental *J*-coupling constants are time-averaged values over the entire population. Therefore, the experimental *J*-coupling (J_{exp}) can be described mathematically by Eq. [1], where $J(\theta)$ equals the *J*-coupling at torsion angle θ and $p(\theta)$ is the population density.

A set of parameterized equations can be obtained from experiment or from density functional theory (DFT) (or other) calculations to treat an ensemble of *J*-couplings that is sensitive to a specific torsion angle, θ. The challenge then becomes how to use this

$$J_{exp} = \int_0^{2\pi} J(\theta)p(\theta)d\theta \qquad \textbf{Eq. [1]}$$

information to determine $p(\theta)$. Several computational methods have been developed to solve this problem. The simplest method assumes that $p(\theta)$ is zero outside of a few discrete values of θ. While the simplicity of this method is appealing, there are several drawbacks. Assuming that torsion angle θ can only populate a few values introduces bias in the treatment. Furthermore, this assumption has no physical basis since it is highly likely that the molecule will experience some degree of libration about the optimal torsion angle or angles and that transition states between the optimal torsion angles will contribute to the experimental *J*-couplings.

Continuous models have been proposed to overcome these problems. To better approximate the true distribution, these models structure $p(\theta)$ as a sum of basis functions. A common example of this approach is the method known as CUPID[1] (*C*ontin*U*ous *P*robab*I*lity *D*istribution of rotamers) that uses a Fourier series to represent $p(\theta)$. Fourier series have the benefit of being well understood functions, and with techniques like fast Fourier transformation (FFT), computations are possible even on limited hardware. However, this approach suffers from several drawbacks. Trigonometric functions are unconventional when describing a probability distribution because their behavior can yield results that would not be expected physically. They do not occur naturally from a large number of trials and they can produce negative values. Most importantly, because there are typically only a few *J*-couplings in an ensemble, the Fourier series must be truncated after a few terms. This truncation produces a function that derives much of its shape from the trigonometric basis instead of from the data.

A logical choice for representing $p(\theta)$ would be a sum of Gaussian distributions. Gaussians arise naturally from a large number of trials and do not require several terms to

produce a series with variable peak widths like trigonometric functions. From a physical standpoint, it is likely that libration about an optimal torsion angle will produce a Gaussian distribution. However, Gaussian distributions are not periodic, that is, they are defined over the entire real line, whereas an angular probability distribution should be defined over a finite interval (e.g., 0 to 2π). Furthermore, evaluating the integral in Eq. [1] numerically for many sample points as part of a nonlinear optimization is computationally intensive, especially as the dimension of the sample space increases (each Gaussian introduces three additional dimensions). The method applied in the *MA'AT* program eliminates these problems by first implementing a mixed wrapped normal distribution and subsequently obtaining a mathematically equivalent expression that can be solved efficiently. The method evaluates Eq. [1] exactly without integration.

Solving Eq. [1] exactly is achieved by representing $p(\theta)$ as a sum of wrapped normal distributions as shown in Eq. [2], where P is the number of peaks, W_p is the weighting parameter per peak, μ_p is the mean position, and σ_p is the circular standard deviation. $J(\theta)$

$$p(\theta) = \int_0^{2\pi} \sum_{p=0}^{P} \sum_{j=-\infty}^{\infty} \frac{w_p}{\sigma_p \sqrt{2\pi}} e^{-\frac{(\theta + 2\pi j - \mu_p)^2}{2\sigma_p^2}} \qquad \textbf{Eq. [2]}$$

is defined as a Karplus-like trigonometric function represented in Eq. [3], where a, c, and s

$$J(\theta) = a + \sum_{n=1}^{N} c_n \cos(n\theta) + \sum_{n=1}^{N} s_n \sin(n\theta) \qquad \textbf{Eq. [3]}$$

are constants for a specific J-coupling and N is the degree of the polynomial (usually 2 or 3). Substituting these two functions into Eq. [1] and evaluating the integral produces the *MA'AT* equation (Eq. [4]). An ensemble of J-couplings gives rise to a series of *MA'AT* equations, making it possible to solve for optimal values W_p, μ_p, and σ_p.

$$J = \sum_{p=0}^{P} w_p \sum_{n=1}^{N} e^{\frac{-(\sigma_p n)^2}{2}} (c_n \cos(\mu_p n) + s_n \sin(\mu_p n)) + k \qquad \textbf{Eq. [4]}$$

The *MA'AT* equation has been encoded into an application that generates population distributions from user supplied Karplus-like equations and experimental data. The application is organized into four separate processes. The first process uploads the data and parameters required for the analysis. The second step generates random population models and then calculates the root mean squared deviation (RMS) of each model. The third step optimizes the models with the lowest RMS from the second step using the Nelder-Mead derivative-free algorithm. The final step organizes the data into numeric

and visual outputs. The standard errors of each parameter are estimated by taking the square roots of the diagonal elements from an inverted hessian matrix.

(1) Dzakula, Z.; Westler, W. M.; Edison, A. S.; Markley, J. L. *J. Am. Chem. Soc*. **1992**, *114*, 6195–6199.