CHAPTER THREE

Quantifying uncertainty in transmembrane stresses and moments in simulation

Samuel L. Foley[a,b,*,1] **and Markus Deserno**[b]
[a]T.C. Jenkins Department of Biophysics, Johns Hopkins University, Baltimore, MD, United States
[b]Department of Physics, Carnegie Mellon University, Pittsburgh, PA, United States
*Corresponding author. e-mail address: sfoley13@jh.edu

Contents

Abstract

The lateral stress profile of a lipid bilayer constitutes a valuable link between molecular simulation and mesoscopic elastic theory. Even though it is frequently calculated in simulations, its statistical precision (or that of observables derived from it) is often left unspecified. This omission can be problematic, as uncertainties are prerequisite to assessing statistical significance. In this chapter, we provide a comprehensive yet accessible overview of the statistical error analysis for the lateral stress profile. We detail two relatively simple but powerful techniques for generating error bars: block-averaging and bootstrapping. Combining these methods allows us to reliably estimate uncertainties, even in the presence of both temporal and spatial correlations,

[1] Currently at Johns Hopkins University.

Methods in Enzymology, Volume 701
ISSN 0076-6879, https://doi.org/10.1016/bs.mie.2024.04.008

which are ubiquitous in simulation data. We illustrate these techniques with simple examples like stress moments, but also more complex observables such as the location of stress profile extrema and the monolayer neutral surface.

We demand rigidly defined areas of doubt and uncertainty!
Douglas Adams, The Hitchhiker's Guide to the Galaxy

1. Arena

1.1 Physical situation

1.1.1 Biomembrane asymmetry

No less than half a century ago, biologists began to realize that the two leaflets of biological lipid membranes differ in lipid composition. Seminal work on human red blood cells (Bretscher, 1972a,b; Verkleij et al., 1973) was soon followed by studies on human platelets (Schick et al., 1976) until asymmetry was also confirmed for nucleated cells (mouse LM fibroblasts) (Sandra & Pagano, 1978). Selective labeling (e.g. with trinitrobenzenesulfonate) or hydrolysis (e.g. with various phospholipases) suggested that some lipids prefer to reside on the outer (exoplasmic) leaflet (e.g., PC (weakly) and sphingomyelin (strongly)), while other lipids strongly prefer the inner (cytosolic) side (e.g., PE and PS). This basic picture has held up remarkably well, but has been significantly refined in recent work (Lorent et al., 2020) that not only resolves lipid head groups but also exquisite details about individual tail length and saturation. Furthermore, comparing predicted surface area profiles of single-pass alpha-helical transmembrane proteins for multiple organisms across the phylogenetic tree suggests that this type of asymmetry is evolutionarily conserved across all of eukarya (Lorent et al., 2020).

While lipid identity is the most conspicuous sign of asymmetry, it is not the only one. In fact, *it cannot be*: once the up-down symmetry of a bilayer is explicitly broken in one particular observable, there is no reason why it should survive in any other one. Indeed, we generally expect asymmetric membranes to have a nonzero spontaneous curvature (meaning, geometric up-down symmetry, aka "flatness", is broken) or a nontrivial electric field (because the dipole potentials of each leaflet do not mirror one another). Recently, Hossein and Deserno argued that the interplay between curvature-elasticity and lipid packing can result in a difference between the individual leaflet tensions,

a condition they refer to as "*differential stress*": $\Delta\Sigma = \Sigma_{\text{upper}} - \Sigma_{\text{lower}} \neq 0$ (Foley et al., 2023; Hossein & Deserno, 2020). Balancing the two membrane torques that result from lipid curvature and unequal leaflet strains shows that (within a simple model) this stress is given by

$$\Delta\Sigma = \frac{\kappa}{z_0}(K_{0\text{b}} - K_0^*), \tag{1}$$

where κ is the bilayer bending modulus (typically tens of $k_\text{B}T$), z_0 is the position of the neutral[1] surface (usually around 1 nm), $K_{0\text{b}}$ is the lipid-derived spontaneous bilayer curvature (which depends on the compositional asymmetry and can easily be tens of inverse microns), and K_0^* is the equilibrium membrane curvature at which bilayer torques are balanced. Putting in numbers, one finds that compositionally asymmetric ($K_{0\text{b}} \neq 0$) but flat ($K_0^* = 0$) membranes (which we know exist, because we can make stable asymmetric giant unilamellar vesicles) must be differentially stressed by several mN/m. This is remarkable, because this stress is at least two orders of magnitude bigger than typical relaxed biomembrane tensions (Morris & Homann, 2001), and in fact surprisingly close to bilayer rupture tension (Evans & Heinrich, 2003). Taken together, these considerations have invigorated investigations into intrinsically stressed lipid bilayers, and more subtle aspects of a membrane's mechanical stress distribution.

1.1.2 The lateral stress profile

Membranes can have any particular tension, or no tension, but as we have just seen, being tensionless does not mean that the underlying material is unstressed: the total bilayer tension $\Sigma = \Sigma_{\text{upper}} + \Sigma_{\text{lower}}$ can vanish while the differential stress $\Delta\Sigma = \Sigma_{\text{upper}} - \Sigma_{\text{lower}}$ can be nonzero—namely, when the two individual leaflet tensions are equal and opposite. This type of subdivision can be microscopically refined by defining a local stress tensor $\boldsymbol{\Sigma}(\boldsymbol{r})$. If the membrane lies in the xy-plane, then $\boldsymbol{\Sigma}$ is diagonal in the xyz-coordinate system and, due to translational symmetry, can only depend on z. Additionally exploiting rotational symmetry around the z-axis, we see that it must have the form $\boldsymbol{\Sigma} = \text{diag}(\Sigma_\parallel(z), \Sigma_\parallel(z), \Sigma_\perp(z))$. Finally, mechanical equilibrium implies $\nabla \cdot \boldsymbol{\Sigma} = 0$, or $\partial_i \Sigma_{ij} = 0$. The only nontrivial

[1] For clarity, we wish to remind the reader that there are two subtly different reference surfaces in use. The neutral surface, which we talk about here, is the surface within a leaflet which, if picked as the reference surface for strain and bending, decouples stretching and bending contributions in an underlying elastic description. The pivotal plane is the surface within a leaflet at which curvature does not create area strain. Notice that the latter does not require any reference to an elastic model to be well-defined, while the former requires by definition such a connection.

condition here is $\partial_z \Sigma_\perp(z) = 0$, which states that the stress along the membrane normal must be constant. However, the tangential component $\Sigma_\parallel(z)$ is unrestricted; in fact, it generally is a strongly varying function of z. Considering a volume preserving deformation that stretches a small rectangular "box" containing the membrane but shrinks it perpendicularly (Rowlinson and Widom, 2003), one realizes that the effective (or "excess") lateral stress is given by the function

$$\Sigma(z) = \Sigma_\parallel(z) - \Sigma_\perp. \tag{2}$$

This so-called "lateral stress profile" plays a major role in membrane mechanics. Roughly assuming that within the membrane headgroup region an oil-water surface tension of $O(30\ \mathrm{mN/m})$ is spread out across $O(1\ \mathrm{nm})$ suggests that we should expect magnitudes for $\Sigma(z)$ in the range of hundreds of bars.[2]

Let us illustrate the usefulness of the stress profile with three examples. First, since $\Sigma(z)$ enters as a contribution to the elastic functional (namely, one that is *linear* in the strain tensor), its moments encode important elastic information; for instance, integrating over the upper ("+") leaflet, one finds (Hamm & Kozlov, 2000; Szleifer, Kramer, Ben-Shaul, Gelbart, & Safran, 1990; Terzi, Ergüder, & Deserno, 2019)

$$\Sigma_{\mathrm{m}+} = \int_0^{\cdots} \mathrm{d}z\, \Sigma(z), \tag{3a}$$

$$-\kappa_{\mathrm{m}+} K_{0\mathrm{m}+} = \int_0^{\cdots} \mathrm{d}z\, \Sigma(z)(z - z_0), \tag{3b}$$

$$\bar{\kappa}_{\mathrm{m}+} + 2\kappa_{\mathrm{m}+,\mathrm{tw}} = \int_0^{\cdots} \mathrm{d}z\, \Sigma(z)(z - z_0)^2. \tag{3c}$$

Here, $\Sigma_{\mathrm{m}+}$ ($\equiv \Sigma_{\mathrm{upper}}$) is the overall tension of the upper leaflet, $\kappa_{\mathrm{m}+}$ and $\bar{\kappa}_{\mathrm{m}+}$ are its bending and Gaussian curvature modulus, respectively, $K_{0\mathrm{m}+}$ is its leaflet spontaneous curvature, $\kappa_{\mathrm{m}+,\mathrm{tw}}$ is its twist modulus, and the integrals extend from the bilayer's midplane all the way out to a distance at which $\Sigma(z)$ has reached its bulk value of 0.

Second, the change of $\Sigma(z)$ under an infinitesimal applied membrane strain gives us a positionally resolved lateral stretching modulus profile (Campelo et al., 2014)

[2] We also alert the reader that some authors prefer to instead discuss a "lateral pressure profile", which differs from the stress profile only in its overall sign.

$$\lambda(z) = A\frac{\partial\Sigma(z)}{\partial A}, \tag{4}$$

which for instance helps us to find the position z_0 of the neutral surface (namely, as the point where the first leaflet moment of $\lambda(z)$ vanishes (Campelo et al., 2014)).

Third, if proteins in biomembranes undergo conformational transitions that are accompanied by a depth-dependent cross-sectional area change $\Delta A(z)$ (think of channels opening or allosteric changes of a receptor), such transitions involve the mechanical work

$$\Delta W = \int \mathrm{d}z\ \Delta A(z)\Sigma(z), \tag{5}$$

which can be surprisingly large (Ollila et al., 2011).

Before moving on, it is worthwhile to spell out one caveat: the stress tensor, and with it the stress profile, can in principle[3] be defined at the molecular level, rendering it accessible to Statistical Mechanics. Objects such as surface tension or a pivotal plane distance, in turn, are observables that can be measured in experiment. Eqs. (3) link these microscopic and macroscopic worlds, but where do they come from? It turns out that many derivations of these equations are piped through an intermediate theory: continuum elasticity. This should neither be necessary, nor does it bode well for the actually derived connections, since it makes them dependent on the validity of a mesoscopic middle man. In fact, Eqs. (3a) and (3b) can be explained without relying on it (they just express stress- and torque balance), while the same does not appear to be true for Eq. (3c). Independent measurements of the Gaussian curvature modulus based purely on macroscopic reasoning indeed are in conflict with this third equation (Hu et al., 2012,2013), and the extra twist correction[4] appearing in it does not resolve the discrepancy in all cases. A similar criticism may be applied to Eq. (4): this stretching modulus profile describes the local elasticity of a continuum model. Our primary purpose for introducing it here is to serve as an illustrative example of a complicated quantity derived from

[3] … up to some serious nuisances pertaining to a possible contour dependence. The reader will find more details in Schofield & Henderson (1982), but these affect our main point only peripherally.

[4] Cognoscenti of $\bar{\kappa}$-lore will notice that the "$+2\kappa_{\mathrm{m}+,\mathrm{tw}}$" term is not usually found in the literature. It was derived in Hamm & Kozlov (2000), where its presence was connected to a specific interpretation of "lateral fluidity", but subsequently buried (or forgotten). It was exhumed in Terzi, Ergüder, & Deserno, 2019, where its inclusion was shown to yield more plausible Gaussian moduli from atomistic simulations. But the very fact that its existence depends on some subtle decisions being made at the mesoscopic level illustrate the original point: this ambiguity is an artifact of the intermediate theory.

the lateral stress profile; its validity and/or applicability is not our chief concern.[5]

These admonitions notwithstanding, the examples above still illustrate why the lateral stress profile is a valuable observable. It is hence rather frustrating that no method seems to exist to determine it in experiment (reproducible well-calibrated local stress measurements at Angström resolution are apparently difficult). However, simulations do not suffer from such limitations, which means we can use them to obtain genuinely new information to which experiment is (so far) blind. This is likely especially rewarding for asymmetric membranes, whose lateral stress profile is no longer just mirrored at the bilayer midplane and hence contains positionally resolved information about the underlying differential stress.

1.2 Challenges

One surprising challenge for the lateral stress profile is that we are not fully sure how to actually define and calculate it. Pointwise stresses are subtle (Rowlinson & Widom, 2003; Schofield & Henderson, 1982)! A first practical and widely used method has been proposed by Goetz and Lipowsky (1998), but subsequent work has pointed out that decomposing multibody forces into pair forces is not unique (Admal & Tadmor, 2010; Schofield & Henderson, 1982), leading to different methods for calculating the stress profile that can yield rather different results. Insisting on further requirements (e.g., enforcing the strong[6] law of action and reaction, and hence a symmetric stress tensor (Vanegas et al., 2014)) may still leave legitimate alternative choices that lead to different stress profiles (Nakagawa & Noguchi, 2016). This is a bit embarrassing, and to the best of our knowledge most practitioners just agree to not talk about it. Since the differences between routines that enforce a symmetric stress tensor appear not to be very noticeable (Nakagawa & Noguchi, 2016; Vanegas et al., 2014), the problem is maybe also not that pressing, and so we will follow the time honored path of dodging the issue.

Another item to pay attention to is that the lateral stress profile $\Sigma(z)$ depends on z, and so it is essential that subsequent membrane snapshots

[5] However, recent work has found that measurements based only on macroscopic elastic considerations are in good agreement with the neutral surface values derived via Eq. (4) (Foley & Deserno, 2024).

[6] Newton's third law states that the forces which two particles exert onto one another are equal in magnitude and opposite in direction. In its strong form the law additionally requires them to act along the line between the two particles. An example where the strong form fails is the Lorentz force between two moving charged particles; albeit, here we should really also consider the momentum stored in the electromagnetic field.

from an MD trajectory are positionally aligned, for instance by fixing the z-position of the center of mass to the middle of the simulation box. But even if the overall position is locked in place, membranes undulate, and this smears out $\Sigma(z)$ *via* some kernel $w(z)$ when statistics are accumulated laterally. Interestingly, as long as $w(z)$ is symmetric, the zeroth, first, and second bilayer moments of $\Sigma(z)$ do not change upon smearing (Hu et al., 2013). The same is unfortunately not true for monolayer moments, such as those in Eq. (3); here the usual strategy is to make sure the blurring is small enough, for example compared to the value z_0 entering the moments. Insisting that the undulation-induced rms-width of $w(z)$ is only fz_0, with f "small enough", restricts one to membranes with lipid number (Hu et al., 2013)

$$N_{\text{lipids}} \lesssim 32\pi^3 f^2 \times \frac{\kappa}{k_{\text{B}}T} \times \frac{z_0^2}{a_\ell}, \tag{6}$$

where a_ℓ is the typical area of a lipid.

In this chapter we will look into the issue that the inevitable spatiotemporal correlations inherent in any MD trajectory need proper accounting when quantifying the uncertainties in $\Sigma(z)$. The temporal correlations could be handled by several different techniques, data blocking being a very popular and straightforward option (Flyvbjerg & Petersen, 1989). Unfortunately this simple trick does not work for the spatial correlations (say, between close stress-bins). Moreover, interesting observables are often complicated functions of the underlying data. We might need to calculate a moment from $\Sigma(z)$, or consider the case of backing out a neutral surface position by finding the location where the first moment of $\lambda(z)$ vanishes, which in itself is a complicated functional of $\Sigma(z)$ (see Eq. (4)). Such convoluted paths make standard linear error propagation à la $(\nabla f|_\mu)^\top \mathbf{C}\, \nabla f|_\mu$ difficult to actually calculate, even if we have a good handle on the correlations, because the calculation of the gradients can turn into a real pain.

2. Accounting for spatiotemporal correlations

Calculating averages on correlated data makes the uncertainty quantification tricky. If multiple measurements of a random variable X are correlated, then their covariance matrix is not diagonal, and hence the variance of the standard estimator of the mean contains off-diagonal contributions (see Sec. 3.1 below). Since these are typically positive ("nearby" measurements are commonly more similar to each other rather than less),

we would expect this to result in an underestimation of the error in our measurement of X if we were to go by the naïve "variance of individual measurements divided by number of measurements" recipe. As we will see below, this is often true—but not always.

2.1 Big idea

Measurements of X can be "nearby" in time or space (or both). Temporal correlations invariably exist downstream of MD trajectories, since subsequent simulation frames inherit all information from their past. One might elect to wait with a new measurement until all these correlations have decayed, but this is almost always a huge waste of resources and time, because (i) correlations might be fairly long and (ii) the fact that measurements are not completely independent does not mean that the new correlated measurement has no valuable new information. The important caveat is merely to make sure how much is new when estimating the uncertainty.

2.2 Main tools

Solving the correlation problem is—at least in theory—straightforward, as long as one actually knows the correlations. In practice, this is difficult because correlations can be expensive to sample, and finding a consistent unbiased estimator is subtle.

Thankfully, for temporal correlations that arise in a stationary stochastic process, a method exists to account for them without having to directly measure them: data blocking (Flyvbjerg & Petersen, 1989). This elegant technique is easy to implement at very little computational overhead, and as a bonus gives information on correlation times that helps answering the crucial question "have we sampled enough?"

For spatial correlations (e.g., nearby bins of positional histograms) blocking is not an option. In that case, directly accounting for the correlations is necessary, irrespective of whether we need to do so in some generalized linear least square fitting context (Paige, 1979), or whether we wish to parametrically bootstrap the data (Efron & Tibshirani, 1994). In both cases the key trick relies on the fact that we can create a correspondence between a zero-mean noise vector $\boldsymbol{c}$ having a covariance matrix $\mathbf{C} = \langle \boldsymbol{c}\boldsymbol{c}^{\top} \rangle$ and an uncorrelated zero-mean-unit-variance noise vector $\boldsymbol{g}$: loosely speaking, correlations are "quadratic objects", and to "undo" them, we need a linear coordinate transformation that is essentially the "square root" of $\mathbf{C}$. The path to glory, hence, is Cholesky decomposition, $\mathbf{C} = \mathbf{L}\mathbf{L}^{\top}$, such that $\boldsymbol{c} = \mathbf{L}\boldsymbol{g}$.

3. Implementation

3.1 Time correlation and blocking

As a molecular simulation runs, coordinate snapshots (along with other observables of interest) are written out, typically at some fixed frequency with time interval Δt. After an initial equilibration period has passed (whose length depends on composition, size, temperature, etc.), it is assumed that the system samples the desired equilibrium ensemble. Measurements before this point are not included in a subsequent analysis, and from this point forward, the output sequence of observables should constitute a stationary stochastic process; that is, one in which the joint probability distribution is invariant under time translations.

Given a time series for a particular observable, here denoted x, a typical goal is to determine the mean value $\mu_x = \langle x \rangle$ in the given simulation ensemble. The standard estimator of the mean of the sequence of N measurements $\{x_1, x_2, \ldots, x_N\}$ is given by $\hat{\mu}_x = \frac{1}{N}\sum_{i=1}^{N} x_i$. The immediate next question is: how accurate is that estimate? The naïve rule of thumb is to calculate the standard error of our sample mean as $\hat{\sigma}_x/\sqrt{N}$, where $\hat{\sigma}_x^2 = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \hat{\mu}_x)^2$ is the (unbiased) sample variance. This is the typical method of producing "error bars" for values obtained through repeated measurements; the result is then reported as $\langle x \rangle = \hat{\mu}_x \pm \hat{\sigma}_x/\sqrt{N}$.

This procedure is of course correct if the individual measurements x_i constitute independent, identically-distributed (i.i.d.) random variables (RVs). In fact, it suffices that the x_i are uncorrelated RVs, but even that is not the case in most instances for time series measurements. If the covariance of x_i and x_j is $\mathrm{C}_{ij} = \mathrm{Cov}(x_i, x_j)$, then a direct calculation shows that

$$\mathrm{Var}(\hat{\mu}_x) = \langle (\hat{\mu}_x - \langle \hat{\mu}_x \rangle)^2 \rangle = \frac{\sigma_x^2}{N} + \frac{1}{N^2}\sum\nolimits_{i \neq j} \mathrm{C}_{ij}, \tag{7}$$

where $\sigma_x^2 = \mathrm{Cov}(x_i, x_i)$ is the "true" variance of x. The naïve result is obviously recovered in the case $\mathrm{C}_{ij} = \sigma_x^2 \delta_{ij}$, that is, $\mathrm{C}_{ij} = 0$ for $i \neq j$. For the case of interest here, namely that of a stationary process, the covariance can only depend on the separation of two measurements in time, indexed by $\tau \equiv |i - j|$ (i.e., $\mathbf{C}$ is a symmetric Toeplitz matrix). In this case,

$$\mathrm{Var}(\hat{\mu}_x) = \frac{\sigma_x^2}{N}\left[1 + 2\sum_{\tau=1}^{N-1} c_\tau + O\left(\frac{1}{N}\right)\right]. \tag{8}$$

Thus, the variance of our estimator is not simply σ_x^2/N; rather this result is scaled by a factor involving a sum over the sequence's autocorrelation function $c_\tau \equiv (\langle x_{i+\tau}x_i\rangle - \langle x\rangle^2)/\sigma_x^2$.[7] Assuming that we have a sufficiently long observation sequence that we may take the limit $N \to \infty$, one can ignore the $O(N^{-1})$ term in Eq. (8) and use the leading order correction to estimate $\mathrm{Var}(\hat{\mu}_x)$.[8]

While we could in principle move on after deriving Eq. (8), it is a bit cumbersome in that it requires the calculation of the autocorrelation function in order to evaluate the correction term. As it happens, there is a simpler way for estimating $\mathrm{Var}(\hat{\mu}_x)$ which can be automated and provides directly visualizable feedback for determining whether one has sufficient data. The technique is known as blocking (Flyvbjerg & Petersen, 1989), and it is an iterative process in which consecutive measurements in the time series are grouped (into "blocks") and averaged, producing a sequence with $N_{k+1} < N_k$ entries, as shown in Fig. 1.

In this example, we use blocks of size $b = 2$, such that each iteration of blocking halves the length of the sequence; $N_{k+1} = N_k/2$ (in general, $N_{k+1} = N_k/b$). Taking $b = 2$ for each blocking step is a common but not necessary step. Other factors are possible, which even permits a "denser" coverage of the blocking axis by "interweaving" the blocking step ladders. At any rate, it is wise to pick the total number N of observations to be a number with many small prime factors. We will briefly comment on some further refinements in Sec. 4.2.2.

This block-averaging is repeated, and at each step k one computes the standard error using the naïve formula $\hat{\sigma}_k/\sqrt{N_k}$. If the data are uncorrelated, this process leaves the mean and its standard error unchanged. However, if the data *are* correlated, then block averaging results in a sequence of values that are effectively less correlated, and the computed standard error increases. Once sufficiently many blocking steps have been carried out, the resulting sequence is effectively uncorrelated, and subsequent blocking steps result in no change to $\hat{\sigma}_k/\sqrt{N_k}$.[9] It is this "plateau value" that

[7] Observe that c_τ is a correlation coefficient and hence lies between −1 and +1. Furthermore, the sum $\sum_{\tau=1}^{N-1} c_\tau$ is a common measure for characterizing the effective correlation time of a process. For instance, in the special case $c_\tau = e^{-\tau/T}$ (i.e., an exponentially decaying correlation function with a single characteristic time T), we recover $\sum_{\tau=1}^{N-1} c_\tau = T + O(1)$ for $N >> T$.

[8] One may worry that at this point Eq. (8) still relies on the "true" parameters σ_x and c_τ, not our actually available sample estimators of these quantities. One can show that when swapping to the estimators, the only new terms that arise contribute to the $O(1/N)$ and higher terms which we already neglect.

[9] Flyvbjerg & Petersen (1989) present this in the framework of the renormalization group, whereby repeated iteration of the blocking transformation takes us to a fixed point of the RG flow.

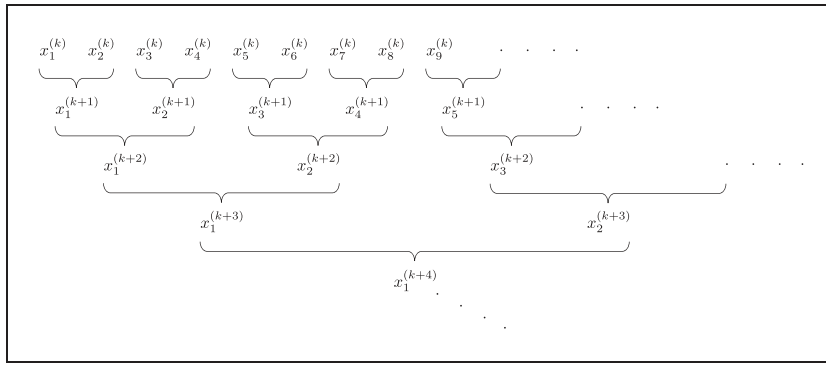

Fig. 1 The iterative block-averaging procedure. The k^{th}-order sequence is computed by averaging adjacent data in sequence $k-1$, demonstrated here for constant block size $b=2$.

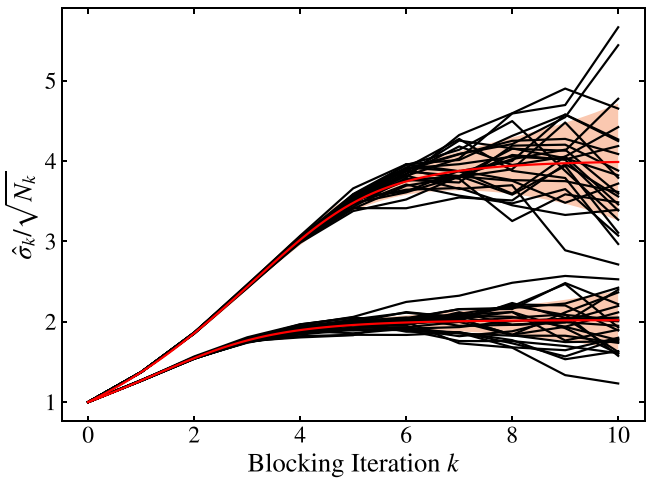

Fig. 2 Blocking curves for exponentially-correlated Gaussian processes of length $N=2^{14}$. The standard error calculated at each iteration is normalized by the original naïve value $\hat{\sigma}/\sqrt{N}$. Correlation times are $T=2$ (lower) and $T=8$ (upper). Blocking was carried out on 25 separate realizations for each correlation time. Red curves show the theoretical blocking curve predicted from Eq. (9), along with the relative error estimate (pale red shading) from Eq. (11).

represents the true uncertainty of the mean, hence the procedure is ostensibly straightforward: block, and find the plateau.

As a concrete example, consider an exponentially correlated Gaussian process. Its autocorrelation function is $c_\tau = \mathrm{e}^{-\tau/T}$, with T being the correlation time.[10] Fig. 2 shows the result of the blocking procedure carried out on several realizations of such Gaussian processes for two different correlation times, $T = 2$ and $T = 8$. The naïvely calculated standard error is plotted for each blocking iteration, normalized by the initial estimate $\hat{\sigma}/\sqrt{N}$ (this way, our plot simply tells us how much our naïve error needs to be scaled up). As can be seen, the shorter correlation time results in a blocking curve which rises less steeply and "saturates" sooner than the cases for the longer correlation time.

As it happens, for the case of exponential time correlations the sum in Eq. (8) is a simple geometric series, easily evaluated in closed form. This allows one to derive a theoretical result for the form of a blocking curve of an exponentially correlated time series (Ergüder & Deserno, 2021; Janke, 2002),[11]

$$\frac{\sigma_k/\sqrt{N_k}}{\sigma/\sqrt{N}} = \sqrt{\frac{1+c}{1-c} - \frac{2c}{B(k)} \times \frac{1-c^{B(k)}}{(1-c)^2}}\,. \tag{9}$$

Here, $c \equiv c_1 = \mathrm{e}^{-1/T}$ and $B(k)$ is the accumulated block size after k blocking steps. If each step blocks b measurements together (e.g., $b = 2$, as in Fig. 1) then $B(k) = b^k$. Taking the limit of large times (i.e., many blocks), we can extract the plateau:

$$\lim_{k\to\infty} \frac{\sigma_k/\sqrt{N_k}}{\sigma/\sqrt{N}} = \sqrt{\frac{1+c}{1-c}} = \sqrt{\coth\frac{1}{2T}} = \begin{cases} 1, & T \ll 1 \\ \sqrt{2T}, & T \gg 1 \end{cases}, \tag{10}$$

which agrees with the well-known result that for large times the true uncertainty is larger than the naïve one by a factor of $\sqrt{2T}$.[12]

[10] Here's how to create such a process—something that can be handy for testing purposes: given a sequence g_n of i.i.d. Gaussian random numbers with zero mean and unit variance, the new sequence $r_0 = g_0$ and $r_{n+1} = cr_n + \sqrt{1-c^2}\,g_{n+1}$ with $c = \mathrm{e}^{-1/T}$ is a sequence of Gaussian random numbers with zero mean, unit variance, and correlation time T.

[11] Similar ideas lead to an expression published in the Appendix of Hess (2002), on which the routine `g_analyze` in GROMACS is based.

[12] A note to keep in mind for all of this is that the correlation "time" T that we have been discussing is in units of "steps", assumed to have duration Δt; the physical correlation time is $T\Delta t$.

Since the blocked data asymptotically approach independent Gaussian variables, we can also estimate their relative uncertainty as (Flyvbjerg & Petersen, 1989)

$$\sigma\left(\hat{\sigma}_k/\sqrt{N_k}\right) = \frac{\hat{\sigma}_k/\sqrt{N_k}}{\sqrt{2(N_k-1)}}, \tag{11}$$

where N_k is the number of data points left after k blocking steps. Strictly, this only holds once we have reached the plateau, since only then do we get independent Gaussian variables; Eq. (11) still offers a lower bound, though.

The theoretical prediction for the entire blocking curve is included in Fig. 2 as the two red curves, one for each correlation time, together with "1-σ error bands" calculated via Eq. (11). The agreement at early blocking iterations k is extremely good, and as the blocking process progresses to fewer and fewer remaining data points, the individual blocking realizations scatter more widely around the asymptotic prediction. Observe also that for $T = 2$ and $T = 8$ the asymptotic plateau is approximately 2 and 4, respectively, in line with the expectation from Eq. (10).

The quick-and-dirty way to evaluate the outcome of a blocking process is to "eyeball" the plateau; but since we have a prediction for the entire curve, a much better procedure suggests itself: fit the blocking curve to the theory from Eq. (9) to determine c. Not only does this take advantage of the information in the more precisely-known pre-plateau region; it is also a single-parameter fit that does double-duty: we get the proper asymptotic error scaling $\sqrt{(1+c)/(1-c)}$ *and* the correlation time $T = -1/\ln c$.

3.2 Spatial correlation and bootstrapping

Now, we will shift our mindset away from stationary time series correlations and instead consider local correlations among M RVs x_i, which we collect together as a vector $\boldsymbol{x}$ of dimension M. We will think of the component RVs of $\boldsymbol{x}$ as corresponding to measured quantities which are spatially near one another, with our motivating example being the membrane lateral stress profile $\Sigma(z)$ measured at discrete locations z_i. Quantities which for the continuous stress profile would be functionals (e.g., moments) are instead simply functions of a vector quantity when considered at discrete points. For simplicity, we will consider for now the spatial correlations in isolation, and thereby assume that our data comprises N i.i.d. measurements of $\boldsymbol{x}$. We shall denote the mean of $\boldsymbol{x}$ in the ensemble of interest by $\boldsymbol{\mu} = \langle \boldsymbol{x} \rangle$, and its covariance matrix $\mathbf{C} = \langle (\boldsymbol{x}-\boldsymbol{\mu})(\boldsymbol{x}-\boldsymbol{\mu})^{\mathsf{T}} \rangle$.

Our quest in this section is to investigate the error on quantities $f = f(\boldsymbol{\mu})$ obtained as a function of the mean of our correlated random vector (read: instantaneous stress profile) $\boldsymbol{x}$. We should note up front that for simple linear function(al)s $f(\boldsymbol{\mu})$, one can simply calculate $f(\boldsymbol{x})$ for each individual measurement and determine the summary statistics from the resulting distribution (or apply the blocking procedure from the previous section to account for time correlation). However, more complicated observables, such as those which are highly nonlinear and perhaps not even well-defined for instantaneous measurements of $\boldsymbol{x}$, require more care in their analysis. A good example of such a functional is the location of stress extrema, and we will examine this case in Sec. 4.1.

Blocking is no longer an option, as the vector $\boldsymbol{x}$ is not a stationary stochastic process. Instead, we may turn to our sample estimator of the covariance matrix,

$$\hat{\mathrm{C}}_{ij} = \frac{1}{N-1} \sum_{n=1}^{N} (x_{n,i} - \hat{\mu}_i)(x_{n,j} - \hat{\mu}_j). \tag{12}$$

Here $x_{n,i}$ refers to measurement n, vector component i. Clearly, the diagonal elements $\hat{\mathrm{C}}_{ii} = \frac{1}{N-1} \sum (x_{n,i} - \hat{\mu}_i)^2 = \hat{\sigma}_i^2$ are the usual unbiased sample estimators of the variance of each component x_i. In our case, where we have assumed that all of our measurements are i.i.d., a direct calculation shows that the covariance of the estimator of the mean (the generalization of the standard error of the mean) is $\mathbf{C}_{\mu} \equiv \langle (\hat{\boldsymbol{\mu}} - \langle \hat{\boldsymbol{\mu}} \rangle)(\hat{\boldsymbol{\mu}} - \langle \hat{\boldsymbol{\mu}} \rangle)^{\top} \rangle = \mathbf{C}/N$, in simple analogy with the single-variable result of $\sigma_{\mu}^2 = \sigma^2/N$.

In the simplest cases, all that remains is to apply the complete version of the ordinary "error propagation formula":

$$\sigma_f^2 \approx (\nabla f|_{\langle \mathbf{x} \rangle})^{\top} \mathbf{C} \, (\nabla f|_{\langle \mathbf{x} \rangle}) = \sum_{i,j} \left. \frac{\partial f}{\partial x_i} \right|_{\langle x_i \rangle} \mathrm{C}_{ij} \left. \frac{\partial f}{\partial x_j} \right|_{\langle x_j \rangle}, \tag{13}$$

which in the *uncorrelated* case $\mathrm{C}_{ij} = \sigma_i^2 \delta_{ij}$ simplifies to the well-known weighted-sum-of-variances expressions.

Often enough this result is sufficient, but a few comments are in order. First, it should be noted that this approximation is only the leading order term in a series expansion with respect to the centered multivariate moments of the underlying x_i. Second, this formula only reports the variance, but f could have a highly asymmetric distribution, warranting distinct high and low error estimates.

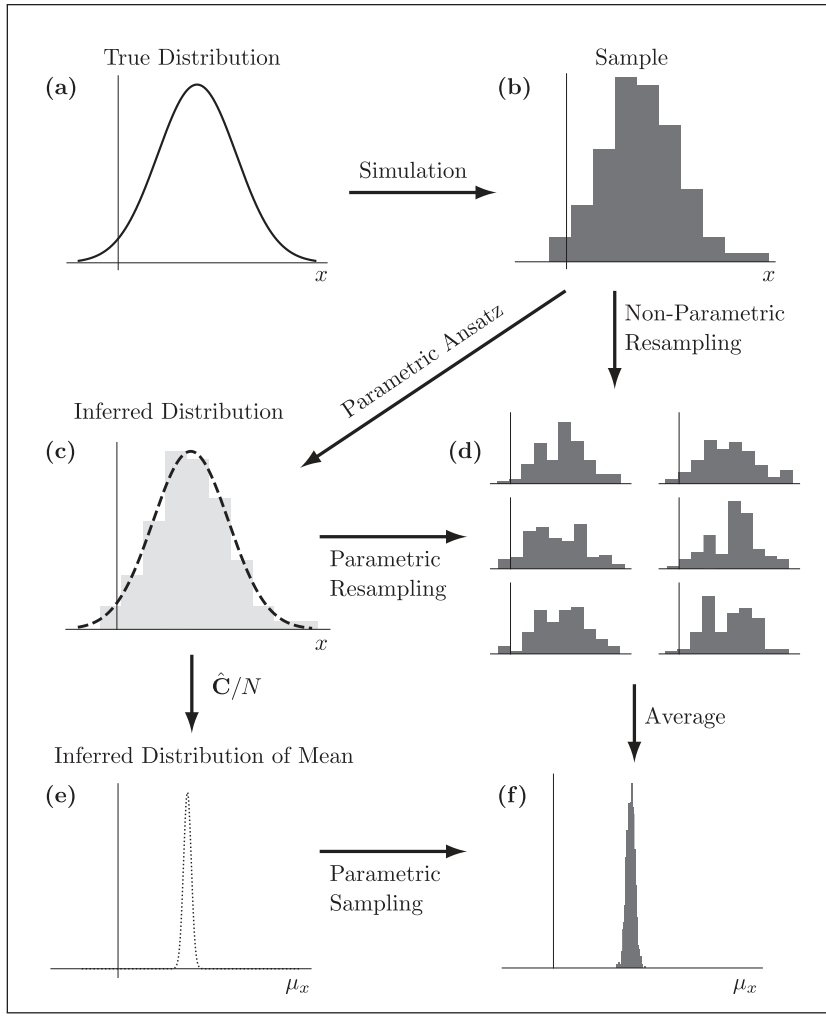

Fig. 3 Diagram of bootstrapping workflows. Sequence **(a)** → **(b)** → **(d)** → **(f)** represents the standard non-parametric bootstrap procedure for generating a bootstrap distribution of the mean, from which one can calculate a bootstrap distribution of $f(\boldsymbol{\mu})$. Alternatively, **(a)** → **(b)** → **(c)** → **(d)** → **(f)** is the *parametric* bootstrap. Sequence **(a)** → **(b)** → **(c)** → **(e)** → **(f)** circumvents the need to generate many bootstrap samples just to average each one. Importantly, in all of these cases it is assumed that the measurements in the sample (B) are uncorrelated with one another.

And finally, this procedure requires being able to reliably evaluate the gradient of f, which often ends up being the most burdensome requirement of Eq. (13) in practical applications. A well-known method made possible by modern

computing which solves all of these issues is the *bootstrap* procedure (Efron, 1992; Efron & Tibshirani, 1994). The bootstrap broadly comes in two flavors: non-parametric and parametric, though both are based on the same principle of treating the sample distribution as though it were the true distribution. Fig. 3 provides a diagrammatic representation of the sequence of steps involved in the different bootstrap methods which we now present.

In the non-parametric bootstrap, we draw N of our measured values of $\boldsymbol{x}$ from our data set $\boldsymbol{\Omega}_{\boldsymbol{x}}$ with replacement, generating a new set $\boldsymbol{\Omega}_{\boldsymbol{x}}^{\mathbf{1}}$ (our first "bootstrap sample", see Fig. 3D). We treat this set as though it is the result of a new simulation of the same system, and calculate its mean $\boldsymbol{\mu}^1 = \frac{1}{N}\sum_{\boldsymbol{x}_i \in \Omega_{\boldsymbol{x}}^1} \boldsymbol{x}_i$ and our observable of interest, $f^{\mathbf{1}} = f(\boldsymbol{\mu}^{\mathbf{1}})$. We then repeat this procedure many times, building up sets of our means, $\boldsymbol{\Omega}_{\boldsymbol{\mu}} = \{\boldsymbol{\mu}^{\mathbf{k}}\}$ (see Fig. 3F), and of our observables, $\boldsymbol{\Omega}_f = \{f^{\mathbf{k}}\}$. These sets represent our best guesses for the distributions of $\boldsymbol{\mu}$ and f, given our sample data. The more bootstrap samples we draw, the more precisely we estimate this "best guess" distribution. This process corresponds to the sequence **(a)** → **(b)** → **(d)** → **(f)** in (Fig. 3).

It is critical to note that this procedure is only valid when our samples of $\boldsymbol{x}$ are uncorrelated with one another, as was assumed at the outset of this section. The fact that the components of $\boldsymbol{x}$ are internally correlated with one another does not affect us here, as long as we resample entire vectors $\boldsymbol{x}$, preserving these internal correlations. When space and time correlations manifest together (as they inevitably do), more care must be taken, as will be discussed in the next section.

In practice, the number of bootstrap iterations varies from hundreds to thousands, with the cutoff criterion usually being convergence of a chosen summary statistic to the desired precision. If the resulting distribution for f is symmetric, one can simply use its variance (provided it exists) to communicate the statistical uncertainty. In more general situations, simplistic asymmetric confidence intervals can be reported by way of reverse percentiles (Efron & Tibshirani, 1994), for example for a 90% confidence interval, locate the value f_+ above which the largest 5% of the bootstrapped values reside, and similarly determine f_- below which the lowest 5% fall. Though there exist much more sophisticated and robust methods to generate confidence intervals by bootstrapping, this simple procedure is usually sufficient for honest error estimation.

In the parametric bootstrap, one makes the additional assumption that the observed data originate from some underlying distribution characterized by a set of parameters (hence the name), which first must be inferred

from the data. New bootstrap sample sets are then generated by drawing from this distribution, rather than re-sampling directly from the actual data set (sequence **(b)** → **(c)** → **(d)** in Fig. 3). As in many cases, our assumption here will be that of a Gaussian distribution. The only stumbling block to be mindful of here is that we must generate resamples of $\boldsymbol{x}$ with the proper internal correlations; that is, we must sample from the complete M-dimensional multivariate Gaussian distribution with mean $\hat{\boldsymbol{\mu}}$ and covariance matrix $\hat{\mathbf{C}}$ (the distribution parameters inferred from our data). Many scientific libraries, such as the Python package `SciPy` (Virtanen et al., 2020), provide built-in functions for generating random vectors from multivariate Gaussian distributions. However, it is instructive to see how to generate correlated variables by hand, as the procedure is not limited to Gaussians.

Suppose that we are able to generate individual i.i.d. random variables g_i with $\langle g_i \rangle = 0$ and $\langle g_i^2 \rangle = 1$. To generate a random vector $\boldsymbol{y}$ with given mean $\boldsymbol{\mu}$ and covariance $\mathbf{C}$, we first calculate its Cholesky factorization, $\mathbf{C} = \mathbf{L}\mathbf{L}^\top$ (Press et al., 2007). Then, the desired result is simply $\boldsymbol{y} = \boldsymbol{\mu} + \mathbf{L}\boldsymbol{g}$, where $\boldsymbol{g}$ is a vector whose components are i.i.d. g_i of the desired distribution type, such that $\langle \mathbf{g}\mathbf{g}^\top \rangle = \mathbf{I}$, the identity matrix. That $\langle \boldsymbol{y} \rangle = \boldsymbol{\mu}$ is trivial, and the covariance proof is pleasantly simple:

$$\langle (\boldsymbol{y} - \boldsymbol{\mu})(\boldsymbol{y} - \boldsymbol{\mu})^\top \rangle = \langle (\mathbf{L}\boldsymbol{g})(\mathbf{L}\boldsymbol{g})^\top \rangle = \mathbf{L}\langle \boldsymbol{g}\boldsymbol{g}^\top \rangle \mathbf{L}^\top = \mathbf{L}\mathbf{L}^\top = \mathbf{C}. \tag{14}$$

With this tool in hand, one can easily carry out the parametric bootstrap resampling, drawing new sets of "measurements" of $\boldsymbol{x}$ from the parametric distribution, and calculating $\hat{\boldsymbol{\mu}}$ and $f(\hat{\boldsymbol{\mu}})$.

However, given that what we are interested in here is $f(\boldsymbol{\mu})$, and the distribution for $\boldsymbol{\mu}$ should be Gaussian under most circumstances due to the central limit theorem, we have an opportunity to "cheat" a bit with the parametric bootstrap. As mentioned above, the covariance of the mean of N independent measurements is simply $\mathbf{C}/N$. Rather than re-sampling $\boldsymbol{x}$ just to average each set, we can skip to sampling from the inferred Gaussian distribution of the mean with covariance $\hat{\mathbf{C}}_\mu = \hat{\mathbf{C}}/N$. This is depicted in (Fig. 3) as the sequence **(c)** → **(e)** → **(f)**. This may seem trivial at the moment, but it will be important for our parametric bootstrap treatment of spatiotemporal correlations in the next section.

3.3 Combined spatiotemporal correlation

Now we will merge what we have laid out in the prior two sections to handle the case presented to us in reality: a time series of spatially-resolved measurements, each individual component of which has correlations with its nearby neighbors as well as with temporally close measurements. Our goal remains the same as in the previous section; that is, to quantify the error in our measurement of $f(\langle \boldsymbol{x} \rangle) = f(\langle x_1 \rangle, \ldots, \langle x_M \rangle)$. We may imagine our measurements comprising a matrix $\mathbf{X}$, whose elements x_{ij} correspond to space index i and time index j. In this framework, one can in principle quantify the spatiotemporal correlations by way of a rank-four covariance tensor with elements $C_{ij,kl} \equiv \mathrm{Cov}(x_{ij}, x_{kl})$. For all our sakes, we will avoid dealing directly with such an object. Instead, we will combine the techniques presented in the prior two sections by blocking and bootstrap sampling our time series for the (spatial) vector $\boldsymbol{x}(t)$ in two distinct ways. In the non-parametric scheme, we will generate bootstrapped time series by randomly selecting contiguous blocks of the original time series. In the second scheme, we will block-average the covariance of the mean and use it to directly generate a parametrically bootstrapped distribution of values for $\boldsymbol{\mu}_{\boldsymbol{x}}$, much like path **(c)** → **(e)** → **(f)** of Fig. 3.

To keep us from getting lost in the multi-dimensional sauce, let us proceed with a concrete, but synthetic, example in mind. Consider a time series of $N = 2^{14}$ measurements (taken at times $\{t_1, \ldots, t_N\}$) of Gaussian vectors $\boldsymbol{x}$ with $M = 25$ components. The time correlation for a chosen spatial component x_i of the vector $\boldsymbol{x}$ is

$$c_i(\tau) \equiv \frac{1}{\sigma_i^2} \left(\langle x_i(t+\tau)\, x_i(t) \rangle - \langle x_i^2 \rangle \right) = \mathrm{e}^{-\tau/T} \tag{15}$$

with $T = 4$. These components are also spatially correlated at any fixed time point t with a time independent exponential form,

$$C_{ij} \equiv \big\langle (x_i(t) - \langle x_i \rangle)(x_j(t) - \langle x_j \rangle) \big\rangle = \sigma_i \sigma_j\, \mathrm{e}^{-|i-j|/\lambda} \tag{16}$$

with decay length $\lambda = 3$ (this being step units as before; physically $\lambda = 3\Delta z$ if one is thinking of a stress profile measured at discrete z values). The standard deviation σ_i of each component of $\boldsymbol{x}$ is taken to vary with i, the form of which does not particularly concern us, and the spatial covariance matrix $\mathbf{C}$ is visualized in (Fig. 4). We will refer back to this example process as we explore how to deal with space-time correlation.

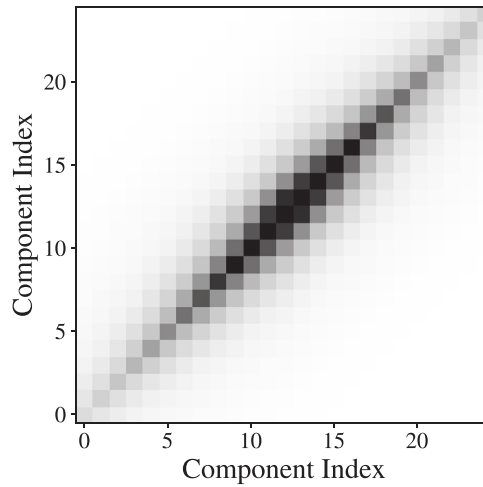

Fig. 4 Visualization of the spatial covariance matrix **C** defined in Eq. (16). Darker pixels correspond to larger covariance. The diagonal elements are the variances σ_i^2 of the components x_i of $\boldsymbol{x}$.

3.3.1 Non-parametric procedure

If we simply follow the same non-parametric bootstrap procedure from the previous section to build a new bootstrapped time series, that is, draw N of our vectors $\boldsymbol{x}(t_i)$ randomly with replacement, the shuffling of time points will destroy all of the time correlation information encoded into the original series. This will result in under-estimated errors for the exact reasons spelled out in Sec. 3.1. If one is intent on trying to generate bootstrapped time series data, then the re-sampling must be done in a way which attempts to preserve the time correlations present in the data.

The simplest way of trying to do this is a procedure known as the "simple" block bootstrap (Kunsch, 1989). In this scheme, the time series data are divided into non-overlapping blocks of b consecutive measurements, which are then re-sampled (with replacement, as before) in order to construct a bootstrapped time series. By maintaining the local temporal ordering within these blocks, (some of) the correlated behavior of the original time series is reproduced in the bootstrap sets. This procedure is diagrammed in Fig. 5. Intuitively, one should make the blocks sufficiently long as to capture the longest correlation time present in the time series. As we shall see in a moment, this intuition may be misleading, as it is generally necessary to make the blocks significantly longer than the longest correlation time present.

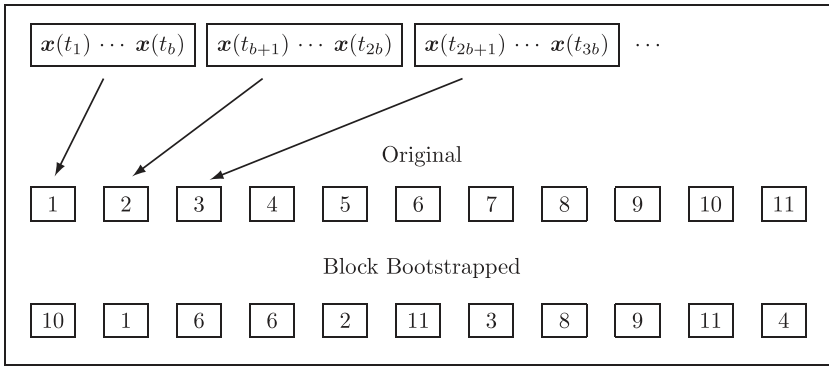

Fig. 5 In the simple block bootstrap, the time series data for $\boldsymbol{x}(t)$ is divided into blocks of length b, which are then taken as the basic units which are selected at random to construct a bootstrapped time series.

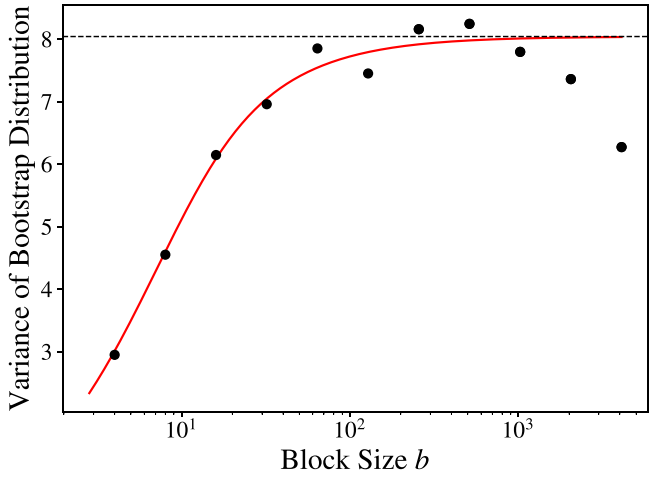

Fig. 6 Effect of block size b on the simple block-bootstrap. Variance of bootstrap distribution of the mean is plotted as a function of block size b relative to the naïvely estimated variances (i.e., block size $b = 1$). Computed for the process described by Eqs. (15) and (16) with $N_{\text{boot}} = 500$. The red curve is the square of Eq. (9), with $B(k) = b$, the horizontal axis here. The horizontal dashed line is the saturation value of $(1 + c)/(1-c)$.

After generating a large number N_{boot} of block-bootstrapped time series, we can take the mean of each one in order to generate our bootstrap distribution of the mean as before (as in path **(b)** → **(d)** → **(f)** in Fig. 3). The natural thing to investigate now is the impact of the block size b on the outcome of this process. Fig. 6 shows how the diagonal elements of the inferred covariance of the mean $\hat{\mathbf{C}}_{\mu}$ vary on average as a function of b for a

realization of our benchmark process described by Eqs. (15) and (16). The naïve bootstrap which ignores time correlations corresponds to $b = 1$, and so the values presented are divided by this result, such that the curve goes to 1 for $b = 1$.

As should be expected, as the block size increases, so does the variance of our bootstrap distribution of the mean, up to a saturation or plateau value, as was the case for our block-averaging procedure in Sec. 3.1. Perhaps unexpected is that this is not just similar, but in fact follows the same theoretical blocking curve presented in Eq. (9). The square of this is plotted as the red curve in Fig. 6 (squared because we are plotting the variance rather than the standard deviation of the mean distribution), in which we simply substitute $B(k) = b$, as this is our control parameter now rather than blocking iteration k. We cannot continue to increase block size without bound, as we eventually have blocks comparable in length to our original time series, and thus have very few distinct bootstrapped series to create from them. As such, for high values of b, the covariance of the mean becomes erratic, in analogy to the previous blocking situation where the curves become erratic when there is not much data left to block-average.

As a final note on non-parametric block bootstrapping, bear in mind that this simple procedure of selecting non-overlapping blocks of fixed size is only the simplest form of block bootstrapping. It can be generalized to overlapping blocks, and for varying block size (Davison & Hinkley, 1997). Of particular interest is the stationary block bootstrap (Politis & Romano, 1994), wherein block sizes are chosen randomly from a geometric distribution. The resulting bootstrapped time series then retains the property of stationarity, in line with the original data. We defer the interested reader to the literature for these more advanced techniques.

3.3.2 Parametric procedure

The non-parametric block bootstrap procedure just covered inadvertently brought us back to the block-averaging process from Sec. 3.1. For our next approach to spatiotemporal correlations, we will directly apply the procedure of block-averaging to our time series $\boldsymbol{x}(t)$ in order to calculate the blocked covariance of the mean $\hat{\mathbf{C}}_{\mu}^{\square}$, the higher-dimensional analogy to our blocked error of the mean for scalar blocking.

The blocking procedure itself is schematically identical to that already shown in Fig. 1, with the added detail that the individual measurements are vectors $\boldsymbol{x}$ with M components. What *is* different is what we calculate at each blocking order k: our naïve estimator error of the mean $\hat{\sigma}_{\mu} = \hat{\sigma}_k / \sqrt{N_k}$

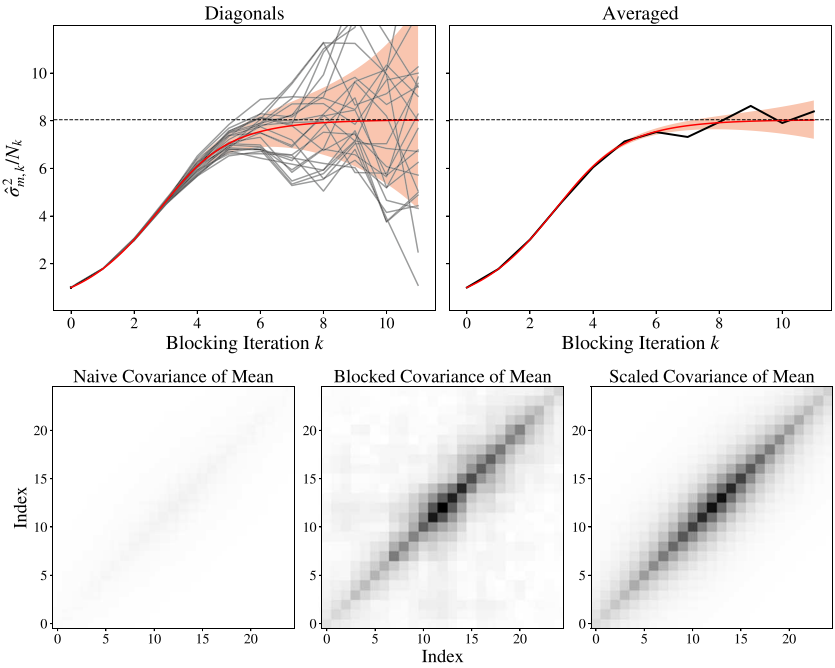

Fig. 7 Top Left: Blocking curves for the diagonal elements of the covariance of the mean, $\hat{\mathbf{C}}/N$, normalized to the naïve estimate; compare to (Fig. 2). Top Right: Averaging all of the blocking curves from the first figure produces a more well-behaved blocking curve. Note: the red shaded standard error is based on the assumption of independent processes, which is *not* the case here due to spatial correlation. Bottom Left: Naïve covariance of the mean without blocking. Bottom Center: Blocked covariance of the mean $\hat{\mathbf{C}}^{\square}_{\boldsymbol{\mu}}$ found by averaging $\hat{\mathbf{C}}_k/N_k$ for $k = 6$, 7, and 8 (beginning of the plateau region in top left plot). Bottom Right: Covariance of the mean estimated by uniformly scaling up the naïve estimate (bottom left) by the blocking plateau value $(1 + c)/(1 - c)$ see Eq. (9). The gray scale is the same across all figures in the bottom row, with white corresponding to zero and darker pixels being higher values.

generalizes to the naïve estimator of the covariance matrix of the mean $\hat{\mathbf{C}}_{\boldsymbol{\mu}} = \hat{\mathbf{C}}_k/N_k$, with the covariance matrix being estimated as usual by Eq. (12). Note that tracking this process for any one of the diagonal components of the matrix is identical (save for a square root) to the situation of scalar blocking we tackled before. For our benchmark process of Eqs. (15) and (16), the blocking curves for the diagonal elements are shown in the top-left plot of (Fig. 7), together with the square of Eq. (9) and shaded error estimate from Eq. (11). Alongside this is shown the result of averaging these curves, which naturally yields a more well-behaved blocking curve. The shaded error region for this plot is

simply scaled down by $1/\sqrt{M}$ (1/5 for our case), based on the result of averaging M Gaussian RVs. This is not precisely correct though, due to the diagonal elements being spatially correlated, but we can at least take this as a lower bound on the error of the mean blocking curve as k increases.

The lower three plots of (Fig. 7) show the full estimated covariance matrices $\hat{\mathbf{C}}_\mu$ from three different calculations. On the left is the naïve result, taking $\hat{\mathbf{C}}/N$ at face value. It is scarcely visible due to the color scale being shared across all three plots, indicating how dramatically this calculation under-estimates the covariance of the mean. The middle lower plot shows the result of the blocking process just described above, obtained by averaging the first few values of $\hat{\mathbf{C}}_k/N_k$ in the "plateau region" ($k = 6, 7, 8$). The final plot in the row is obtained by scaling the naïve estimate from the left plot by the estimate of the plateau value of the normalized blocking curve (dashed line in upper figures, corresponding to $(1 + c)/(1 - c)$ from Eq. (9). This produces a much less noisy result than the directly blocked version (center), but importantly relies on the assumption that the correlation time T does not vary between components of the vector $\boldsymbol{x}$, which is true in our benchmark case (and likely true in most real situations of interest here), but not generally.

With our blocked covariance of the mean $\hat{\mathbf{C}}_\mu^{\square}$ in hand, either from averaging the plateau region or the simple scaling method, we are able to generate samples of $\boldsymbol{\mu}$ parametrically from our inferred distribution of the mean. This corresponds to path **(b)** → **(c)** → **(e)** → **(f)** in Fig. 3, with the important modification being that we no longer use the naïve estimate $\hat{\mathbf{C}}/N$ in going from **(c)** to **(e)**, but rather use our block-averaging procedure to determine the parameters of the inferred Gaussian distribution of the mean. Just as before, once we have a bootstrap sample of $\boldsymbol{\mu}$, we feed this to our function to generate our bootstrap distribution of f, from which we calculate the desired statistics.

4. Applications and usage

4.1 Examples

Having now gone over the ins and outs of the statistical treatment in general language, we return to our context of interest: the membrane lateral stress profile. In practical situations, we measure $\Sigma(z)$, defined in Eq. (2), at a finite set of locations z_i with even spacing $\Delta z = z_{i+1} - z_i$. These measurements are

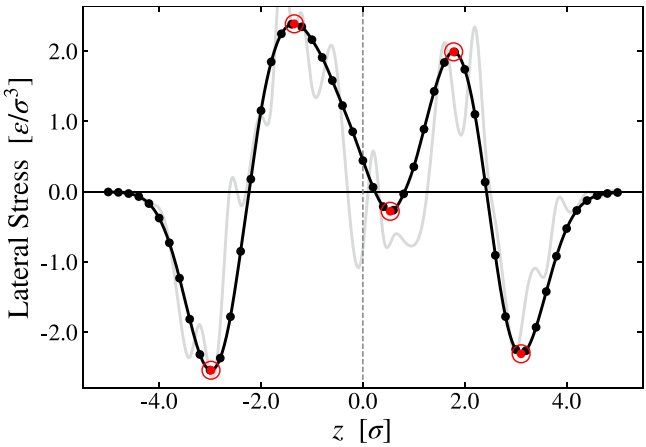

Fig. 8 Lateral stress profile for an asymmetric Cooke model membrane. The bilayer midplane is located at $z = 0$ (dashed vertical line). Black dots are average lateral stress measurements, whose standard errors (computed by blocking) are comparable to the size of the plotted points. The solid curve is the cubic spline interpolation. The circled red points indicate the locations of the extrema of the interpolated stress profile. The faint grey curve is a spline interpolation of the instantaneous stress profile measured from one simulation snapshot. The units ε and σ appearing in the labels are the coarse-grained energy and length, respectively.

themselves taken at fixed time intervals $\Delta t = t_{i+1} - t_i$. The stress profile is usually conceived of as a well-defined continuum function, such that it is obtained by averaging instantaneous stress measurements taken throughout the duration of a simulation. Quantities of interest are then computed from this mean stress profile. For our examples here, we will focus on the lateral stress profile measured from membrane simulations employing the ultra-coarse-grained Cooke lipid model (Cooke et al., 2005) with its recent flip-fix modification (Foley & Deserno, 2020). The stress profile for one such membrane is shown in Fig. 8. The structure of the stress profile for this implicit-solvent model is rather different (in general, simpler) from that of more finely-resolved models, but this is not of concern at the moment, as the observables of interest and methods of analysis are unchanged by this fact.

4.1.1 Tension and higher moments

As a reminder, the n^{th} monolayer moment $\mathcal{M}_n^+$ of $\Sigma(z)$ is defined by

$$\mathcal{M}_n^+ = \int_0^{\cdots} \mathrm{d}z \; \Sigma(z)\,(z - z_0)^n, \tag{17}$$

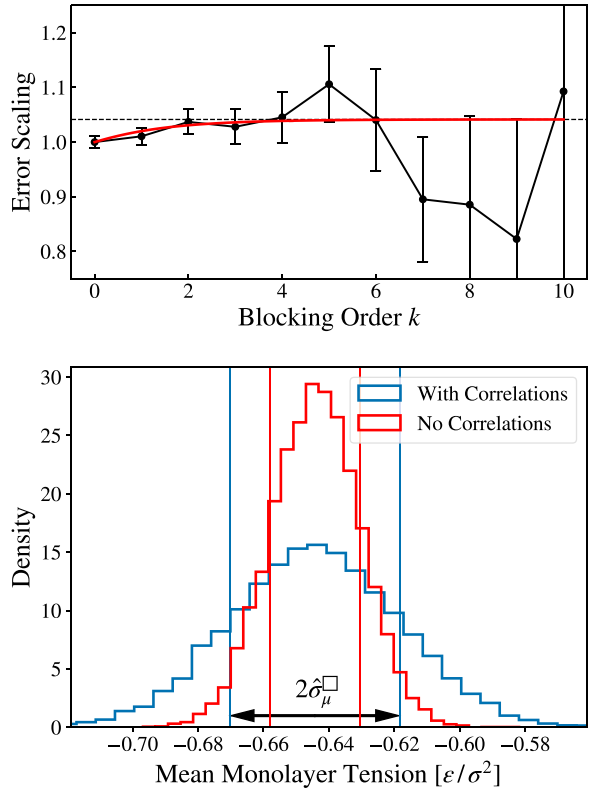

Fig. 9 Top: Blocking curve for the tension of the upper monolayer of the membrane whose stress profile is shown in Fig. 8. The red curve is the fit to Eq. (9). Bottom: Parametric bootstrap distributions of the mean monolayer tension for the same system ($N_{\text{boot}} = 5000$). The red curve is calculated without accounting for correlations, leading to an overly-narrow distribution. The width (mean ± standard deviation) of the correctly bootstrapped distribution (blue) is seen to agree with the blocked error of the mean $\hat{\sigma}_{\mu}^{\square}$ calculated based on the top plot. The units ε and σ appearing in the labels are the coarse-grained energy and length, respectively.

where the + indicates that we are defining this moment over the "upper" monolayer (located in the region of positive z). The zeroth moment is the tension, and the first and second moments are related to the mean and Gaussian curvature moduli in the surface theory. Being linear functionals of $\Sigma(z)$, these moments are simple enough observables that it does not matter whether we calculate them from the mean stress profile or for each instantaneous stress measurement and then average. This means both that their error analysis is in principle not as tricky and that we can use them as a

way of verifying that our more sophisticated methods reproduce the same results.

First, we integrate each instantaneous measurement of $\Sigma(z)$ over the upper half of the profile. Since we have measured it at fixed, equally-spaced points, this is accomplished using one's favorite Newton-Cotes quadrature (or, if one had the foresight to measure at $2^k + 1$ values of z, Romberg integration (Sauer, 2011)). This gives us a scalar time series of instantaneous monolayer tensions Σ_{m+} for the upper leaflet, whose error can be estimated *via* block averaging (Sec. 3.1) to obtain the blocked error of the mean, $\hat{\sigma}_{\mu}^{\square}$. The normalized blocking curve for this tension is shown in the upper plot of Fig. 9. The fit to the theory presented in Eq. (9) isn't quite as good as in the artificially generated examples; this could be due to non-exponential temporal correlations. Additionally, the tension measurements are found to only be weakly correlated in time in this simulation (blocking only results in a $\sim$ 4% increase in standard error). The time interval between measurements is $\Delta t = 10\tau$, where τ is the coarse-grained time unit and the production portion of the simulation lasts $8 \cdot 10^4\ \tau$.

The (unnecessarily, in this case) more complicated way to get the same answer is to carry out one of the bootstrapping procedures presented in Sec. 3.3 to generate a bootstrap distribution for the mean tension of the upper leaflet. The blue histogram in the lower plot of Fig. 9 shows exactly this, calculated *via* the parametric procedure presented before. That is, we

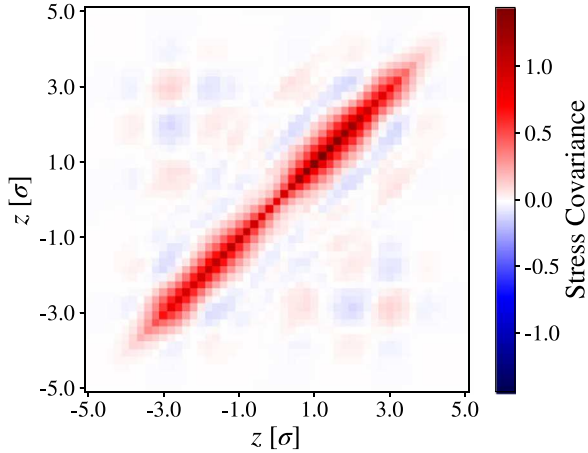

Fig. 10 Visualization of the covariance matrix for the lateral stress profile plotted in (Fig. 8). Here σ is the coarse-grained length unit.

first block-average the spatial covariance matrix of the stress profile (shown in Fig. 10) to obtain the blocked covariance of the mean. From this, we generate N_{boot} mean stress profiles from a Gaussian distribution with this prescribed covariance. Each of these is integrated to generate a mean tension, which is then binned to produce the histogram.

Also shown in Fig. 9 is an example of what happens if one does not account for correlations, resulting in the red histogram with nearly half the width of the proper mean distribution. This is generated by two blunders: The first is the skipping of the blocking step and using the naïve covariance of the mean (which in this case is a relatively minor contribution). The second is that the parametric samples are generated assuming that each $\Sigma(z_i)$ is independent from all its neighbors; that is, we ignore the off-diagonal elements of the spatial covariance matrix. In the non-parametric case, this is equivalent to resampling each $\Sigma(z_i)$ independently instead of redrawing entire stress profile snapshots.

As can be seen, spatiotemporal correlations in stress measurements can result in pronounced under-estimation of error measures of the zeroth moment. The same logic carries through to higher moments as well, with the only modification being the multiple of $(z - z_0)^n$ in the integrand.

4.1.2 Locations of stress extrema

Next, we will quantify the errors in estimates of structural features of the lateral stress profile, as properties such as maxima and minima are sometimes attributed special theoretical importance (Campelo et al., 2014). In Fig. 8 we have circled the red points lying at the five local extrema of the stress profile. For the equilibrium simulation average $\Sigma(z)$, these peaks are well-defined. However, these are emergent structures: instantaneous snapshots of the lateral stress profile, like the one shown as the thin grey line in Fig. 8, consist of noisy rapid oscillations, masking the existence of the time-averaged extrema. As such, we no longer have the option of measuring these observables for each configuration, as in the previous case, and we instead turn to the methods of Sec. 3.3.

In fact, the main portion of work necessary to accomplish our analysis here has already been covered in the previous example. We need a bootstrap sample of mean lateral stress profiles $\Sigma(z)$, which in the previous example we computed parametrically. Here, for the sake of comparison, we also carry out the non-parametric method as well. The equilibrium portion of the trajectory (8000 measurements) was divided into 50 blocks of 160 consecutive measurements. $N_{\text{boot}} = 5000$

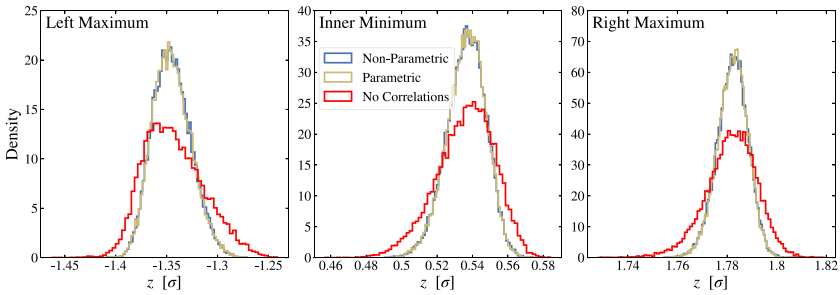

Fig. 11 Bootstrap distributions of the middle three extrema of the stress profile in Fig. 8. All histograms were created from $N_{\text{boot}} = 5000$ values. The red histogram shows the resulting (incorrect) bootstrap distribution when correlations are ignored. σ is the coarse-grained length unit.

bootstrapped time series of equal length were then constructed by randomly selecting from these blocks. For each of these series, a mean stress profile is calculated and interpolated by a natural cubic spline in order to locate extrema.

Fig. 11 plots the distributions of the locations of the central three extrema of the stress profile using both the non-parametric and parametric approaches (blue and yellow, respectively). The two methods agree excellently. Although the effect is somewhat small (partly due to having good sampling statistics), we can see that these distributions are skewed. This is one of the many benefits of the bootstrap: since we have a distribution, we can calculate asymmetric confidence intervals from the quantiles. For the three extrema shown in the figure, the locations are $z = -1.346\,\sigma$, $z = 0.537\,\sigma$, and $z = 1.782\,\sigma$ with 95% confidence intervals $[-1.379, -1.305]$, $[0.514, 0.557]$, and $[1.769, 1.794]$, respectively.

Also shown in the plots of Fig. 11 as the red histograms are the bootstrap distributions obtained if one erroneously ignores correlations and re-samples each individual stress profile point individually. Note that we see exactly the opposite effect to what we saw in the previous example: ignoring correlations here results in significantly over-estimating the widths of the distributions. This is actually rather intuitive; if one re-samples neighboring points independently, the locations of maxima and minima are much more likely to move than if neighboring points move in a correlated manner. These wider distributions also have more pronounced skewness. Thus, we see that spatiotemporal correlations can have strong effects on error estimates in either direction!

4.1.3 Lateral stretching modulus profile and neutral surface

Finally, we turn now to our most complicated example: the lateral stretching modulus profile, and from it, the neutral surface location z_0. As mentioned in the introduction, the lateral stretching modulus profile, which we denote by $\lambda(z)$, essentially gives an area expansion modulus per unit thickness to every height within the membrane and is defined as

$$\lambda(z) = A\frac{\partial\Sigma(z)}{\partial A}.$$

While it is an interesting quantity in its own right, it has particular significance for providing a way to calculate the location of the monolayer neutral surface z_0, the location of the reference surface for which bending and stretching de-couple in the free energy. In particular, it has been shown (Campelo et al., 2014) that

$$z_{0+} = \frac{\int_0^{\cdots} \mathrm{d}z\; z\lambda(z)}{\int_0^{\cdots} \mathrm{d}z\; \lambda(z)}. \tag{18}$$

Let's briefly outline the steps we need to take just to compute this numerically, before tackling the steps required to put error bars on the number we get. We first run a few small (256 lipids), flat, symmetric membrane simulations at increasing area strains $u = (A - A_0)/A_0$ and calculate the lateral stress profile $\Sigma(z; u)$ from each one; from these, we can numerically approximate $\partial\Sigma/\partial u = A_0\ \partial\Sigma/\partial A$ by fitting a line through $\Sigma(z_i, u)$ for each fixed z_i point.[14] We take this as our approximate $\lambda(z)$ measured for the rest area A_0 and then numerically evaluate the integrals in Eq. (18) to arrive at our z_0 measurement.

Multiple stress profiles, linear regressions, and numerical integrations all compose together to produce the final answer. This is once again where the conceptual simplicity of bootstrapping shines: we need a distribution of z_0 values to quantify our uncertainty, and we get that by re-sampling our data and re-running the calculation for every bootstrap iteration. More concretely, we simulated our small membrane patch at six area strains, starting from zero strain, yielding six lateral stress profiles

[14] Actually, it is subtly more complicated than this, as briefly mentioned in Campelo et al. (2014) and further discussed in the SI of Foley & Deserno, 2024. Since the membrane thickness changes for different values of area strain u, the stress profiles from each simulation must be re-scaled such that material points from each simulation are being compared correctly across measured functions $\Sigma(z)$. This requires interpolating between the discretely measured stresses, which we do with natural cubic splines.

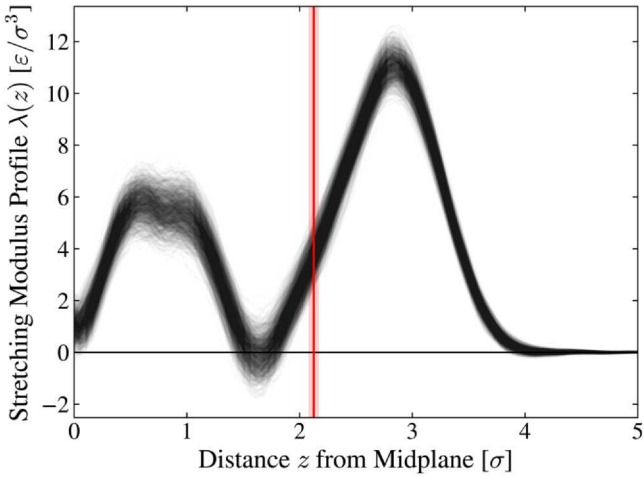

Fig. 12 Bootstrap sample of $N_{\text{boot}} = 1000$ lateral stretching modulus profiles $\lambda(z)$, plotted as faint black curves. The neutral surface location z_0 as calculated from Eq. (18) is plotted as the vertical red line, with its one-sigma confidence interval shown as the shaded band. The units ε and σ appearing in the labels are the coarse-grained energy and length, respectively.

$\Sigma(z)$. For each bootstrap iteration, we resample all six of the these profiles using the parametric method of Sec. 3.3, and then proceed through the steps described above to calculate $\lambda(z)$ and then z_0. We repeated this $N_{\text{boot}} = 1000$ times, and our bootstrapped $\lambda(z)$ profiles are plotted in Fig. 12, along with the inferred value for z_0 and its one-sigma confidence interval. If one neglects correlations, the resulting error bars are underestimated by about a factor of two (not plotted).

4.2 Parameter tuning

All numerical methods come with a list of parameters that define the performance of one's algorithm, and whose choice is based on some meta-considerations. For MD simulations, some very obvious choices include total simulation time, system size, parameters in the thermostat or barostat, how much of the initial trajectory to discard, etc. Here we wish to specifically mention a few choices directly related to the routines we have discussed, and make some suggestions for how to pick good values.

4.2.1 Stress profile analysis

- An important first choice is the membrane size. Bigger systems reduce finite-size effects and increase the statistics, but the increased undulations

blur the stress profile in the normal direction, say by some Gaussian kernel $g(x)$. One could deal with this in multiple ways:

- Put the bilayer under a minor tension. This need not strongly affect the stress profile, since fairly minor tensions already suffice to flatten membranes. A membrane with bending rigidity κ under tension Σ has its undulations strongly suppressed on scales larger than $\lambda \sim \sqrt{\kappa/\Sigma}$. For typical values $\kappa \sim 30\, k_B T$, a small tension $\Sigma = 0.5\,\mathrm{mN/m}$ gives $\lambda \sim 16\,\mathrm{nm}$.
- Another possibility, which we would only recommend if blurring is still quite suite small, is to try a raw deconvolution *via* the fairly simplistic procedure $\sigma_0(x) := \overleftarrow{\mathrm{FFT}}\left[\overrightarrow{\mathrm{FFT}}[\sigma(x)] / \overrightarrow{\mathrm{FFT}}[g(x)]\right]$.
- The approach that is probably both safest and most straightforward remains to simply simulate a smaller membrane. Eq. (6) might be a good first guide to how small is "small enough".

- How many bins should one use to resolve the stress profile? Many useful recipes have been developed for how to pick a suitable bins-size while sampling a histogram, but this usually refers to a situation where a fixed number of measurements are to be binned. This is not the case here since the number of entries in our finite Δz stress bins arise as averages over a large number of inter-particle stress measurements that are strongly correlated within a bin and even across bins. At this point, we have no simple answer and instead encourage the reader to make a good judgement call that balances loss of detail (too few bins) against just plotting noise (too many bins). Since the values of the stress are generally calculated at evenly spaced intervals Δz, one practical consideration is to choose the number of points of the form $2^k + 1$ for some k, as this allows one to numerically compute integrals to higher accuracy using Romberg's method (Sauer, 2011).
- We also wish to point out that binning is not exclusively a matter of numerics, since locally resolving continuum thermodynamic observables requires some spatial averaging, which to first order is what the bins are doing. Picking too small a bin size is therefore not just numerically awkward but also physically dubious. Clearly, this question deserves further study.

4.2.2 Blocking

- What block size do we choose? Since we want to block many times, picking $b = 2$ (or, rather, picking $N = 2^n$ with some n) seems most natural, but there are other options. Realize that all we need is a number

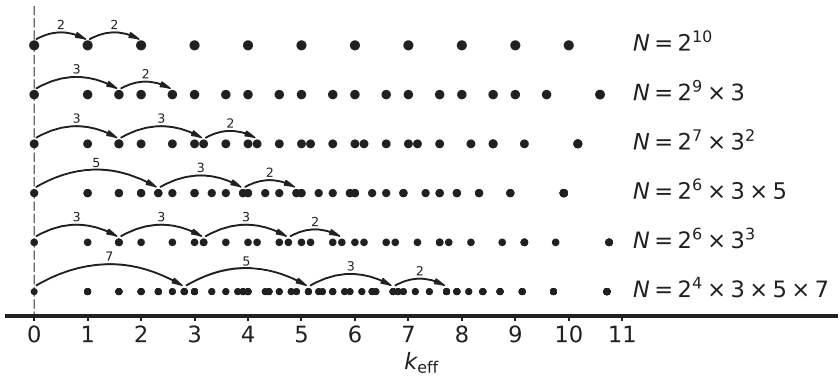

Fig. 13 Example how a number N of data points not equal to a power of 2 can be turned into a larger and denser set of effective k values along the blocking axis. The arrows show some example blocking sequences.

that contains many small prime factors. For instance, $N = 1536 = 2^9 \times 3$ also offers many blocks. Furthermore, we can choose the *order* in which we pick factors and thus create more data-points on the abscissa: the two sequences (3, 2, …, 2) and (2, …, 2, 3) create an interlaced set of two blocking ladders that captures *all* points reachable by one 3-block and the rest being 2 blocks, no matter at what point "3" happens. When we graph the resulting block averages, not having exclusively used $b = 2$, we just plot the result against an "effective" blocking number

$$k_{\text{eff}} = \log_2(f_1 \times f_2 \times f_3 \times \cdots \times f_k) = \sum_{i=1}^{k} \log_2(f_i), \tag{19}$$

which is just k if all blocking factors f_i are equal to 2. Fig. 13 illustrates the principle. With many distinct factors this interlacing yields a fairly dense coverage of the blocking axis, but be warned that these values are not all independent of each other—which matters if we fit a blocking curve to it.

4.2.3 Bootstrapping

- For all bootstrapping methods, the choice of N_{boot} is a free parameter to be chosen, and it determines in how much detail one resolves the bootstrap distribution. It generally varies from hundreds to thousands depending on the situation, and the general idea is to pick a value large enough that whatever statistic you calculate from the bootstrap

distribution should come out the same every time (within the desired accuracy). That is, your error bar should not be itself a wildly varying stochastic variable just because you calculate it using a stochastic method!

- For the simple block bootstrap we must choose a block size b. As we have seen, the influence of this choice follows the same theoretical form as block averaging (see Fig. 6). This suggests that if we (somehow) know the correlation time beforehand, we can choose a block size which puts us comfortably into the plateau region of Fig. 6 by inverting Eq. (9). Barring knowing the correlation time at the outset, one can do exactly what is shown in that figure: block bootstrap with increasing block size until the error saturates. This is unfortunately somewhat slow, depending on the size of the time series.

4.3 Words of caution

The algorithms we have discussed here help to better quantify the uncertainties arising when calculating observables related to a membrane's lateral stress profile in the presence of ubiquitous spatiotemporal correlations. Once they are black-boxed into convenient packages, it is tempting to forget what's under the hood. But like any set of numerical routines, the user ought to be alert to some of the more common failure modes. In the following we collect a number of caveats, none of which are particularly mysterious, especially in hindsight. Our goal is merely to instill a certain level of vigilance in future users.

4.3.1 Blocking

- Do we even have enough data for the blocking curve to plateau? This is usually easy to see by just looking at it, but this evidently requires looking at it. One tell-tale sign that we might need to do so is an inferred correlation time that is not "sufficiently much" shorter than the total time of our trajectory. Fig. 14 gives an idea of what exactly this means by plotting the inferred correlation time T_{fit} extracted by fitting to Eq. (9) for trajectories of varying length N but always the same single-exponential correlation time T. The result is sobering: even for a trajectory 8 times longer than the true correlation time T (shaded band in Fig. 14), the estimated T_{fit} comes out around 30% too low, and the range of potential fit values is quite wide. The inset of the figure shows a selection of blocking curves from this process, indicating a lack of a clear plateau. Recall, though, that in reality you will only see one of them! This could be one that is still strongly increasing or, maybe worse, one

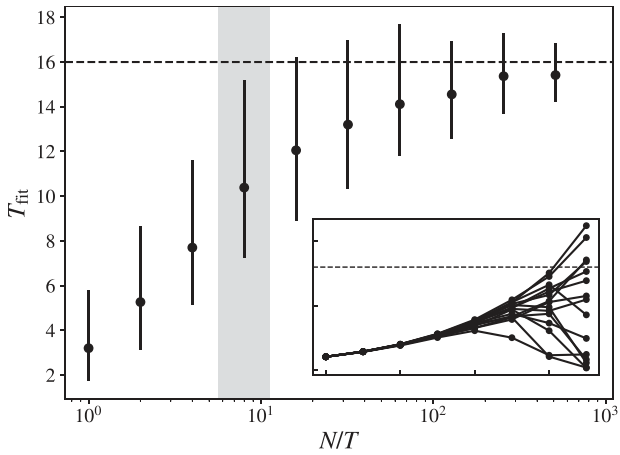

Fig. 14 Inferred correlation time T_{fit} determined from fitting the blocking curve to the theory in Eq. (9) for varying time series trajectory length N. In all cases, series are exponentially correlated Gaussian processes with correlation time $T = 16$ steps (indicated by the dashed horizontal line). Points are the median of 1000 runs, bars indicate the surrounding middle 50% of fitted values. Inset: 15 example blocking curves with N/T corresponding to the highlighted data value in the main plot, with true plateau shown by the dashed line.

that has started to decrease again, falsely suggesting that the plateau has already happened. So, if your blocking-inferred correlation time is more than $\sim 10\%$ of your trajectory length, you probably need more data!

- One may wonder whether simply measuring more often can allow us to escape the mire of "not enough data", as this would increase our N without having to actually simulate longer. Unfortunately, it is T/N which matters, where T is the correlation time in *measurement steps*. Thus, if we double our measurement frequency (cutting our time interval Δt in half) to achieve $2N$ data points, we also double $T = \tau/\Delta t$, and will not have gained any ground in our blocking analysis. Intuitively, measuring more frequently gets us more data, but these more closely-spaced data points are more highly correlated, and the net result is a wash.[15]
- We admit that there is a back door—but it is very risky. When we sample more frequently, we will of course pin down the *early parts* of the blocking curve very well. *If* we happen to know that we indeed only

[15] This statement is informed by our general experience, and tested against exponentially correlated data; it is possible that this does not hold for other forms of correlation one might encounter.

have a single correlation time T, then even those early parts determine T—and hence the plateau—quite well. The catch is that we rarely know that this is true, especially if "maybe not having sampled enough" is one of our primary concerns.

- The analytical prediction (9) assumes a process with a single exponential correlation time, but this is not always true. If it is not, the prediction fails to describe the blocking curve and will generally give an incorrect idea of both the plateau and the implied correlation time. A conceivably automatic way to check for this is to monitor the goodness of fit to the blocking data.
- The easiest-to-treat deviation from a single exponential process is one that contains two well-separated exponential time scales T_1 and T_2, giving a correlation function of the form $\sigma^2 \left(w_1 \, \mathrm{e}^{-\tau/T_1} + w_2 \, \mathrm{e}^{-\tau/T_2}\right)$, where w_1 and w_2 are weights that add to 1. It is easy to generalize Eq. (9) for this case, or even for the case of more correlation times, but the resulting function is usually *very* poorly constrained by the measured blocking curve. The reason is that we are basically on our way to demand an inverse Laplace transform—a quintessential ill-posed problem. Hence, instead of automating even a two-times (and hence three-parameter) fit, we instead recommend looking out for the failure of a one-time fit and, if a two-times scenario appears plausible, try to fit this very carefully by judiciously picking initial values for the times and the weighting factor.

4.3.2 Bootstrapping

- A full bootstrap does not merely give distribution functions for the observables of interest; it gives a joint distribution function for all of them. Hence, if these observables are $\{x_1, x_2, \ldots, x_n\}$, and one is interested in some function of them, $f(x_1, x_2, \ldots, x_n)$, we recommend to not just determine the uncertainties δx_i and from there infer the uncertainty δf by error propagation à la Eq. (13) (let alone its simplified version that drops all the correlation). Instead, one should bootstrap f, because this includes not just all correlations but accounts for all nonlinearities in the transformation from $\{x_i\}$ to f.
- We are used to taking Gaussian errors in "raw" measurements for granted, because we can frequently appeal to the Central Limit Theorem. However, for random variables that are functions of Gaussian random variables this is generally no longer true. This is challenging for any nonlinear function, but it becomes particularly bad if the function

has a singularity at a point with a finite probability density. Let us give two examples:

1. Consider a random variable X with a probability density $p_X(x)$ that is nonzero at $x = 0$, and assume we are interested in the random variable $Z = 1/X$. It is easy to see that its probability density is $p_Z(z) = p_X(1/z)/z^2$ and thus has a fat tail with $1/z^2$ asymptotics.
2. Assume that both X and Y are (uncorrelated) Gaussian random variables with zero mean and standard deviations σ_X and σ_Y. Then $Z = X/Y$ follows the Cauchy distribution $p_Z(z) = a/[\pi(a^2 + z^2)]$ with the scale parameter $a = \sigma_X/\sigma_Y$. Observe that it again has a fat tail.[16]

This in particular implies that (in either case) $p_Z(z)$ has infinite variance, the Central Limit Theorem does not apply to it, and the error of the mean does not decrease with additional measurements. We would notice this during bootstrapping because the distribution function for Z does not narrow down when doing more bootstraps.

- Having to divide two measured random variables happens of course all the time. For instance, Eq. (18) for the neutral surface z_0 is exactly of this form. But since in this case the denominator has a vanishingly small probability density at 0, we do not run into fat-tail difficulties.
- If even the raw measurements are not Gaussian, we could strive to sample from whatever non-Gaussian distribution we presume to be applicable. However, we should then recall that any subsequent least-square fitting might no longer be the correct procedure, given that χ^2-minimization is only a maximum likelihood estimator in the case of Gaussian errors.
- If we are interested in a distinct peak in $\Sigma(z)$, which we identify by running some peak-finding routine (either applied to the raw data or some numerical interpolation of it), then this can get difficult if the peak is only weakly pronounced and possibly in the vicinity of other weak peaks. The problem is that resampled data might shift the peak beyond recognition or even eliminate it altogether for some fraction of resampled profiles. Technically, this corresponds to another fat-tailed distribution function for the peak position, but the more pragmatic conclusion is that such a peak is simply not robustly enough present to reason about it. For instance, the stress profile in Fig. 15 has two clearly pronounced maxima at $z \approx \pm 1.54\,\sigma$ and two pronounced minima at

[16] While we're at it: the ratio of two Gaussians with nonzero mean and non-identical standard deviations does not even have to be monomodal. Check out $X \sim \mathcal{N}(2, 1)$ and $Y \sim \mathcal{N}(2, 5)$.

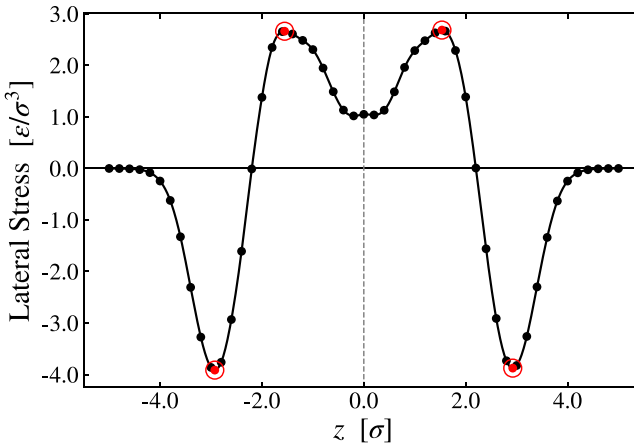

Fig. 15 Lateral stress profile for a symmetric Cooke model membrane. Note the ambiguous extrema near the midplane.

$z \approx \pm 2.93\,\sigma$, but what happens near $z = 0$? Is this a single minimum, or two very close minima with a maximum at $z = 0$? The answer depends on the individual bootstrap, which suggests to not press the point.

5. Connections

This double-volume of "Methods in Enzymology" is dedicated to membranes exhibiting lateral and/or transverse heterogeneities. And while the conceptually most visible aspect of this relates to the lipids, physical properties often have a habit of percolating down the chain of observables. For instance, if two different lipid membrane phases have different lateral stress profiles, and these phases come into contact, not only will we physically have to navigate a "compositional transition", in which somehow the phase-dependent lipid mole fractions have to switch from one value to another; we also face a "stress transition", in which not just an overall balance of tension must happen but also higher moments (such as torques) will demand attention. Depending on how dissimilar the phases are, this might lead to a local change of lipid order, the excitation of lipid tilt, a localized "kink" of the membrane across the contact line, or other interesting phenomena. For asymmetric membranes the two sides of the stress profiles are different, which gives us different leaflet-moments and hence material properties, and due to differential stress the two sides can

additionally be "offset" by some amount. All these effects are highly informative of both local and global physics, and so the stress profile is likely an even more interesting observable for such heterogeneous systems.

Let us give just a few pointers to how this might beneficially complement the methods discussed in other chapters in this volume. To begin, creating asymmetric membranes for simulations poses challenges not present in the symmetric case, namely, what relative abundance of generally different lipids one places into the two leaflets. Chapters 5 and 9 discuss in detail how this might be achieved, with Chapter 5 discussing a variety of methods while Chapter 9 discusses in detail so-called $P2_1$ boundary conditions. One possible (albeit not the only plausible) condition one might elect to achieve is a membrane in which differential stress between the leaflets is canceled. This requires measuring leaflet stresses *via* the stress profile, and comparing different methods we must be able to ascertain whether stress differences are statistically significant. Chapter 6 examines many ways in which differential stress can affect both the morphology of vesicles, as well as their propensity to undergo fission or fusion. Due to the focus on a spherical geometry, the lateral stress profile is also calculated for such curved surfaces, which gives rise to interesting new effects, but of course the uncertainty analysis discussed here will work just the same. Next, Chapter 1 discusses phase coexistence between different fluid membrane phases (such as L_o/L_d). The balance of stresses and their higher moments across the domain interfaces we mentioned above would add to our understanding of domain formation. In fact, Chapter 4 proposes a specific protocol in which the location of that interface can be controlled and aligned with the simulation box, and in such a setup the detailed mechanics of the stress-transition could be exceptionally well studied. In Chapter 14, the authors propose computing the average curvature sampled per lipid on a fluctuating bilayer, from which they infer the spontaneous curvature of individual lipids (relative to the mean spontaneous curvature of the bilayer). This is complementary to the lateral stress profile, which indicates the product of the mean spontaneous curvature and bending modulus. In principle, the combination of the two techniques, along with computation of the bending modulus, would provide absolute spontaneous curvatures for each lipid. The authors also connect specific interactions between lipids to spontaneous curvature—interactions that may be evident as peaks in the lateral stress profile.

References

Admal, N. C., & Tadmor, E. B. (2010). A unified interpretation of stress in molecular systems. *Journal of Elasticity, 100*, 63–143.

Bretscher, M. S. (1972a). Asymmetrical lipid bilayer structure for biological membranes. *Nature: New Biology, 236*(61), 11–12.

Bretscher, M. S. (1972b). Phosphatidyl-ethanolamine: Differential labelling in intact cells and cell ghosts of human erythrocytes by a membrane-impermeable reagent. *Journal of Molecular Biology, 71*(3), 523–528.

Campelo, F., Arnarez, C., Marrink, S. J., & Kozlov, M. M. (2014). Helfrich model of membrane bending: From Gibbs theory of liquid interfaces to membranes as thick anisotropic elastic layers. *Advances in Colloid and Interface Science, 208*, 25–33.

Cooke, I. R., Kremer, K., & Deserno, M. (2005). Tunable generic model for fluid bilayer membranes. *Physical Review E, 72*(1), 011506.

Davison, A. C., & Hinkley, D. V. (1997). *Bootstrap methods and their application. Number 1*. Cambridge University Press.

Efron, B., & Tibshirani, R. J. (1994). *An introduction to the bootstrap*. CRC Press.

Efron, B. (1992). *Bootstrap methods: Another look at the jackknife. Breakthroughs in statistics: Methodology and distribution*. Springer, 569–593.

Ergüder, M. F., & Deserno, M. (2021). Identifying systematic errors in a power spectral analysis of simulated lipid membranes. *The Journal of Chemical Physics, 154*(21), 214103.

Evans, E., & Heinrich, V. (2003). Dynamic strength of fluid membranes. *Comptes Rendus Physique, 4*(2), 265–274.

Flyvbjerg, H., & Petersen, H. G. (1989). Error estimates on averages of correlated data. *The Journal of Chemical Physics, 91*(1), 461–466.

Foley, S., & Deserno, M. (2020). Stabilizing leaflet asymmetry under differential stress in a highly coarse-grained lipid membrane model. *Journal of Chemical Theory and Computation, 16*(11), 7195–7206.

Foley, S. L., & Deserno, M. (2024). Asymmetric membrane "sticky tape" enables simultaneous relaxation of area and curvature in simulation. *The Journal of Chemical Physics, 160*(6), 064111.

Foley, S. L., Varma, M., Hossein, A., & Deserno, M. (2023). Elastic and thermodynamic consequences of lipid membrane asymmetry. *Emerging Topics in Life Sciences, 7*(1), 95–110.

Goetz, R., & Lipowsky, R. (1998). Computer simulations of bilayer membranes: self-assembly and interfacial tension. *The Journal of Chemical Physics, 108*(17), 7397–7409.

Hamm, M., & Kozlov, M. M. (2000). Elastic energy of tilt and bending of fluid membranes. *European Physical Journal E, 3*, 323–335.

Hess, B. (2002). Determining the shear viscosity of model liquids from molecular dynamics simulations. *The Journal of Chemical Physics, 116*(1), 209–217.

Hossein, A., & Deserno, M. (2020). Spontaneous curvature, differential stress, and bending modulus of asymmetric lipid membranes. *Biophysical Journal, 118*(3), 624–642.

Hu, M., Briguglio, J. J., & Deserno, M. (2012). Determining the gaussian curvature modulus of lipid membranes in simulations. *Biophysical Journal, 102*(6), 1403–1410.

Hu, M., de Jong, D. H., Marrink, S. J., & Deserno, M. (2013). Gaussian curvature elasticity determined from global shape transformations and local stress distributions: a comparative study using the martini model. *Faraday Discussions, 161*, 365–382.

Janke, W. (2002). Statistical analysis of simulations: Data correlations and error estimation. In J. Grotendorst, D. Marx, & A. Muramatsu (Eds.). *Quantum simulations of complex many-body systems: From theory to algorithms, volume 10 of NIC series* (pp. 423–445). Jülich: John von Neumann Institute for Computing.

Kunsch, H. R. (1989). The jackknife and the bootstrap for general stationary observations. *The Annals of Statistics, 17*(3), 1217–1241.

Lorent, J. H., Levental, K. R., Ganesan, L., Rivera-Longsworth, G., Sezgin, E., Doktorova, M., ... Levental, I. (2020). Plasma membranes are asymmetric in lipid unsaturation, packing and protein shape. *Nature Chemical Biology, 16*(6), 644–652.

Morris, C. E., & Homann, U. (2001). Cell surface area regulation and membrane tension. *The Journal of Membrane Biology, 179*, 79–102.

Nakagawa, K. M., & Noguchi, H. (2016). Nonuniqueness of local stress of three-body potentials in molecular simulations. *Physical Review E, 94*(5), 053304.

Ollila, O. H. S., Louhivuori, M., Marrink, S. J., & Vattulainen, I. (2011). Protein shape change has a major effect on the gating energy of a mechanosensitive channel. *Biophysical Journal, 100*(7), 1651–1659.

Paige, C. C. (1979). Computer solution and perturbation analysis of generalized linear least squares problems. *Mathematics of Computation, 33*(145), 171–183.

Politis, D. N., & Romano, J. P. (1994). The stationary bootstrap. *Journal of the American Statistical Association, 89*(428), 1303–1313.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (2007). *Numerical recipes: The art of scientific computing* (3rd ed.). Cambridge University Press.

Rowlinson, J. S., & Widom, B. (2003). *Molecular theory of capillarity*. Dover.

Sandra, A., & Pagano, R. E. (1978). Phospholipid asymmetry in LM cell plasma membrane derivatives: Polar head group and acyl chain distributions. *Biochemistry, 17*(2), 332–338.

Sauer, T. (2011). *Numerical analysis*. Addison-Wesley Publishing Company.

Schick, P. K., Kurica, K. B., & Chacko, G. K. (1976). Location of phosphatidylethanolamine and phosphatidylserine in the human platelet plasma membrane. *Journal of Clinical Investigation, 57*(5), 1221–1226.

Schofield, P., & Henderson, J. R. (1982). Statistical mechanics of inhomogeneous fluids. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences, 379*(1776), 231–246.

Szleifer, I., Kramer, D., Ben-Shaul, A., Gelbart, W. M., & Safran, S. A. (1990). Molecular theory of curvature elasticity in surfactant films. *The Journal of Chemical Physics, 92*(11), 6800–6817.

Terzi, M. M., Ergüder, M. F., & Deserno, M. (2019). A consistent quadratic curvature-tilt theory for fluid lipid membranes. *The Journal of Chemical Physics, 151*(16), 164108.

Vanegas, J. M., Torres-Sánchez, A., & Arroyo, M. (2014). Importance of force decomposition for local stress calculations in biomembrane molecular simulations. *Journal of Chemical Theory and Computation, 10*(2), 691–702.

Verkleij, A. J., Zwaal, R. F. A., Roelofsen, B., Comfurius, P., Kastelijn, D., & Van Deenen, L. L. M. (1973). The asymmetric distribution of phospholipids in the human red cell membrane. a combined study using phospholipases and freeze-etch electron microscopy. *Biochimica et Biophysica Acta (BBA)—Biomembranes, 323*(2), 178–193.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., ... van Mulbregt, P. SciPy 1.0 Contributors. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in python. *Nature Methods, 17*, 261–272.