Universal and Accessible Entropy Estimation Using a Compression Algorithm

Ram Avinery,* Micha Kornreich,† and Roy Beck‡
The Raymond and Beverly Sackler School of Physics and Astronomy, Tel Aviv University, Tel Aviv 69978, Israel

(Received 14 May 2018; revised manuscript received 14 November 2018; published 22 October 2019)

Entropy and free-energy estimation are key in thermodynamic characterization of simulated systems ranging from spin models through polymers, colloids, protein structure, and drug design. Current techniques suffer from being model specific, requiring abundant computation resources and simulation at conditions far from the studied realization. Here, we present a universal scheme to calculate entropy using lossless-compression algorithms and validate it on simulated systems of increasing complexity. Our results show accurate entropy values compared to benchmark calculations while being computationally effective. In molecular-dynamics simulations of protein folding, we exhibit unmatched detection capability of the folded states by measuring previously undetectable entropy fluctuations along the simulation timeline. Such entropy evaluation opens a new window onto the dynamics of complex systems and allows efficient free-energy calculations.

DOI: 10.1103/PhysRevLett.123.178102

Utilizing the exponentially growing power of computers enables *in silico* experiments of complex and dynamic systems [1]. In these systems, entropy (S) and enthalpy (H) should be evaluated to appraise the system thermodynamic properties. While enthalpy can be directly calculated from the interaction strength between the system's components, computing the entropy of an equilibrated canonical system essentially requires inferring the probabilities of all relevant microstates (i.e., specific configurations). Consequently, for large systems, contemporary computational capabilities struggle to simulate sufficient microstates for adequate mapping of the free-energy landscape. This fact limits the current ability to estimate thermodynamic properties of interesting systems and phenomena including, e.g., protein folding [1–3].

Present strategies to estimate the entropy from simulations include density- or work-based methods [4]. These methods have been proven useful, though they rely on plentiful computational power, for simulations away from the designated realization [5–7]. Notably, no single method for entropy and free-energy evaluation can be viewed as superior to others, and in many cases, the choice is system dependent [8]. As an alternative path, a reduced phase-space assignment can be used, as previously demonstrated in protein folding simulations [1,2,9]. There, using *a priori* knowledge, such as the experimental native protein structure, can be used to attribute each frame a specific state (e.g., folded or unfolded protein states). Following, a rough estimate of the system's free-energy and thermodynamic properties is then attained using the respective state populations, although entropy values are not directly assigned.

Seminal papers in information theory by Shannon [10] and Kolmogorov [11] introduced measures of uncertainty which are mathematically identical to the statistical-mechanics definition of entropy at the large dataset limit. Lossless compression algorithms are essentially practical implementations attempting to realize Kolmogorov complexity [12,13]. Recognizing these relations has produced novel analytical methods for studying mutual information in sequences of symbols, internet traffic analysis, and redundancy anomaly detections for medicinal signal analysis in electroencephalography, electrocardiography, and more [14–17]. Despite these important links, studies of physical systems using lossless compression are rather sparse. Exceptions include recent studies on thermodynamic phase transitions [18–21]. Additional details on previous studies involving compression algorithms for physical systems are given in the Supplemental Material [22].

Here, we present a framework for accessible and accurate asymptotic entropy (S_A) calculation using a lossless-compression algorithm. Conceptually, the redundancy of information stored in a recorded simulation is tightly related to the entropy of the physical system being simulated. At the foundation of our method, we use a lossless-compression algorithm which is optimized to remove information redundancy by locating repeated patterns within a stream of data. Thus, the ability to compress a digital representation of a physical system is directly related to the entropy of that system [22,27]. We note that other methods exist for the estimation of information entropy in a data stream [27]. Here, we chose to utilize lossless compression, due to its availability and ease of use.

As a proof of concept, we verify our entropy estimation on various model systems where the entropy is analytically calculated and compared. Later, the direct application of asymptotic entropy calculation is demonstrated

on protein folding simulations, where entropy estimation is challenging.

Most adopted lossless-compression implementations derive from schemes introduced by Lempel and Ziv (LZ) [28–30]. LZ algorithms process an input sequence of symbols in a finite alphabet and produce a compressed output sequence by replacing short segments with a reference to a previous instance of the same segment. Fundamentally, the ratio of LZ compressed to input sequence lengths has been proven to converge to Shannon's entropy definition [16,31]. This convergence is guaranteed for an infinite sequence of symbols, produced by an ergodic random source. A sequence of independent microstates sampled from a physical system in equilibrium is in accord with the required random source [22].

One expects LZ schemes to produce an upper bound on physical entropy and approach it asymptotically for large datasets [31]. In practice, our entropy estimation converges to within a few percent from expected values, even for relatively small datasets. This result, in combination with readily available enthalpy values from the simulations, allows us to construct enthalpy-entropy population diagrams for the complex and dynamic simulation of protein folding.

To calculate entropy using a compression-based algorithm, we must quantitatively map the information content (compressed length) to entropy in the proper scale. However, preliminary steps are required to eliminate spurious effects that result from the combination of translating physical systems into 1D datasets, the physical nature of the specific problem, and the algorithm limitations.

Several physical systems are represented using continuous variables. There, each variable requires an enormous alphabet to represent each degree of freedom. This poses a difficulty for compression since the least-significant digits are noisy, and hence incompressible. Therefore, a preprocess is required to reduce the alphabet variability to a coarse-grained representation with n_s values. For additional preprocessing details, see [22].

In the following, we now take the discretized configurations and store them contiguously in a 1D file [22]. We define the original and compressed file sizes, measured in bytes, by $\tilde{C}_d$ and C_d, respectively. To properly evaluate the asymptotic entropy S_A, we generate two additional datasets having the original dataset length. In the first, data over the entire phase space are replaced with a single repeating symbol (e.g., zero). In the second, all the dataset is replaced with random symbols from the alphabet. The resulting two compressed dataset file sizes are denoted by C_0 and C_1, respectively. The ratios $C_0/\tilde{C}_0$ and $C_1/\tilde{C}_1$ converge at the large dataset limit to a value that depends on the size of the alphabet [22].

Since the degenerate and random datasets represent the extreme cases of minimal and maximal entropy, the compressed file size for the simulated state (C_d) lays within these two extremes. Therefore, we define the incompressibility by $\eta = (C_d - C_0)/(C_1 - C_0)$. For physical systems $0 \leq \eta \leq 1$, and converges to a constant in equilibrium with sufficient sampling.

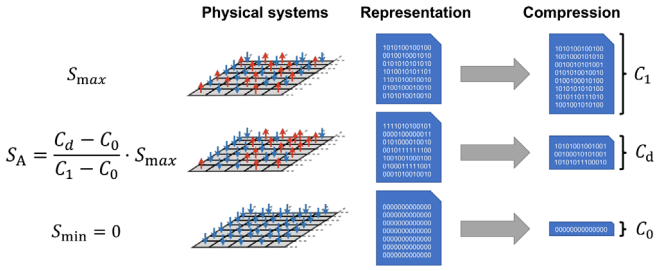

FIG. 1. Schematic asymptotic entropy calculation. Simulations of physical systems are preprocessed and encoded into data files [22]. Entropy is directly calculated from the size of the compressed (C_d) and calibration (C_0, C_1) data, as well as the entropy range (S_{min}, S_{max}).

Finally, mapping η to S_A can be conducted in various ways, for example from prerequisite knowledge on specific entropy values. Alternatively, we recognize that for each of the D degrees of freedom in the system, represented with n_s discrete values, the maximal entropy is given by $k_B \log n_s$, where k_B is the Boltzmann constant. Therefore, as a first order approximation, we linearly map η to entropy, up to an additive constant, by taking $S_A/k_B = \eta D \log n_s$ (Fig. 1) [22]. Below, we demonstrate that this linear mapping asymptotically quantifies the entropy even with finite sampling and far from the large dataset limit (e.g., number of microstates).

We are now ready to evaluate our scheme for several benchmark systems. Herein, we use the Lempel-Ziv-Markov chain-Algorithm compression algorithm, although other algorithms produce qualitatively similar results [19]. We compare S_A to analytical entropy calculation of five different systems [Figs. 2(a)–2(f)]: finite energy levels (ϵ, 2ϵ, 70ϵ, 80ϵ) with an arbitrary energy scale (ϵ) simulated at different temperatures (T), a 2D Ising model on a square lattice, 2D ferromagnetic and antiferromagnetic (frustrated) Ising models on triangular lattices, and an ideal chain fluctuating in 2D with fixed end-to-end distance (R). The Ising models have exchange energy J, the ideal chain is simulated with monomer length b, and all systems are simulated using Monte Carlo algorithms [22]. The results agree well with the theoretical calculation [Figs. 2(a)–2(d)]. In fact, for the Ising model on a square lattice, maximal residues from analytical values are smaller than $0.04k_B$ [Fig. 3(a)]. For the ideal chain simulation, our entropy estimation matches the known entropy dependence of $S(R) - S(0) = -R^2/b^2(N-1)$, where N is the number of monomers, without any fitting parameters [Fig. 3(e)] [22]. Also, our results present a smooth trend and enable us to differentiate S_A for specific heat and critical exponent derivations [Figs. 2(e) and 2(f)].

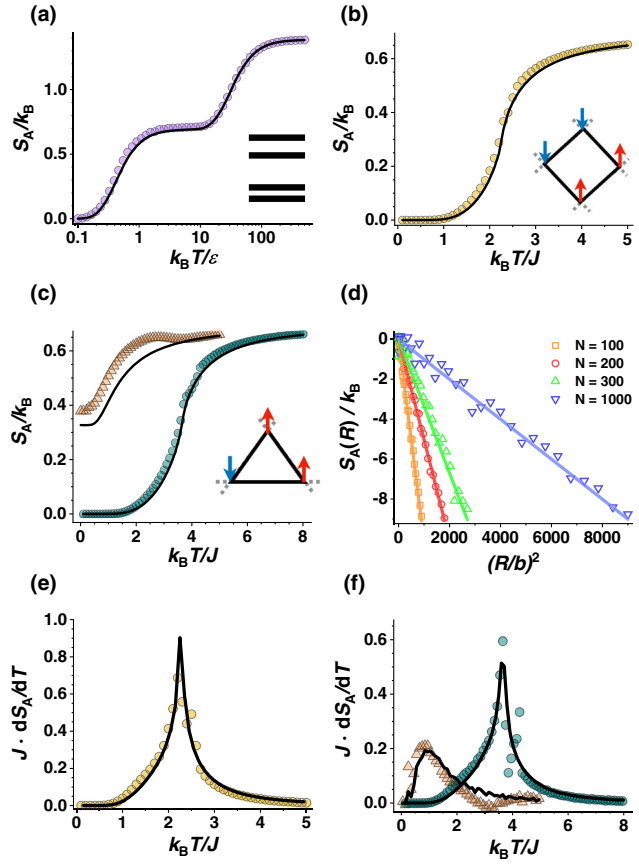

FIG. 2. Validation to asymptotic entropy calculation to Monte Carlo simulation data on benchmark model systems, by comparing analytical entropy calculation (lines) and compression algorithm method (symbols). (a) Discrete energy levels; Ising models on (b) square lattice, and (c) triangular lattice, either with antiferromagnetic ($J < 0$, triangles) or ferromagnetic ($J > 0$, circles); (d) Ideal chains held at varying end-to-end distance (R). (e) and (f) Entropy derivatives of the Ising model simulations [(b) and (c), respectively].

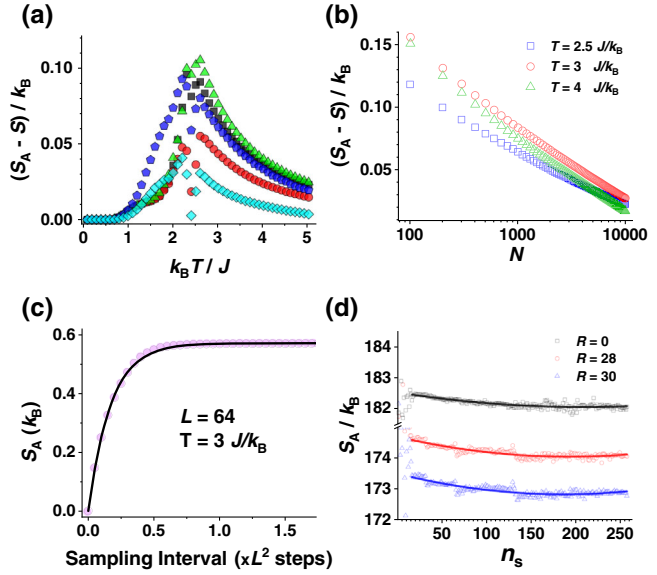

FIG. 3. (a) Residuals from analytical entropy for various 1D reductions of the Ising model on a square lattice: row by row (squares), spiral (triangles), Hilbert (circles), Hilbert with 1 site/byte (pentagons), Hilbert with 3 sites/byte (diamonds). (b) and (c) Asymptotic entropy convergence for the Ising model on a square lattice. (b) Varying sample size (N) and fixed sampling interval ($140L^2$). (c) Varying sampling interval with fixed sample size (10^4), exhibiting exponential convergence (solid line). (d) Effect of coarse-graining level on S_A for the ideal chain. Lines are a guide for the eye.

Since compression algorithms result in an upper bound for the entropy, we can evaluate and optimize different preprocessing protocols [22]. For example, a comparison between different 2D to 1D transformations for the Ising model on a square lattice shows that the Hilbert scan [32] is slightly better than other naive transformations [Fig. 3(a)]. Notably, we can use data compression to evaluate ergodicity and proper sampling intervals [Fig. 3(b)] [22,31]. While the convergence of S_A with an increasing sampling interval is exponential, its convergence with additional sampling is logarithmic [Fig. 3(c)], as expected [33], and will level off as it approaches the actual value, similarly to trends in random data Fig. S1 [22]. Moreover, for as low as 1000 frames S_A estimate is a few percent off the analytically calculated values.

Next, we consider the case of continuous variables which must be coarse grained for further processing and apply it on simulated lattice-free ideal chains [22]. The optimal discretization (n_s) should depend on the correlations in the system and the number of sampled configurations. Furthermore, the choice of a coordinate system representing the degrees of freedom in the system can introduce or eliminate correlations. In our case, the ideal chain simulation is recorded with 64-bit floating numbers for each Cartesian coordinate, but the analysis is applied to the 1D bond-angle representation [22]. We note that standard compression algorithms work best with short range correlations. For physical systems exhibiting long-range correlations, more attention will be required, possibly by transformation to an alternative representation (e.g., Fourier transform).

The ideal chain example validates that optimal coarse graining can be identified using our procedure [Fig. 3(d)]. At the low n_s limit significant information is lost, and the entropy estimate cannot be resolved well. On the other hand, pattern matching by the compression algorithm is hindered by finite sampling and the estimate increases towards the maximal entropy at the high n_s limit. Indeed, S_A evaluation shows a shallow minimum that deepens as the chain is stretched [22].

Encouraged by our results, we test our entropy estimation scheme where free-energy evaluation is a serious concern, namely in protein folding simulation. There, entropy evaluation is currently limited when using the

simulation data alone [34]. Specifically, we quantify entropy for the reversible protein folding of a Villin headpiece C-terminal fragment simulated by molecular dynamics (MD) [2]. The system is sampled at equilibrium and demonstrates short transition times between folded-unfolded states and a long lifetime at each given state [2]. Piana *et al.* [2] calculated the fragment's thermodynamic properties from the population ratio of folded to unfolded states via the transition-based assignment [9] aided by the experimental folded structure [35]. In particular, the difference in entropy between folded-unfolded states (ΔS_f) was estimated from the states' lifetimes (Table S5). Using our compression framework, along with the above-mentioned frame assignments into two ensembles, we attained the backbone's entropy values of the folded-unfolded states [22].

Moreover, the assignment to either of the two states can be done using a sliding entropy estimate from lossless compression, without any *a priori* experimental input. This scheme can potentially detect yet unidentified, competing, low free-energy structures which eluded experimental observation. In Fig. 4(a) we show S_A evaluated for sequences of configurations within a sliding window of length τ_w through the timeline of a simulation. At each time point t, configurations sampled by the simulation between t and $t+\tau_w$ are preprocessed and compressed, to arrive at $S_A(t)$ [22]. We chose the window length $\tau_w = 0.4$ μs as a reasonable compromise between convergence of S_A and time resolution [22]. This choice limits our observations to dynamic processes slower than the chosen time window. Figure 4(a) clearly demonstrates the correspondence between low S_A and Piana's preassigned folded states (shaded areas).

Using these sliding-window S_A values, we can now construct an enthalpy-entropy population diagram [Fig. 4(b)] [22]; a valley between clustered events in the S_A-H plot allows us to assign the folded-unfolded states, without *a priori* experimental knowledge, with 95.3%–96.3% agreement [22] allowing the low free-energy folded structure to be extracted from the simulation and compared to the experimental crystal structure [Fig. 4(c)]. In agreement with Piana's assignments, changes in the ratio between folded-unfolded populations are clearly revealed by S_A distributions, as simulation temperature is varied [Fig. 4(d), Table S5].

Following an assignment to the two folded-unfolded ensembles, we can now use our method to directly estimate the entropic difference between the ensembles. In order to reduce spurious effects resulting from time correlations between neighboring frames, we resampled each ensemble (folded-unfolded) separately [22]. In the following, we optimize the number of coarse-grained dihedral angles, as described above, and estimate each ensemble's entropy value using lossless compression. The difference between the estimated values (ΔS_A) is given in Table S5. We note

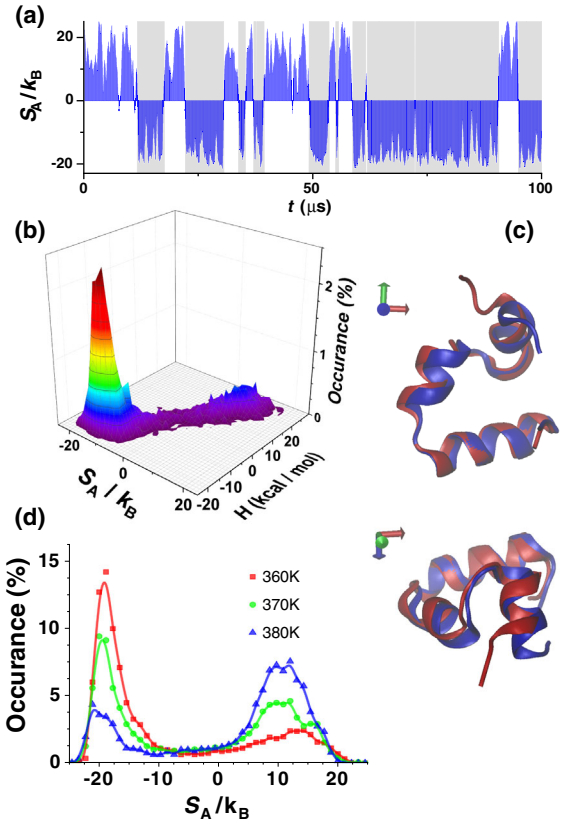

FIG. 4. (a) Representative scan through the Villin headpiece simulation timeline, with mean-subtracted S_A (blue bars), overlaying regions identified as folded using transition-based assignment (gray area). (b) Enthalpy–entropy population diagram at $T = 360$ K. (c) Protein structure collected from randomly sampled low S_A states simulated at 360 K (red) overlaid with the protein fragment (2F4K) crystal structure (blue) [7,35], generated with PyMOL [36]. (d) Distributions of sliding-window S_A, at three simulation temperatures.

that ΔS_A represents only the protein backbone's entropic contribution, as the information is taken solely from the dihedral angles. Further contributions, originating from the solvent, side chains, or other solutes may contribute as well to the overall entropy difference. These contributions will be addressed in future work.

By construction, the successful operation of compression algorithms is derived from identifying domains that repeat within 1D datasets. This is of great convenience for effectively 1D objects such as polymers and proteins. We show that lossless-compression algorithms allow efficient estimation of entropy in a wide variety of physical systems, including protein folding simulations, and without any *a priori* knowledge about specific states. Additionally, our framework can easily assess sufficient sampling, ergodicity, and coarse-graining optimality for many-body simulations. We expect that our methodology will be useful for experimental systems [21] and additional athermal models, where entropy estimation is hard or inaccessible.

Entropy is defined for equilibrated or almost-stationary systems. However, S_A estimates can be useful also away from equilibrium, to detect divergent trends in information content and disorder [21]. Our results demonstrate modern MD simulations have sufficient statistics to allow entropy estimation even for small fragments of simulated trajectories, and that lossless-compression algorithms can be conveniently used for this estimation. The resulting observation of continuous entropy dynamics, including the detection of transient ordered states (i.e., a protein's fold), opens a new avenue in characterizing dynamics of complex systems.

We greatly appreciate fruitful discussions and comments from A. Aharony, D. Andelman, P. Chaikin, H. Diamant, E. Eisenberg, O. Farago, D. Frenkel, M. Goldstein, G. Jacoby, D. Levin, R. Lifshitz, Y. Messica, H. Orland, P. Pincus, Y. Roichman, Y. Shokef, and H. Suchowski. Special thanks for David Shaw laboratory for sharing the MD data. The work is supported by the Israel Science Foundation (550/15) and United States–Israel Binational Science Foundation (201696).

*ramavine@mail.tau.ac.il
†Present address: Department of Physics, New York University, New York, New York 10003, USA.
‡roy@tauex.tau.ac.il

[1] R. O. Dror, R. M. Dirks, J. P. Grossman, H. Xu, and D. E. Shaw, Annu. Rev. Biophys. **41**, 429 (2012).
[2] S. Piana, K. Lindorff-Larsen, and D. E. Shaw, Proc. Natl. Acad. Sci. U.S.A. **109**, 17845 (2012).
[3] S. Piana, J. L. Klepeis, and D. E. Shaw, Curr. Opin. Struct. Biol. **24**, 98 (2014).
[4] D. A. Kofke, Fluid Phase Equilib. **228–229**, 41 (2005).
[5] D. P. Landau and K. Binder, *A Guide to Monte Carlo Simulations in Statistical Physics*, 4th ed. (Cambridge University Press, Cambridge, England, 2014).
[6] D. Frenkel and B. Smit, *Understanding Molecular Simulation: From Algorithms to Applications* (Academic Press, New York, London, 2001), Vol. 1.
[7] J. Kubelka, T. K. Chiu, D. R. Davies, W. A. Eaton, and J. Hofrichter, J. Mol. Biol. **359**, 546 (2006).
[8] N. Hansen and W. F. van Gunsteren, J. Chem. Theory Comput. **10**, 2632 (2014).
[9] N.-V. Buchete and G. Hummer, J. Phys. Chem. B **112**, 6057 (2008).
[10] C. E. Shannon, Bell Syst. Tech. J. **27**, 623 (1948).
[11] A. N. Kolmogorov, Int. J. Comput. Math. **2**, 157 (1968).
[12] T. Downarowicz, *Entropy in Dynamical Systems* (Cambridge University Press, Cambridge, England, 2011), Vol. 18.
[13] W. Krieger, Trans. Am. Math. Soc. **149**, 453 (1970).
[14] T. Henriques, H. Gonçalves, L. Antunes, M. Matias, J. Bernardes, and C. Costa-Santos, J. Eval. Clin. Pract. **19**, 1101 (2013).
[15] M. Aboy, R. Hornero, D. Abásolo, and D. Álvarez, IEEE Trans. Biomed. Eng. **53**, 2282 (2006).
[16] D. Benedetto, E. Caglioti, V. Loreto, and V. Loreto, Phys. Rev. Lett. **88**, 048702 (2002).
[17] J. M. Amigó, J. Szczepański, E. Wajnryb, and M. V. Sanchez-Vives, Neural Comput. **16**, 717 (2004).
[18] O. Melchert and A. K. Hartmann, Phys. Rev. E **91**, 023306 (2015).
[19] E. Vogel, G. Saravia, and L. Cortez, Physica (Amsterdam) **391A**, 1591 (2012).
[20] E. E. Vogel, G. Saravia, and A. J. Ramirez-Pastor, Phys. Rev. E **96**, 062133 (2017).
[21] S. Martiniani, P. M. Chaikin, and D. Levine, Phys. Rev. X **9**, 011031 (2019).
[22] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevLett.123.178102 for additional discussion, method details and results, which includes Refs. [23–26].
[23] U. Wolff, Phys. Rev. Lett. **62**, 361 (1989).
[24] A. G. Schlijper and B. Smit, J. Stat. Phys. **56**, 247 (1989).
[25] L. Onsager, Phys. Rev. **65**, 117 (1944).
[26] G. H. Wannier, Phys. Rev. **79**, 357 (1950).
[27] T. M. Cover, *Elements of Information Theory*, 2nd ed. (Wiley-Interscience, Hoboken, NJ, 2006).
[28] J. Ziv and A. Lempel, IEEE Trans. Inf. Theory **23**, 337 (1977).
[29] J. Ziv and A. Lempel, IEEE Trans. Inf. Theory **24**, 530 (1978).
[30] I. M. Pu, *Fundamental Data Compression* (Butterworth-Heinemann, Oxford, 2005).
[31] A. Lesne, J.-L. Blanc, and L. Pezard, Phys. Rev. E **79**, 046208 (2009).
[32] B. Moon, H. V. Jagadish, C. Faloutsos, and J. H. Saltz, IEEE Trans. Knowl. Data Eng. **13**, 124 (2001).
[33] E. Plotnik, M. J. Weinberger, and J. Ziv, IEEE Trans. Inf. Theory **38**, 66 (1992).
[34] N. Singh and A. Warshel, Proteins **78**, 1724 (2010).
[35] T. K. Chiu, J. Kubelka, R. Herbst-Irmer, W. A. Eaton, J. Hofrichter, and D. R. Davies, Proc. Natl. Acad. Sci. U.S.A. **102**, 7517 (2005).
[36] Schrödinger LLC, The PyMOL Molecular Graphics System, Version 2.1 (2015).