Low Cost Recurrent and Asymptotically Unbiased Estimators of Statistical Uncertainty on averaged fields for DNS and LES

Margaux Boxho[1,3*], Thomas Toulorge[1], Michel Rasquin[1], Grégory Dergham[2], Koen Hillewaert[1,3]

[1*]Cenaero, Charleroi, 6041, Belgium.
[2]Safran Tech, Châteaufort, 78114, Magny-les-Hameaux, France.
[3]Department of Aerospace and Mechanics, ULiege, Liège, 4000, Belgium.

*Corresponding author(s). E-mail(s): margaux.boxho@cenaero.be;
Contributing authors: gregory.dergham@safrangroup.com;
koen.hillewaert@uliege.be;

Abstract

In the context of fundamental flow studies, experimental databases are expected to provide uncertainty margins on the measured quantities. With the rapid increase in available computational power and the development of high-resolution fluid simulation techniques, Direct Numerical Simulation and Large Eddy Simulation are increasingly used in synergy with experiments to provide a complementary view. Moreover, they can access statistical moments of the flow variables for the development, calibration, and validation of turbulence models. In this context, the quantification of statistical errors is also essential for numerical studies. Reliable estimation of these errors poses two challenges. The first challenge is the very large amount of data: the simulation can provide a large number of quantities of interest (typically about 180 quantities) over the entire domain (typically 100 million to 10 billion of degrees of freedom per equation). Ideally, one would like to quantify the error for each quantity at any point in the flow field. However, storing a long-term sequence of signals from many quantities over the entire domain for a posteriori evaluation is prohibitively expensive. The second challenge is the short time step required to resolve turbulent flows with DNS and LES. As a direct consequence, consecutive samples within the time series are highly correlated. To overcome both challenges, a novel economical co-processing approach to estimate statistical errors is proposed, based on a recursive formula and the rolling storage of short-time signals.

Keywords: Statistical Uncertainty, Turbulent Statistics, Time Series, DNS, LES, Unbiased Estimators

1 Introduction

Scale-resolving approaches for the simulation of turbulent flows, such as Large Eddy Simulations (LES) and Direct Numerical Simulations (DNS), are increasingly used in complement to experiments for the study of fundamental flows. This evolution is made possible by the rapid growth in available computational resources combined with improvements in numerical methods. Since computations provide access to all flow variables and statistical data, they represent excellent opportunities for the improvement of turbulence models, in particular using Machine Learning and Deep Learning techniques(e.g., (Duraisamy et al, 2019; Brunton et al, 2020)). In contrast to experiments, DNS results are not often accompanied by a rigorous quantification of the uncertainty on the statistics (i.e., mean velocity profile, Reynolds stress profiles, turbulent kinetic energy, lift coefficient, and so on). As pointed out by Fonseca et al (2022), the lack of convergence of statistical fields is a source of error in the Reynolds stress tensor and hence in the calibration of RANS models. Note that it is common practice in experiments to report results with uncertainty margins to allow relevant conclusions on the considered physics. Nevertheless, DNS literature does not systematically report these uncertainties.

In experiments, uncertainty can be classified, according to Favier (2010), into two categories: measurement and data acquisition errors on the one hand and uncertainty associated with the statistical data reduction on the other. One could add a third category related to the uncertainty of the operating conditions, such as the uniformity of the flow at inlets, drift of conditions during wind tunnel operations, etc.

A similar categorization can also be applied to simulation errors. Whereas the category related to statistical errors applies as such, the first category corresponds to uncertainties linked to numerical approximation error, whereas the third would be related to boundary conditions, in particular concerning potential reflections and local distortions of the flow. One could cope with the last category by considering the ”imperfections” to be actually part of the numerical experiment.

The statistical error also impacts the determination of discretization errors. In the absence of an actual exact solution, the grid convergence for simulations is usually assessed using the Richardson extrapolation, which estimates the discretization error from a sequence of simulations on progressively finer meshes. However, this method is not directly applicable to DNS because it can not separate discretization errors from statistical errors; removing the latter would require infinite simulation times.

Therefore, estimating statistical errors is of the highest priority. Fortunately, the evolution of statistical errors over time can be quantified using sampling error estimators. Then, the accepted statistical uncertainty of the quantity of interest will dictate the duration of a DNS or LES. However, this estimate is more complicated due to the high and unknown correlation in the time signal. Moreover, this correlation may vary significantly in the spatial domain.

Assuming that the data used to compute the statistics are *identically, independently distributed (i.i.d.)* samples x_i and using the CLT, the sample mean (or also called the finite time average) $\bar{x}_n$ follows a Normal distribution of mean equal to the infinite-time-average μ and standard deviation

$$\epsilon_n = \sigma/\sqrt{n}, \tag{1}$$

where σ is the standard deviation of a single sample, and n is the number of samples. This expression can be interpreted as a confidence interval, and as n increases the sample mean (Equation 6) converges to the true mean. However, in most problems, the i.i.d. assumption does not hold. In numerical experiments, where a continuous-time chaotic system is simulated, consecutive samples are highly correlated due to the small time steps required to resolve turbulence structures accurately. Therefore two samples in time are far from being independent. A first approach would be to take samples far apart in time to reduce the correlation and treat them as independent (Donzis et al, 2008). Unfortunately, such a simple approach underestimates the uncertainty if the samples are not sufficiently separated. Contrarily, if they are taken too far apart, the sampling error is overestimated. Lumley and Panofsky (1964) updated this basic estimator (Equation 1) to account for the *integral time scale* $\mathcal{T}$. However, the correlation and the integral time scale are a priori unknown and vary usually over the simulation domain.

The best option is therefore to use all available data and take correlation into account. A first approach to account for the correlation was proposed by Hoyas and Jiménez (2008). Since the temporal coherence of turbulence data is unknown *a priori*, they create an intermediate *coarse-grained* time series to evaluate the desired variance. Nonetheless, such a method is difficult to automate and requires user intervention and interpretation. A promising approach is to estimate the auto-correlation directly from the data, which is in itself a challenge. This task of estimating auto-correlation functions has already been addressed by the weather and climate communities (Trenberth, 1984). In the work of Alimohammadi and He (2016), the auto-correlation is modeled with an exponential decay of the form $\exp\left(-\alpha_f t\right)$ where α_f is a fitting parameter obtained from the available data via empirical modeling and t the time. One can also cite the work of Broersen (2002, 2006), who fitted an autoregressive (AR) model for the auto-correlation function estimation. In other words, this technique consists of formulating a statistical model of the random process. Then, the parameters of the mathematical model are adjusted via a maximum likelihood formulation (Hosking, 1981; Oliver et al, 2014). Beyhaghi et al (2018) proposed a *multiscale* method based on an auto-correlation model tuned to fit the statistic of interest for a range of different timescales. All of these methods adopt models for the integral length scale or correlation to estimate the sampling error. The recent work of Rezaeiravesh et al (2023) also modeled the ACF using a convex combination of exponential functions and provided an in-situ implementation of the developed estimator in the flow solver Nek5000.

The uncertainty estimation in the statistics of steady-state simulations has also been addressed using autoregressive moving average (ARMA) and batch means methods. Although batch means methods are widely used because they are fast, they are not very accurate in all applications, as they have been shown to produce biased estimates. Batch means methods can be decomposed in nonoverlapping

(NOBM) (Conway et al, 1959; Conway, 1963; Schmeiser, 1982) and overlapping (OBM) (Meketon and Schmeiser, 1984; Law and Kelton, 2000). ARMA models (Box et al, 2015) offer a comprehensive representation of a (weakly) stationary stochastic process using two polynomials, one for auto-regression (AR) and the other for the moving average. Estimation of the full correlation function is required to evaluate the variance of the sample mean, resulting in a slower approach than, for example, NOBM. Russo and Luchini (2017) proposed a Batch Means and Batch Correlation (BMBC) algorithm, inspired by batch means methods for their efficiency and from ARMA for their accuracy.

To reduce model dependencies, block bootstrap methods (Politis and White, 2004; Bernardes and Dias, 2010; Boufidi et al, 2020) have been developed. The classical bootstrap algorithm was initially proposed by Efron (1979). These methods are resampling algorithms, which allow the computation of any statistics on a given population when the probability distribution is unknown. They were then extended to account for the correlation time scale, known as the *Moving Block Bootstrap (MBB) method* proposed by Kunsch (1989). Although these methods seem very attractive and give great results for experimental data, they are too expensive (i.e., they require the storage of all samples to evaluate the statistics of interest) to be used in a numerical solver where the bottleneck is memory storage.

The present work is a first step in the development of a statistical uncertainty quantification methodology for first-order statistics (e.g., the mean velocity v or pressure p) of DNS computations. It focuses on developing unbiased estimators to quantify the expected deviation from the infinite-time-averaged statistic without any prior knowledge of the statistical process. One of the challenges of practical DNS is the processing of large data sets, which implies that only a small portion of the time series can be maintained in memory. Therefore, all statistical quantities need to be computed with cumulative approaches.

This work aims to provide reliability bounds to quantify the uncertainty of the statistical mean and the correlation time scale, which are computed with a cumulative approach, using only a limited number of samples and without any *a priori* hypotheses concerning the data. No probability distribution is inferred from the data to deduce the mean, variance, and auto-correlation. The assumptions are that the stochastic process is stationary and ergodic after a certain initial transient. Under those assumptions, multiple asymptotically unbiased estimators for the variance of the sample mean are developed and tested on three representative test problems: a generic autoregressive process, the solutions of the Kuramoto–Sivashinsky equations, and a DNS of the turbulent flow over a two-dimensional periodic hill, which features a large separation from a curved wall. Compared to existing techniques, our estimators are cumulative, work with short-duration time signals, account for the correlation, and result in little computational overhead.

The remainder of the paper is structured as follows. Section 2 is dedicated to a complete description of the statistical framework. This section also provides a quick review of the MBB method. Section 4 enumerates the three estimators for the variance of the sample mean and discusses the possibilities to accumulate those estimators to work with a limited number of samples. Section 5 present three representative test cases: the auto-regressive process, the Kuramoto–Sivashinsky equations, and the DNS of the two-dimensional periodic hill at Re_b = 10,595. Our three estimators are evaluated against the MBB and the estimator developed by Beyhaghi et al (2018), considered here as the reference. Section 5.5 addresses the memory bottleneck of the estimators using the example of the two-dimensional periodic hill. They are based on the summation of the m first terms of the auto-correlation function where m is deduced from the integral time scale $\mathcal{T}$. The idea is to subsample the discrete time series to reduce memory storage while preserving the accuracy of the estimators.

2 Stochastic processes and time series

In this section, basic concepts are recalled, including the definitions of stationary stochastic processes, time series, and confidence intervals. This section ends with a presentation of two reference methods, (i) the Moving Block Bootstrap (MBB) typically used in experiments (Boufidi, 2021) and (ii) the method developed by Beyhaghi et al (2018).

2.1 Stochastically stationary processes

Stochastic processes

A stochastic or random process is a sequence of random variables. Usually, the sequence index refers to the time. The variables composing the random process are randomly distributed according to a system of joint probability distributions. The process is statistically stationary if the system of joint probability distributions is invariant under translation. Hereafter are the definitions of the most important characteristics of these distributions.

- The mean μ defines the level around which x fluctuates and the variance σ^2 measures the spread around this level. The mean corresponds to the **expected value** of x, computed as the time average over an infinitely long period T:

$$\mu = \mathbb{E}\left[x\right] = \lim_{T \to \infty} \frac{1}{T} \int_0^T x \, \mathrm{d}t \, . \tag{2}$$

- The **variance** is also a constant and is defined as,

$$\sigma^2 = \mathbb{V}\mathrm{ar}\left(x\right) = \mathbb{E}\left[(x-\mu)^2\right] \, . \tag{3}$$

- Due to the underlying physical process, there is a correlation between two values of x at different times, which typically decreases with a delay τ between two samples, which is quantified through the **auto-covariance function** γ defined as,

$$\gamma(\tau) = \mathbb{E}\left[\left(x(t)-\mu\right)\left(x(t+\tau)-\mu\right)\right] \, . \tag{4}$$

- The **auto-correlation function** (ACF) is a dimensionless quantity and is defined as $\rho(\tau) = \gamma(\tau)/\sigma^2$. According to Pope (2000), this ACF has the following properties:

$$\rho(0) = 1 \quad \text{and} \quad |\rho(\tau)| \leq 1 \, .$$

This ACF is an even function because $\rho(-\tau) = \rho(\tau)$. If the stochastic process $x(t)$ is periodic of period T then $\rho(\tau)$ is also periodic such that $\rho(\tau + T) = \rho(\tau)$. For processes occurring in turbulent flow (such as those addressed in this paper), the correlation is expected to decrease as the time lag τ increases. Usually, the ACF diminishes sufficiently rapidly for the integral,

$$\mathcal{T} = \int_0^\infty |\rho(\tau)| \mathrm{d}\tau \tag{5}$$

to converge. Then $\mathcal{T}$ is defined as the **integral time scale**, which is a characteristic time scale for the dynamics of measured quantities.

As a consequence, the mean, variance, ACF, ... of statistically stationary processes are independent of time.

2.2 Time series, sample mean and variance

The characteristics defined in the previous section are based on the knowledge of the full evolution of the continuous process. However, due to measurement or computational resolution, we only have access to **time series** (i.e., a series of subsequent realizations at discrete time steps). Within this paper, we will consider equispaced time steps $t_i = t_0 + i \cdot \Delta t$. In the context of DNS and LES, these subsequent realizations are snapshots of pressure, velocity, vorticity, ... at (multiples of) the numerical time step Δt. The individual discrete observations are then denoted $\{x(t_0), x(t_1), \ldots, x(t_i), \ldots x(t_n)\} = \{x_0, x_1, \ldots, x_i, \ldots, x_n\}$. The goal is to provide unbiased relations between the statistical properties of the series to those of the continuous process.

The **sample mean** $\overline{x}_n$ is an estimator of the mean μ of the stochastic process. It is defined for a time series, containing n observations as

$$\overline{x}_n = \frac{1}{n} \sum_{t=1}^{n} x_t. \tag{6}$$

The sample mean is itself a stochastically distributed quantity. It is an unbiased estimator because the expected value of $\overline{x}_n$ is the mean μ of the stochastic process itself,

$$\mathbb{E}\left[\overline{x}_n\right] = \frac{1}{n}\sum_{t=1}^{n}\mathbb{E}\left[x_t\right] = \mu.$$

Hence the sample mean $\overline{x}_n$ is distributed around the actual mean for any given length n of the time series. One can show that the variance of the distribution of $\overline{x}_n$ reduces as n becomes larger, and therefore the sample mean converges to the real mean. The objective of this paper is to provide an estimate of how close the sample mean is to the true mean for a given series length n, using confidence intervals, which are introduced in the next section.

One can then also define the unbiased **sample variance** in analogy with the variance as:

$$s_n^2 = \frac{1}{n-1}\sum_{t=1}^{n}\left(x_t - \overline{x}_n\right)^2. \tag{7}$$

The sample variance is also a statistical quantity. Nevertheless, the expected value of the sample variance is not the variance, but rather

$$\mathbb{E}\left[s_n^2\right] = \sigma^2 - \frac{2}{n-1}\sum_{k=1}^{n-1}\left(1-\frac{k}{n}\right)\gamma_k = \sigma^2\left[1 - \frac{2}{n-1}\sum_{k=1}^{n-1}\left(1-\frac{k}{n}\right)\rho_k\right] \tag{8}$$

with $\gamma_k = \mathbb{E}\left[(x(t)-\mu)(x(t+\tau_k)-\mu)\right]$, the value of the auto-covariance function for a time delay $\tau_k = k\Delta t$, and the associated values of the auto-correlation function $\rho_k = \gamma_k/\sigma^2$. Unlike the sample mean, the sample variance is not a priori distributed around the continuous variance σ^2. Only if the samples are uncorrelated, then $\rho_k = 0$ and $\mathbb{E}\left[s_n^2\right] = \sigma^2$. However, if the correlation vanishes (*i.e.*if $\lim_{k\to\infty}\rho_k = 0$), we still find $\lim_{n\to\infty}\mathbb{E}\left[s_n^2\right] = \lim_{n\to\infty}s_n^2 = \sigma^2$.

2.3 Uncertainty of the sample mean

The variance of the sample mean is a measure of the uncertainty of the estimator. According to textbook definition, Equation 2.5.4 in Wei (2006), the **variance of the sample**

mean can be written as,

$$\mathbb{V}\text{ar}\left(\overline{x}_n\right) = \frac{\sigma^2}{n}\left[1 + 2\sum_{k=1}^{n-1}\left(1 - \frac{k}{n}\right)\rho_k\right] \tag{9}$$

where σ^2 and ρ_k are unknown. Assuming that the samples x_t are independently identically distributed (i.i.d), the correlation terms ρ_k are zero and therefore Equation 8 reduces to

$$\mathbb{V}\text{ar}\left(\overline{x}_n\right) = \sigma^2/n = \mathbb{E}\left[s_n^2\right]/n\,.$$

We retrieve Equation 1 described in the work of Beyhaghi et al (2018). The inverse proportionality with respect to n confirms the intuition that the average becomes more reliable as the length of the averaged series grows. This estimator can not be used as such for the time series generated by DNS, since successive time steps are highly correlated. Note that σ^2 can be written as a function of $\mathbb{V}\text{ar}\left(\overline{x}_n\right)$ and $\mathbb{E}\left[s_n^2\right]$,

$$\sigma^2 = \frac{n-1}{n}\mathbb{E}\left[s_n^2\right] + \mathbb{V}\text{ar}\left(\overline{x}_n\right)\,. \tag{10}$$

This last expression will be useful when developing our estimator for the variance of the sample mean in Section 4.

3 Reference methods

In this section, two types of estimators available in the literature for accessing the statistical uncertainty in the numerical approximation of infinite time-averaged statistics are presented in detail. The method proposed by Beyhaghi et al (2018) solves an optimization problem to fine-tune model parameters, while the Moving Block Bootstrap method is a resampling algorithm used when the data distribution is unknown. Both approaches require the full-time series to quantify the uncertainty.

3.1 Estimator proposed by Beyhaghi *et al.* [2018]

Concerning the estimation of the averaging error, their method is based on two fundamental aspects:

- the definition of the *shifted sample means* $m_{l,l+s}$ from which the *mean-squared shifted sample mean* $\overline{m}_s^2$ is computed and finally used in the objective function of the optimization problem;
- an optimization problem dedicated to the fine-tuning of the auto-correlation function, defined as:

$$\hat{\rho}(k;\hat{\theta}) = \sum_{i=1}^{m} \hat{A}_i \hat{\tau}_i^k \quad \text{where} \quad \hat{\theta} = \left[\hat{A}_1, \hat{A}_2, \ldots, \hat{A}_m, \hat{\tau}_1, \hat{\tau}_2, \ldots, \hat{\tau}_m\right] ,$$

with the two following constraints $0 \le \hat{\tau}_i \le 1$ and $\sum_{i=1}^{m} \hat{A}_i = 1$. The optimization problem then writes as,

$$\{\hat{\theta}_N, \hat{\sigma}_N, \hat{\mu}_N\} = \arg\min f(\hat{\theta}, \hat{\sigma}, \hat{\mu}) = \sum_{s=1}^{q_N} \left[g_s(\hat{\theta}, \hat{\sigma}, \hat{\mu})\right]^2 ,$$

where $g_s(\hat{\theta}, \hat{\sigma}, \hat{\mu}) = \hat{\mu}^2 + \frac{\hat{\sigma}^2}{s}\left[1 + 2\sum_{k=1}^{s-1}(1 - \frac{k}{s})\hat{\rho}(k;\hat{\theta})\right] - \overline{m}_s^2$, and $q_N = \lfloor\sqrt{N}\rfloor$.

According to their notation, N corresponds to the number of samples composing the time series, s is the number of samples used to compute the sample mean, and the subscript l varies between 0 and $N - s$. Note that no information is given about m, the number of model parameters $\hat{\theta}$. The model parameters $\{\hat{\theta}, \hat{\sigma}, \hat{\mu}\}$ are tuned to accurately match the model of the expected squared averaging error (Equation 9), at a range of different timescales s, based on the available data. Although this method gives good results when combined with the detection of the initial transient, it cannot be used in the framework of DNS or LES since the method requires the full-time history. For more information on their implementation and a detailed description of the various terms, readers can read the paper by Beyhaghi et al (2018).

3.2 Moving Block Bootstrap (MBB) method

The Bootstrap method is a resampling algorithm used when the probability distribution of the data is unknown. It allows to infer any statistics (e.g., mean, variance, ...) from a single time series. The method was originally proposed by Efron (1979) and applied when samples were independent. The idea behind the Bootstrap method is to generate B sample series constructed by randomly resampling the original time series. For each of those B new series, the statistics θ_i are computed, leading, to the generation of a population θ_B for which a mean value μ_B and a variance σ_B^2 are deduced.

The method was extended by Kunsch (1989) to preserve correlation in time series, leading to the Moving Block Bootstrap (MBB) method. Instead of randomly resampling the original time series, $N - c + 1$ random overlapping blocks of size c are chosen. Then N/c blocks are extracted and randomly concatenated into a new data series $x_{b,i}^*$ having roughly the same size as the original series. This process, illustrated in Figure 1, is repeated B times to create a population of the statistic parameters of interest. To work well, the block length c is based on the integral length scale.

Politis and White (2004) then proposed a methodology to automatically estimate the optimal block length c. Given $X_1, \ldots, X_N$ observations from a strictly stationary real-valued sequence having a mean $\mu = \mathbb{E}\left[X_t\right]$ and an auto-correlation sequence $R(s) = \mathbb{E}\left[(X_k - \mu)(X_{t+|s|} - \mu)\right]$, the block size for the circular bootstrap (or the moving block bootstrap) is the one that minimizes the large-sample MSE($\sigma_{b,CB}^2$):

$$c_{opt,CB} = \left[\left(\frac{2G^2}{D_{CB}}\right)^{1/3} N^{1/3}\right], \tag{11}$$

where $D_{CB} = (4/3)g^2(0)$, $G = \sum_{k=-\infty}^{\infty} |k|R(k)$, and g being the spectral density function defined as, $g(w) = \sum_{s=-\infty}^{\infty} R(s)\cos(ws)$. Note that both $R(s)$ and μ are unknown. In their paper, they define two estimators $\hat{g}$ and $\hat{G}$ of g and G, respectively. Those estimators care based on the *flat-top* lag window $\lambda(t)$ that has a trapezoidal shape symmetric around zero, i.e.,

$$\lambda(t) = \begin{cases} 1, & 0 \leq |t| \leq 0.5 \\ 2(1 - |t|), & 0.5 \leq |t| \leq 1 \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

From this definition, they estimate $g(w)$ by $\hat{g}(w) = \sum_{k=-M}^{M} \lambda(k/M)\hat{R}(k)\cos(wk)$. Note that the estimator of the auto-correlation is simply $\hat{R} = N^{-1}\sum_{i=1}^{N-|k|}(X_i - \overline{X}_N)(X_{i+|k|} - \overline{X}_N)$. Similarly, they get the estimate of G by $\hat{G} = \sum_{k=-M}^{M} \lambda(k/M)|k|\hat{R}(k)$. Finally, the estimator of the optimal block size is given by,

$$\hat{c}_{opt,CB} = \left[\left(\frac{2\hat{G}^2}{\hat{D}_{CB}} \right)^{1/3} N^{1/3} \right]. \tag{13}$$

The parameter M is defined as $M = 2\hat{m}$ where $\hat{m}$ is the smallest integer for which $\rho(k) = R(k)/R(0)$ becomes negligible. A more precise formulation of $\hat{m}$ is given by Politis and White (2004). The test of the implicit assumption on the auto-correlation is described as follows: $\hat{m}$ is defined as the smallest positive integer such that $|\hat{\rho}(\hat{m} + k)| < a\left\{\log(N)/N\right\}^{1/2}$, for $k = 1, \cdots, K_N$, where $a > 0$ is fixed to a constant and K_N is a positive, non-decreasing integer value function of N. The recommended values are $a = 2$ and $K_N = \max\left(5, \log(N)^{1/2}\right)$.

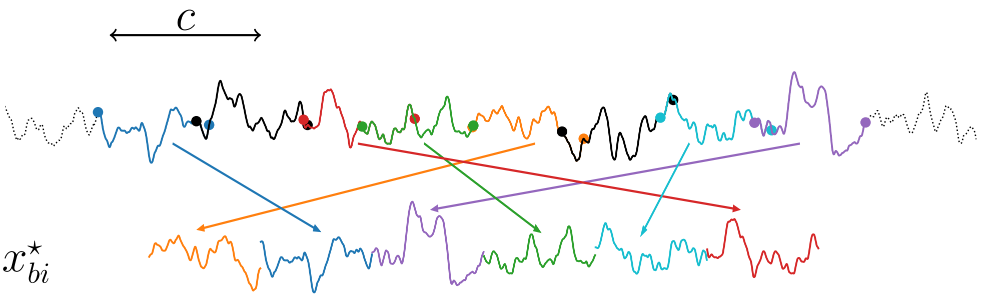

Fig. 1: Moving Blocks Bootstrap method

To summarize, the Moving Block Bootstrap method works as follows,

1. Identify the smallest $\hat{m}$ based on the implicit hypothesis test stated above;

2. Using the value $M = 2\hat{m}$, compute $\hat{G}$, $\hat{g}$, and $\hat{D}_{CB}$;
3. Estimate the optimal block length as $\hat{c}_{opt,CB}$ given by Equation 13;
4. Create $N - \hat{c}_{opt,CB} + 1$ random overlapping blocks of size $\hat{c}_{opt,CB}$ from the original time series
5. Randomly extract $N/\hat{c}_{opt,CB}$ blocks to create a new data series $x^{\star}_{i,b}$;
6. Compute the statistics of interest θ_i;
7. Repeat points 5 and 6 B times to create a population θ_B.

4 $\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)$ estimation from a limited number of samples

Due to memory storage in ongoing DNS computations, it is impossible to keep the full-time series, in particular, if we want to verify the variation of the statistical quantities over the domain. Fortunately, the sample average of a series growing in time can be computed recursively (Welford, 1962) as

$$\overline{x}_n = \frac{n-1}{n}\overline{x}_{n-1} + \frac{x_n}{n}\,. \tag{14}$$

The sample variance can be rewritten as

$$s_n^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left(x_i - \overline{x}_n\right)^2 = \frac{1}{n-1}\left(\sum_{i=1}^{n} x_i^2 - n\overline{x}_n^2\right) = \frac{1}{n-1}\sum_{i=1}^{n}\left(x_i^2 - \overline{x}_n^2\right)$$

and can therefore be computed cumulatively by either accumulating of the sum of all x_i^2, in a similar way to the sample mean, or recursively as

$$\begin{aligned}(n-1)s_n^2 &= (n-2)s_{n-1}^2 + (n-1)\overline{x}_{n-1}^2 + x_n^2 - n\overline{x}_n^2\\ \Rightarrow s_n^2 &= \frac{n-2}{n-1}s_{n-1}^2 + \frac{1}{n}\left(x_n - \overline{x}_{n-1}\right)^2\,.\end{aligned}$$

The last expression can be also found in the work of Welford (1962). Both computations do not require additional storage. However, the computation of σ^2, $\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)$ as well as $\mathcal{T}$ involve the correlation coefficient ρ_k. Therefore, estimates which can be accumulated while retaining only a small subset $\{x_{n-m}, \ldots, x_n\}$ in memory, instead of the full-time series, have to be developed.

In many problems and in the case of turbulent flow, there is a strong correlation between successive values in the time series. Therefore, an estimate for the correlations ρ_k is required. The sum expression in Equation 9 includes correlations up to $n-1$ (the total duration of the time series), and therefore, in theory, one would need to store the entire time series to compute this sum directly. Unfortunately, to reduce the memory consumption, only a limited amount of the ACF terms can be stored. Because the correlation is a summable sequence, one may define the minimal set of terms that represent this correlation. In other words, there exists an integer m such that the coefficients $\{\rho_k, \forall k \geq m\}$ become negligible. The sum can be, therefore, truncated to the m first coefficients,

$$\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right) \approx \frac{\sigma^2}{n}\left[1 + 2\sum_{k=1}^{m}\left(1 - \frac{k}{n}\right)\rho_k\right]. \tag{15}$$

This minimal set (i.e., the number of coefficients) is numerically computed based on an estimation of the integral length scale $\mathcal{T}$, which in turn requires the autocorrelation function to be evaluated according to Equation 5. Nonetheless, a large number of coefficients may have to be stored to estimate $\mathcal{T}$ correctly in practical cases. To avoid a prohibitive storage cost, it is convenient to approximate the autocorrelation function with a Gamma Exponential Function (GEF), defined as,

$$\mathrm{GEF}(r; l, \gamma) = \exp\left[-\left(\frac{r}{l}\right)^{\gamma}\right], \tag{16}$$

whose parameters l, γ can be found via Non-Linear Least Square Error regression on a reduced set of terms (e.g., Gauss-Newton). Moreover, at the beginning of the computation, only a few terms are known, and this approximation is, therefore, helpful to extrapolate the ACF. The GEF is strictly positive, which is not a general assumption for real ACFs of turbulent quantities. Therefore, other kernels can be used to approximate the integral time scale depending on the physical problem being addressed. Moreover, this approximation problem does not aim to evaluate exactly $\mathcal{T}$, but to get an approximation of it with the least amount of terms possible r_k, for $k = \{1, \cdots, K\}$ to reduce the memory storage. The constraint optimization problem can be written as,

$$\min_{l,\gamma} \sum_{k=0}^{K} \left(\mathrm{GEF}(r_k; l, \gamma) - \hat{\gamma}_k^n / \hat{\gamma}_0\right)^2 , \quad \text{subjected to } \gamma > 0 , \tag{17}$$

where $\hat{\gamma}_k^n$ by Equation 19, and K is the number of correlations coefficients used to performed the optimization procedure. This optimisation problem can be solved using the steepest descent method or the Gauss-Newton method. Once the optimization problem is solved, the number of coefficients m is determined as the smallest integer for which the fitted Gamma Exponential function drops below a given threshold:

$$m = \arg\min_{k} \left\{ \left| \mathrm{GEF}(r_k; l^\star, \gamma^\star) - \epsilon_{thr} \right| \right\} , \tag{18}$$

where the threshold ϵ_{thr} is set to 5%. Note that as a co-processing tool, this integral time scale is evaluated within the DNS simulation code and adapted at each time step.

4.1 Short overview of covariance estimation

Many auto-covariance estimators can be found in the literature. First of all, the *classical* autocovariance estimator

$$\hat{\gamma}_k^n = \frac{1}{n-1} \sum_{t=1}^{n-k} (x_t - \overline{x}_n)(x_{t+k} - \overline{x}_n) \tag{19}$$

which is based on the method of moments, is mentioned in most textbooks on (time) series analysis, see *e.g.* Box et al (2016). The denominator is set to $n-1$ (instead of $n-k$ in standard expression) to ensure positive definiteness of the sampling autocovariance matrix (Box et al, 2016; Dürre et al, 2015) at the cost of increased bias. Robust ACF estimators that remain close to the true underlying ACF, even when outliers are present in the time series were presented by Ma and Genton (2000) and Dürre et al (2015). Liao et al (2016) revisited the autocovariance function estimate as a constrained penalized regression. Vogelsang and Yang (2016) constructed the estimator as a linear combination of population autocovariances and they show that it drastically reduces the bias. Andersen et al (2001) proposed quadratic variations as an estimator for the variance and covariance that are used for non-synchronous data (Hayashi and Yoshida, 2005).

4.2 $\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)$ estimator based on classical ACF estimator

The expectation of the classical ACF estimator $\hat{\gamma}_k^n$ (see Equation 19) is found to be

$$
\begin{aligned}
\mathbb{E}\left[\hat{\gamma}_k^n\right] &= \frac{1}{n-1}\mathbb{E}\left[\sum_{t=1}^{n-k}(x_t-\overline{x}_n)(x_{t+k}-\overline{x}_n)\right]\\
&= \frac{1}{n-1}\mathbb{E}\left[\sum_{t=1}^{n-k}[(x_t-\mu)-(\overline{x}_n-\mu)][(x_{t+k}-\mu)-(\overline{x}_n-\mu)]\right]\\
&= \frac{1}{n-1}\mathbb{E}\left[\sum_{t=1}^{n-k}(x_t-\mu)(x_{t+k}-\mu)\right]+\frac{1}{n-1}\mathbb{E}\left[\sum_{t=1}^{n-k}(\overline{x}_n-\mu)^2\right]\\
&-\frac{1}{n-1}\mathbb{E}\left[\sum_{t=1}^{n-k}(\overline{x}_n-\mu)(x_{t+k}-\mu)\right]-\frac{1}{n-1}\mathbb{E}\left[\sum_{t=1}^{n-k}(x_t-\mu)(\overline{x}_n-\mu)\right]\\
&= \frac{n-k}{n-1}\gamma_k-\frac{n+k}{n-1}\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)+\frac{1}{n-1}\mathbb{E}\left[(\overline{x}_n-\mu)\left(\sum_{t=n-k+1}^{n}(x_t-\mu)+\sum_{t=1}^{k}(x_t-\mu)\right)\right]\\
&= \frac{n-k}{n-1}\gamma_k-\frac{n+k}{n-1}\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)+\frac{2k}{n-1}\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)\\
&= \frac{n-k}{n-1}\left(\gamma_k-\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)\right)
\end{aligned}
$$

To find γ_k as a function of $\mathbb{E}\left[\hat{\gamma}_k^n\right]$ and of $\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)$,

$$
\gamma_k=\frac{n-1}{n-k}\mathbb{E}\left[\hat{\gamma}_k^n\right]+\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right).
$$

This expression is injected in Equation 15 to get,

$$
\begin{aligned}
\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right) &\approx \frac{\sigma^2}{n}+\frac{2}{n^2}\sum_{k=1}^{m}(n-k)\Big(\frac{n-1}{n-k}\mathbb{E}\left[\hat{\gamma}_k^n\right]+\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)\Big)\\
&= \frac{\sigma^2}{n}+\frac{2(n-1)}{n^2}\sum_{k=1}^{m}\mathbb{E}\left[\hat{\gamma}_k^n\right]+\frac{2}{n^2}\Big(nm-\frac{m(m+1)}{2}\Big)\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)\\
&= \frac{\sigma^2}{n}+\frac{m(2n-m-1)}{n^2}\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)+\frac{2(n-1)}{n^2}\sum_{k=1}^{m}\mathbb{E}\left[\hat{\gamma}_k^n\right].
\end{aligned}
$$

In this last expression, σ^2 is replaced by Equation 10 to obtain,

$$\begin{aligned}
\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right) &\approx \frac{1}{n}\Big(\frac{n-1}{n}\mathbb{E}\left[s_n^2\right] + \mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)\Big) + \frac{m(2n-m-1)}{n^2}\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right) + \frac{2(n-1)}{n^2}\sum_{k=1}^{m}\mathbb{E}\left[\hat{\gamma}_k^n\right] \\
&= \frac{n-1}{n^2}\mathbb{E}\left[s_n^2\right] + \Big(\frac{1}{n} + \frac{m(2n-m-1)}{n^2}\Big)\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right) + \frac{2(n-1)}{n^2}\sum_{k=1}^{m}\mathbb{E}\left[\hat{\gamma}_k^n\right] \\
&= \frac{n-1}{n^2}\mathbb{E}\left[s_n^2\right] + \frac{n(2m+1)-m(m+1)}{n^2}\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right) + \frac{2(n-1)}{n^2}\sum_{k=1}^{m}\mathbb{E}\left[\hat{\gamma}_k^n\right] \\
\Rightarrow \mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)&\Big(1 - \frac{n(2m+1)-m(m+1)}{n^2}\Big) \approx \frac{n-1}{n^2}\mathbb{E}\left[s_n^2\right] + \frac{2(n-1)}{n^2}\sum_{k=1}^{m}\mathbb{E}\left[\hat{\gamma}_k^n\right].
\end{aligned}$$

Using a direct estimator for the auto-correlation, a first estimator for the variance of the sample mean, noted as $\hat{\mathcal{V}}_n^{(1)}$, is obtained,

$$\boxed{\hat{\mathcal{V}}_n^{(1)} = a\left(s_n^2 + 2\sum_{k=1}^{m}\hat{\gamma}_k^n\right),} \tag{20}$$

where $a = (n-1)\left((m-n)^2 + (m-n)\right)^{-1}$. From this last expression we can write that $\mathbb{E}\left[\hat{\mathcal{V}}_n^{(1)}\right] = \mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)$, which confirms that the estimator is (asymptotically) unbiased by construction.

The standard ACF estimator cannot be evaluated with a *standard* cumulative method. However, it can be rewritten in three distinct terms to facilitate its evaluation:

$$(n-1)\hat{\gamma}_k^n = \underbrace{\sum_{t=1}^{n-k} x_t x_{t+k}}_{=\hat{\gamma}_{k,1}^n} - \overline{x}_n \underbrace{\sum_{t=1}^{n-k}(x_t + x_{t+k})}_{=\hat{\gamma}_{k,2}^n} + (n-k)\overline{x}_n^2\,.$$

The *accumulation* is performed on those three terms, with the k previous terms $\{x_{n+1-k}, \ldots, x_{n+1}\}$, $\hat{\gamma}_{k,1}^{n+1} = \hat{\gamma}_{k,1}^n + x_{n+1-k}x_{n+1}$, and $\hat{\gamma}_{k,2}^{n+1} = \hat{\gamma}_{k,2}^n + (x_{n+1-k} + x_{n+1})$, as

$$n\hat{\gamma}_k^{n+1} = \hat{\gamma}_{k,1}^{n+1} + \overline{x}_{n+1}^2\hat{\gamma}_{k,2}^{n+1} + (n+1-k)\overline{x}_{n+1}^2\,,$$

where $\overline{x}_{n+1}$ is given by Equation 14.

4.3 $Var(\overline{x}_n)$ estimator based on quadratic variation estimator

For comparison purposes, the quadratic variation estimator, initially proposed by Andersen et al (2001) to construct estimates of daily exchange rate volatility in a financial context, is also considered with

$$\hat{\delta}_k^n = \frac{1}{n}\sum_{t=1}^{n-k}(x_t - x_{t+k})^2 \,. \tag{21}$$

Similarly to Section 4.2, the expectation of this estimator is computed,

$$\begin{aligned}
\mathbb{E}\left[\hat{\delta}_k^n\right] &= \frac{1}{n}\mathbb{E}\left[\sum_{t=1}^{n-k}(x_t - x_{t+k})^2\right] \\
&= \frac{1}{n}\mathbb{E}\left[\sum_{t=1}^{n-k}((x_t-\mu)-(x_{t+k}-\mu))^2\right] \\
&= \frac{1}{n}\mathbb{E}\left[\sum_{t=1}^{n-k}(x_t-\mu)^2\right] - \frac{2}{n}\mathbb{E}\left[\sum_{t=1}^{n-k}(x_t-\mu)(x_{t+k}-\mu)\right] + \frac{1}{n}\mathbb{E}\left[\sum_{t=1}^{n-k}(x_{t+k}-\mu)^2\right] \\
&= 2\left(1-\frac{k}{n}\right)\left(\sigma^2-\gamma_k\right) = 2\sigma^2\left(1-\frac{k}{n}\right)\left(1-\rho_k\right)\,.
\end{aligned}$$

This estimator cannot be computed using a *standard* cumulative method. The following accumulation process can be set up using the k previous terms $\{x_{n+1-k},\ldots,x_{n+1}\}$,

$$\hat{\delta}_k^{n+1} = \frac{1}{n+1}\left(n\hat{\delta}_k^n + (x_{n+1-k}-x_{n+1})^2\right)\,. \tag{22}$$

Note that a series of $\{x_{n+1-k},\ldots,x_{n+1}\}$ terms need to be retained in memory. This number of terms is limited due to the truncated sum defined in Equation 15. Moreover, it will be further reduced by undersampling. This technique is discussed in Section 5.5. Based on this estimator, an expression for the variance of the mean is established. First, γ_k is expressed as a function of σ^2 and $\mathbb{E}\left[\hat{\delta}_k^n\right]$ to get,

$$2\left(1-\frac{k}{n}\right)\gamma_k = 2\left(1-\frac{k}{n}\right)\sigma^2 - \mathbb{E}\left[\hat{\delta}_k^n\right]\,,$$

and this result is injected in Equation 15, to get,

$$\begin{aligned}
\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right) &\approx \frac{\sigma^2}{n} + \frac{1}{n}\sum_{k=1}^{m}\left(2\left(1-\frac{k}{n}\right)\sigma^2 - \mathbb{E}\left[\hat{\delta}_k^n\right]\right) \\
&= \frac{\sigma^2}{n} + \frac{1}{n^2}\sum_{k=1}^{m}2(n-k)\sigma^2 - \frac{1}{n}\sum_{k=1}^{m}\mathbb{E}\left[\hat{\delta}_k^n\right] \\
&= \frac{\sigma^2}{n} + \frac{2}{n^2}\left(nm - \frac{m(m+1)}{2}\right)\sigma^2 - \frac{1}{n}\sum_{k=1}^{m}\mathbb{E}\left[\hat{\delta}_k^n\right] \\
&= \frac{n+2nm-m(m+1)}{n^2}\sigma^2 - \frac{1}{n}\sum_{k=1}^{m}\mathbb{E}\left[\hat{\delta}_k^n\right] \\
&= \frac{n^2-(n-m)^2+n-m}{n^2}\sigma^2 - \frac{1}{n}\sum_{k=1}^{m}\mathbb{E}\left[\hat{\delta}_k^n\right].
\end{aligned}$$

In this last expression, σ^2 is replaced by Equation 10, to finally obtain,

$$\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right) \approx \frac{n^2-(n-m)^2+n-m}{n^2}\left(\frac{n-1}{n}\mathbb{E}\left[s_n^2\right] + \mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)\right) - \frac{1}{n}\sum_{k=1}^{m}\mathbb{E}\left[\hat{\delta}_k^n\right] \tag{23}$$

$$\Rightarrow \boxed{\hat{\mathcal{V}}_n^{(2)} = \frac{1}{n(1-a)}\left(a(n-1)s_n^2 - \sum_{k=1}^{m}\hat{\delta}_k^n\right)} \tag{24}$$

where $a = (n^2-(n-m)^2+n-m)(n^2)^{-1}$. From this last expression we can write that $\mathbb{E}\left[\hat{\mathcal{V}}_n^{(2)}\right] = \mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)$, which confirms that the estimator is (asymptotically) unbiased by construction.

4.4 $\mathbb{V}\mathrm{ar}\left(\overline{x}_n\right)$ estimator based on non-centered classical ACF estimator

To construct this third estimator, the sample mean is removed from the direct estimator of the auto-correlation treated in Section 4.2 and a factor of $1/(n-k)$ is used instead of $1/(n-1)$ used in Equation 19. This third estimator is defined as

$$\hat{\varphi}_k^n = \frac{1}{n-k}\sum_{t=1}^{n-k}x_t x_{t+k}. \tag{25}$$

This estimator can be rewritten in a *cumulative form*, using the k previous terms $\{x_{n+1-k}, \ldots, x_{n+1}\}$,

$$\hat{\varphi}_k^{n+1} = \frac{(n-k)\hat{\varphi}_k^n + x_{n+1-k}x_{n+1}}{n+1-k}$$

Once the number of coefficients m is known, this third estimator can be evaluated using a cumulative method. Only the last m terms of the time series need to be kept in memory. Once again, the expectation of this estimator is evaluated as

$$\begin{aligned}
\mathbb{E}\left[\hat{\varphi}_k^n\right] &= \frac{1}{n-k}\mathbb{E}\left[\sum_{k=1}^{n-k} x_t x_{t+k}\right] \\
&= \frac{1}{n-k}\mathbb{E}\left[\sum_{k=1}^{n-k}(x_t-\mu+\mu)(x_{t+k}-\mu+\mu)\right] \\
&= \frac{1}{n-k}\mathbb{E}\left[\sum_{k=1}^{n-k}(x_t-\mu)(x_{t+k}-\mu)\right] + \frac{1}{n-k}\mathbb{E}\left[\sum_{k=1}^{n-k}\mu^2\right] \\
&+ \frac{1}{n-k}\mathbb{E}\left[\sum_{k=1}^{n-k}\mu(x_{t+k}-\mu)\right] + \frac{1}{n-k}\mathbb{E}\left[\sum_{k=1}^{n-k}\mu(x_t-\mu)\right] \\
&= \gamma_k + \mu^2 .
\end{aligned}$$

From above, an expression of the ACF is obtained,

$$\gamma_k = \mathbb{E}\left[\hat{\varphi}_k^n\right] - \mu^2 . \tag{26}$$

Note that an estimator of the square of the mean is required, since $(\mathbb{E}[\overline{x}])^2 \neq \mathbb{E}[\overline{x}^2]$. The expectation of the square of the sample mean gives an expression for the estimator,

$$\mathbb{E}\left[\overline{x}_n^2\right] = \mathbb{V}\text{ar}\left(\overline{x}_n\right) + \mu^2 \Rightarrow \mu^2 = \mathbb{E}\left[\overline{x}_n^2\right] - \mathbb{V}\text{ar}\left(\overline{x}_n\right) . \tag{27}$$

In Equation 26, μ^2 is substituted by the expression reported in Equation 27, and the result is injected in Equation 15, to get

$$\begin{aligned}
\mathbb{V}\text{ar}\left(\overline{x}_n\right) &\approx \frac{\sigma^2}{n} + \frac{2}{n}\sum_{k=1}^{m}\Big(1-\frac{k}{n}\Big)\Big(\mathbb{E}\left[\hat{\varphi}_k^n\right] - \mathbb{E}\left[\overline{x}_n^2\right] + \mathbb{V}\text{ar}\left(\overline{x}_n\right)\Big) \\
&= \frac{\sigma^2}{n} + \frac{2}{n}\sum_{k=1}^{m}\Big(1-\frac{k}{n}\Big)\mathbb{E}\left[\hat{\varphi}_k^n\right] - \frac{2}{n^2}\Big(nm - \frac{m(m+1)}{2}\Big)\Big(\mathbb{E}\left[\overline{x}_n^2\right] - \mathbb{V}\text{ar}\left(\overline{x}_n\right)\Big).
\end{aligned}$$

Finally, σ^2 is replaced by Equation 10 to get,

$$\begin{aligned}\mathbb{V}\text{ar}\left(\overline{x}_n\right)\left(1-\frac{1}{n}\right) &\approx \frac{n-1}{n^2}\mathbb{E}\left[s_n^2\right]+\frac{2}{n}\sum_{k=1}^{m}\left(1-\frac{k}{n}\right)\mathbb{E}\left[\hat{\varphi}_k^n\right] \\ &\quad -\frac{2nm-m(m+1)}{n^2}\left(\mathbb{E}\left[\overline{x}_n^2\right]-\mathbb{V}\text{ar}\left(\overline{x}_n\right)\right)\end{aligned} \tag{28}$$

$$\Rightarrow \boxed{\hat{\mathcal{V}}_n^{(3)} = a\left((n-1)s_n^2+2\sum_{k=1}^{m}(n-k)\hat{\varphi}_k^n+b\overline{x}_n^2\right),} \tag{29}$$

where $a=[(n-m)^2-n+m]^{-1}$, and $b=m(m-2n+1)$. From this last expression we can write that $\mathbb{E}\left[\hat{\mathcal{V}}_n^{(3)}\right]=\mathbb{V}\text{ar}\left(\overline{x}_n\right)$, which confirms that the estimator is (asymptotically) unbiased by construction.

4.5 $\mathbb{V}$ar $(\overline{x}_n)$ estimator based on a smooth approximation of the ACF

The standard estimator of the ACF, used in Section 4.2, is known to suffer from spurious oscillations at large time lags, making it difficult to understand at which lag the ACF goes to zero. For this reason, the truncation m is approximated using an optimization procedure that aims to fit a smooth estimator of the ACF, i.e., the Gamma-Exponential Function (Equation 16), on the standard estimator of the ACF. This procedure is summarized by the equations 17 and 18. A fourth estimator $\hat{\mathcal{V}}_n^{(4)}$ is derived by replacing, in Equation 9, σ^2 by its estimator s_n^2 and γ_k by the smooth approximation of the ACF, to get

$$\boxed{\hat{\mathcal{V}}_n^{(4)} = \frac{s_n^2}{n}\left[1+2\sum_{k=1}^{n-1}\left(1-\frac{k}{n}\right)\text{GEF}(r_k;l^*,\gamma^*)\right],} \tag{30}$$

This new estimator is not based on the definition of m and therefore the sum over the correlation coefficients is no more truncated. Compared to the other three estimators, $\hat{\mathcal{V}}_n^{(4)}$ is not asymptotically unbiased by construction. This last estimator is very close to the technique used in the paper of Rezaeiravesh et al (2023), except that the GEF is replaced by a convex combination of exponential functions, where three parameters are required to fit the model to the ACF.

4.6 Summary

Note that these four estimators for the variance of the sample mean are justified by the generalization of the Central Limit Theorem, as described in Oliver et al (2014). Using the four estimators: $\hat{\gamma}_k$, $\hat{\delta}_k$, $\hat{\psi}_k$, and $\mathrm{GEF}(r_k; l, \gamma)$ defined respectively by Equations 19, 21, 25, and 16 combined with the expressions listed in Equations 8 and 9, estimators $\hat{\mathcal{V}}_n^{(1)}$, $\hat{\mathcal{V}}_n^{(2)}$, and $\hat{\mathcal{V}}_n^{(3)}$ for the variance of the sample mean are constructed to be asymptotically unbiased, while estimator $\hat{\mathcal{V}}_n^{(4)}$ is not, but is even lighter in memory because it only requires the knowledge of two parameters l and γ. Their expressions are summarized in Table 1. Note that the constant m is evaluated though the optimization procedure defined in Equation 18. A parallel can be drawn with the definition of the optimal block length in the MBB method.

Our asymptotically unbiased estimators differ slightly from the one proposed very recently in Rezaeiravesh et al (2023). In their work, they chose not to truncate the sum over the ACF coefficients because the standard estimator of the ACF is prone to non-vanishing oscillations at higher lags. Instead, the authors used a modelled ACF that is smooth for all lags. In their paper, they went one step further and has already implemented their estimator in the flow solver, Nek5000. Except for the adjustment of the ACF, the development of the cumulative approach for mean and variance is similar to that proposed by in the present paper (see Table 1).

Table 1: Estimators for the variance of the sample mean

	Estimator 1	**Estimator 2**	**Estimator 3**	**Estimator 4**
Acc. data	$\overline{x}_n = \frac{n-1}{n}\overline{x}_{n-1} + \frac{x_n}{n}$ $s_n^2 = \frac{n-2}{n-1}s_{n-1}^2 + \frac{1}{n}(x_n - \overline{x}_{n-1})^2$ $\hat{\gamma}_{k,1}^n = \hat{\gamma}_{k,1}^{n-1} + x_{n-k}x_n$ $\hat{\gamma}_{k,2}^n = \hat{\gamma}_{k,2}^{n-1} + (x_{n-k} + x_n)$ $(n-1)\hat{\gamma}_k^n = \hat{\gamma}_{k,1}^n + \overline{x}_n^2\hat{\gamma}_{k,2}^n + (n-k)\overline{x}_n^2\,,$	$\overline{x}_n = \frac{n-1}{n}\overline{x}_{n-1} + \frac{x_n}{n}$ $s_n^2 = \frac{n-2}{n-1}s_{n-1}^2 + \frac{1}{n}(x_n - \overline{x}_{n-1})^2$ $\hat{\delta}_k^n = \frac{n-1}{n}\hat{\delta}_k^{n-1} + \frac{(x_{n-k} - x_n)^2}{n}$	$\overline{x}_n = \frac{n-1}{n}\overline{x}_{n-1} + \frac{x_n}{n}$ $s_n^2 = \frac{n-2}{n-1}s_{n-1}^2 + \frac{1}{n}(x_n - \overline{x}_{n-1})^2$ $\hat{\varphi}_k^n = \frac{n-1-k}{n-k}\hat{\varphi}_k^{n-1} + \frac{x_{n-k}x_n}{n-k}$	$\overline{x}_n = \frac{n-1}{n}\overline{x}_{n-1} + \frac{x_n}{n}$ $s_n^2 = \frac{n-2}{n-1}s_{n-1}^2 + \frac{1}{n}(x_n - \overline{x}_{n-1})^2$ -
Optim.	Find m with Eq. 18	Find m with Eq. 18	Find m with Eq. 18	Find l^* and γ^* with Eq. 17
$\hat{\boldsymbol{\mathcal{V}}}_{\boldsymbol{n}}$	$= a\left(s_n^2 + 2\sum_{k=1}^{m}\hat{\gamma}_k^n\right)$	$= \frac{a(n-1)s_n^2 - \sum_{k=1}^m \hat{\delta}_k^n}{n(1-a)}$	$= a\Big[(n-1)s_n^2 + 2\sum_{k=1}^{m}(n-k)\hat{\varphi}_k^n + b\overline{x}_n^2\Big]$	$= \frac{s_n^2}{n}\Big[1 + 2\sum_{k=1}^{n-1}\left(1 - \frac{k}{n}\right) \mathrm{GEF}(r_k; l^*, \gamma^*)\Big]$
Const.	$a = \frac{n-1}{(m-n)^2 + (m-n)}$	$a = \frac{n^2 - (n-m)^2 + n - m}{n^2}$	$a = \left[(n-m)^2 + m - n\right]^{-1}\,,$ $b = m(m - 2n + 1)$	-
Storage	$\{x_{n-m}, \dots, x_n\}$ $\{\overline{x}_n, s_n^2, \hat{\gamma}_{1,i}^n, \dots \hat{\gamma}_{m,i}^n\}$	$\{x_{n-m}, \dots, x_n\}$ $\{\overline{x}_n, s_n^2, \hat{\delta}_1^n, \dots \hat{\delta}_m^n\}$	$\{x_{n-m}, \dots, x_n\}$ $\{\overline{x}_n, s_n^2, \hat{\varphi}_1^n, \dots \hat{\varphi}_m^n\}$	$\{x_{n-m}, \dots, x_n\}$ $\{\overline{x}_n, s_n^2, \gamma^*, l^*\} + r_{\leq m}$ ACF coef.
Mem. cost	$3m + 2$	$2m + 2$	$2m + 2$	$r + m + 2 + 2 \rightarrow 4$ (as the fitting converges)

5 Applications

This section is devoted to the evaluation of our three estimators on three representative test cases: an auto-regressive process (in Section 5.1), the Kuramoto-Sivashinsky (KS) equations (in Section 5.2), and the two-dimensional periodic hill at $Re_b = 10,595$ (in Section 5.4). For the auto-regressive process, the three estimators will be compared to the exact evaluation of the confidence interval. Indeed, an analytical expression of ρ_k can be computed using the Yule-Walker equations (see *e.g.*Box et al (2016)) from which the variance of the sample mean can be exactly evaluated. For the two other test cases, since analytical expressions of μ, σ^2 and ρ_k are not available, the method developed by Beyhaghi et al (2018) and the MBB method (presented in Section 3.2) are considered as the reference to validate our three estimators. Note that in our MBB method, the MatLab function `opt_block_length_REV_dec07`, implementing the optimal block length of Politis and White (2004) (found here) has been revisited in Python.

5.1 Autoregressive process

A first validation is performed on an auto-regressive process AR(p), which is a synthetic model for a stochastic process with known mean, variance and correlation function. It is constructed by computing a new sample value x_i as a finite weighted sum of the p previous sample values plus a white noise:

$$x_i = \sum_{j=1}^{p} \alpha_p x_{i-j} + \epsilon_i \,, \tag{31}$$

where α_j are real coefficients and ϵ_i is a sample of a white Gaussian noise, i.e., $\epsilon_i \sim \mathcal{N}\left(0, \sigma_\epsilon^2\right)$. The resulting system is statistically stationary, with a zero mean, after a "*certain*" initial transient, due to the initialization of the first p value of the process. The coefficients of the auto-covariance function are obtained with the Yule-Walker equations described in Box et al (2016) (chapter *Autoregressive Processes*).

An AR(6) process is considered. The same α_j coefficients in Beyhaghi et al (2018) are used: $\alpha_1 = 3.1378$, $\alpha_2 = -3.9789$, $\alpha_3 = 2.6788$, $\alpha_4 = -1.0401$, $\alpha_5 = 0.2139$, $\alpha_6 =$

-0.0133. The simplest initialization ($x_{-5} = \cdots = x_0 = 0$) has no initial transient.

The optimization process presented in Eq. 17 is sensitive to the number of coefficients K. Figure 2 shows that the convergence to stable l^* and g^* can be slow. In the present case, the optimal values of l and γ become independent of the number of ACF coefficients for $K \geq 100$. Of course, this threshold depends on the selected test cases. Therefore, an automatic procedure must be set up. The optimal block length of the MBB method (see Section 3.2) is used because it provides a block length that should be larger than the integral length scale. The method also requires an ACF, which is replaced by a smooth approximation (i.e., the GEF). This method converges in two steps, represented by the red dots in Figure 2. The different steps are described in Algorithm 1.

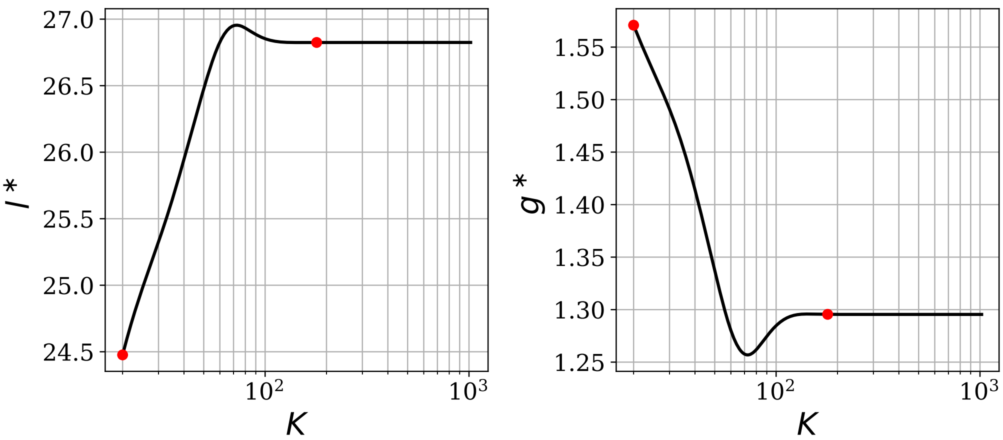

Fig. 2: Evolution of the parameters l and γ of the Gamma Exponential function with the number of coefficients K; black line indicates the evolution of the parameters while increasing linearly the number of coefficients K; red points indicates the evolution using the optimal block length size of the MBB method 3.2

Figure 3 shows the evolution of the value of m (i.e., the size of the truncated sum). Its value seems to converge as the number of samples in the series increases. Looking at the correlation on the left, the number of non-negligible coefficients is approximately $60 - 80$ which is in agreement with the evaluation of m using the procedure 18 that uses a threshold of 5%.

Algorithm 1 Optimize the parameters l and γ of the GEF using the optimal block length described in Section 3.2

```
Knowing $\hat{\gamma}_k$
Initialize $l \leftarrow 20$, and $\gamma \leftarrow 1$
Optimize $l^*$ and $\gamma^*$ via Equation 17 with $K = 20$
Compute $B^\star$ via MBB method (Section 3.2) with GEF$(l^*, \gamma^*)$
for $i = 1, \cdots, 10$ do
    Optimize $l^*$ and $\gamma^*$ via Equation 17 with $K = B^\star$
    Compute $B^\star_{new}$ via MBB method (Section 3.2) with GEF$(l^*, \gamma^*)$
    if $B^\star == B^\star_{new}$ then
        Stop, $l^*$ and $\gamma^*$ are obtained
    else
        $B^* \leftarrow B^\star_{new}$
    end if
end for
```

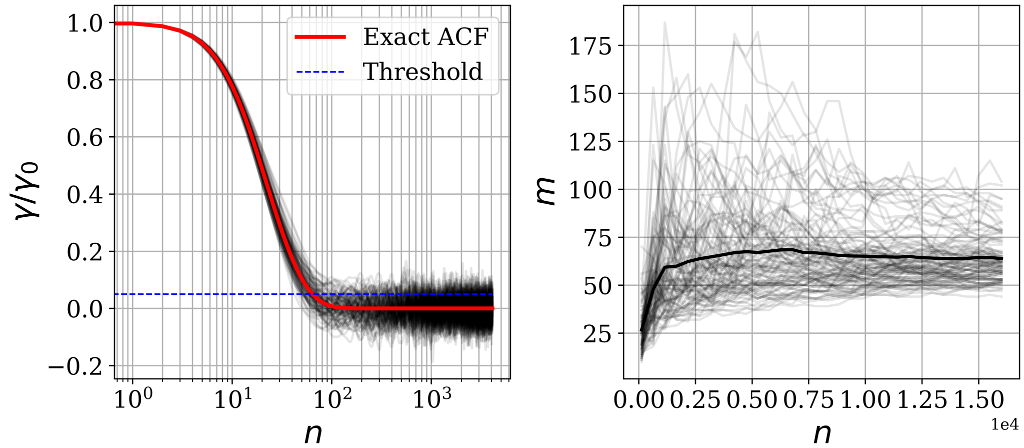

Fig. 3: Evolution of m (the size of the truncated sum) as a function of n

Figure 4 shows the evolution of the confidence interval size with increasing realization length, taken between 2^7 and 2^{14}, on an ensemble of $B = 100$ distinct time series. Since B AR(6) processes are generated, we have obtained B confidence intervals for each n value. Therefore, for each n value, the variance over the confidence intervals is evaluated to plot the error bar in Figure 4. For a fair comparison, the exact size of the confidence interval, computed with the analytical expression of μ, σ^2, and ρ_k, is drawn in a red dotted line. The basic estimator (Equation 1) is also plotted in the insert graph. Note how this estimator underestimates the true statistical error by neglecting the correlation between consecutive samples. Our four estimators behave similarly and are asymptotically unbiased. Even estimator 4, which is based on the smooth approximation of the ACF, is unbiased in this example. For the largest number of samples of 2^{14}, the relative error with respect to the exact confidence interval is approximately 0.2−0.3%. One can observe that the MBB method slightly underestimates the exact confidence interval size with a relative error of 5.2% at $n = 2^{14}$. However, increasing the number of observations reduces the relative error to only 1%. The estimator of the variance of the sample mean provided by the MBB method is not biased. It just converges more slowly than our estimators. The main advantage of the estimator proposed by Beyhaghi et al (2018) lies in the optimization of the autocorrelation function based on the available data, which leads to good estimates even for a small n. However, the computation of its estimator is expensive. Therefore, for Beyhaghi et al (2018)'s estimator, a population of only 30 time series is used instead of 100. We conclude that the four estimators presented in the present study agree well with Beyhaghi et al (2018)'s at moderate and large n, and they converge faster than the MBB.

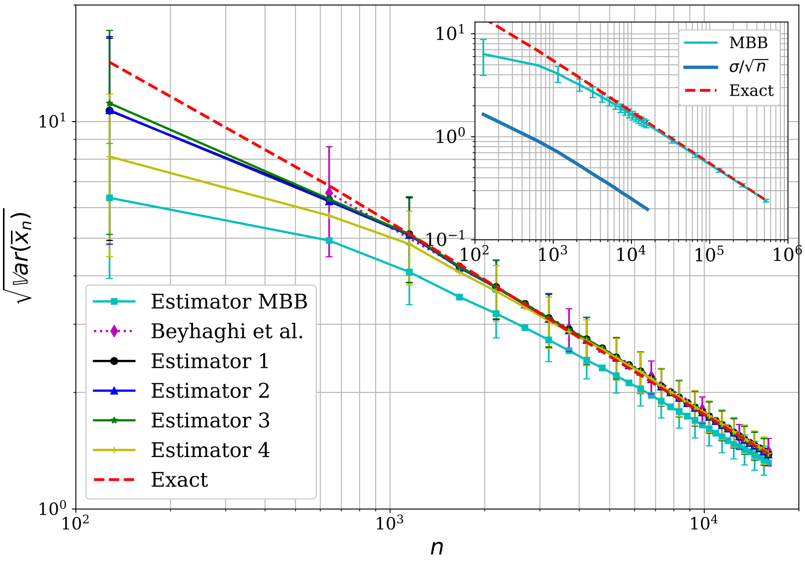

Fig. 4: Evaluation of the four estimators (see Table 1) and of the MBB method on 100 realizations of an AR(6) process; Beyhaghi et al (2018)'s estimator on 30 realization of the AR(6) process (magenta diamond dotted line); Ensemble average (plain line), ensemble variance of the confidence interval size (error bars) and exact evaluation (red dotted line)

5.2 Synthetic chaotic solutions

The Kuramoto-Sivashinsky (KS) equation, given by Equation 32, is a one-dimensional fourth-order nonlinear partial differential equation. This equation is originally derived to model the diffusive-thermal instabilities in a laminar flame front by Kuramoto (1978). It is the simplest one-dimensional equation that generates space-time chaos. The solutions, presented in Figure 5, contain rich dynamical characteristics. Indeed, on a periodic domain, a series of bifurcations strongly affects the dynamics, and eventually triggers the onset of chaotic behavior. This test case is a first step towards the analysis of more complex turbulent flow cases. Moreover, Beyhaghi et al (2018) examines the same test case, which allows a better comparison with our estimators.

$$u_t + uu_x + u_{xx} + u_{xxxx} = 0 \quad \text{for} \quad 0 \leq x/L \leq 1\,, \tag{32}$$

The equation 32 is closed by imposing periodic boundary conditions at both extremities of the one-dimensional domain. This periodicity allows the spatial derivatives to be computed in the Fourier domain. The solutions of this system are therefore computed using a pseudo-spectral method combined with a fourth-order Runge-Kutta scheme for the time integration. The initial condition is given by

$$u(x,0) = \sin\left(\frac{\pi x}{2}\right) + \sin\left(\frac{85\pi x}{100}\right) + 0.2\epsilon, \quad \epsilon \sim \mathcal{N}(0,1).$$

The domain size is set to $L = 200$ with a resolution of $N_x = 512$. The dimensionless time step $L\Delta t/\sigma_0$ is set to 0.25, where $\sigma_0 = \mathbb{V}\mathrm{ar}\,(u(0,0))$. The statistic of interest is the spatially averaged energy, defined as

$$k = \frac{1}{L}\int_0^L \frac{u^2}{2}\mathrm{d}x.$$

A single realization of the stochastic process is available for the present test case. Moreover, there is no analytical expression for μ, σ^2, and ρ_k. The MBB method and the estimator of Beyhaghi et al (2018) are considered as the references. The number of coefficients m is evaluated with Algorithm 1. Figure 6 illustrates the evolution of the confidence

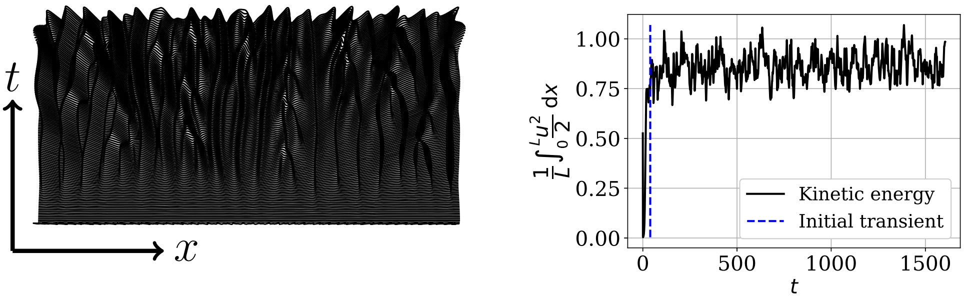

Fig. 5: (Left) Three-dimensional time-space graph of the solution obtained by solving the KS equation. (Right) Kinetic energy evolution of the Kuramoto-Sivashinsky equation with the position of the initial transient.

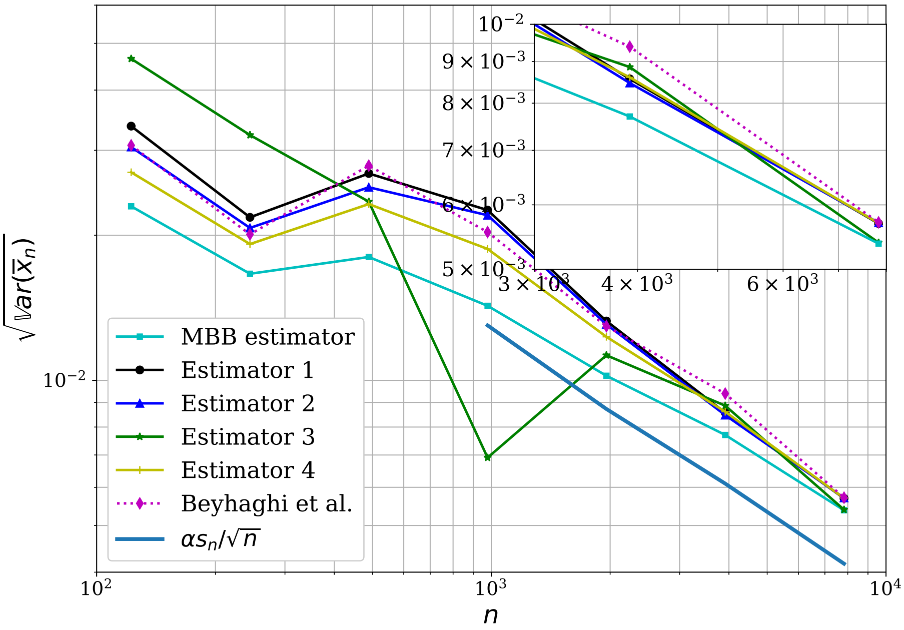

Fig. 6: Evaluation of the four estimators (see Table 1), of the MBB method, and of the estimator proposed by Beyhaghi et al (2018) on the mean kinetic energy measured in a simulation of the KS model. The basic estimator (Equation 1) is drawn in dark blue with a coefficient $\alpha = 5.5$ to make it visible in the graph.

interval size with the number of samples n in the time series. Estimators 1, 2, and 4 present a similar behavior, as they mainly overlap at every sample number n. These three estimates closely follow Beyhaghi et al (2018)'s estimate, deviating from it by less than 1% at the largest n value. Estimator 3 shows a different behavior, oscillating at lower values of n.

5.3 LES of a turbulent channel flow

The present section addresses a more realistic application, namely the estimation of the statistical error of flow data obtained by Large Eddy Simulation on a turbulent channel flow at the friction Reynolds number Re_τ of 950. The flow field is computed using the in-house flow solver Argo-DG (Hillewaert, 2013; Carton de Wiart et al, 2014), developed at Cenaero. The wrLES turbulent channel flow data are extracted from Argo-DG on structured probes. This test case is identical to the one described in Boxho et al (2022) and used to compute space-time correlations.

The present work is performed *offline* using the same Python scripts developed for the previous tests and is based on the expressions summarized in Table 1. The near future goal is to have an *online* evaluation of these confidence intervals. The main objective is to obtain confidence interval maps as shown in Figure 8 for various averaged flow fields (e.g., velocity, pressure gradient, and Reynolds stresses).

One of the main statistics of interest is the mean streamwise velocity profile plotted along the wall-normal direction. The instantaneous data are extracted from the simulations by numerical probes located in the near-wall region at $y^+ = 100$. For each case, the estimators are evaluated over $n \in [2^7, 2^{12.5}]$. Considering a nondimensional time step $\Delta t^+ = 10^{-2}$ and a nondimensional flow time of $t_c^+ = 0.314$, the period of the averaging T/t_c extends approximately from 12 to 190. Knowing that the z-direction is homogeneous, an ensemble of 192 time series, equispaced along the z-direction, is used as new realizations of the stochastic process. The estimators are evaluated for each of these new realizations and then averaged along the spanwise direction.

As in the previous test case, there is no analytical expression for the mean, variance and ACF, so the estimators of MBB and Beyhaghi et al (2018) are considered as references. The number of autocorrelation terms m is again automatically evaluated with the algorithm 1.

Figure 7 shows the evolution of our four estimators as the number of samples n increases. The MBB method and Beyhaghi et al (2018)'s estimator are again used as references. Compared to Sections 5.1 and 5.2, Beyhaghi et al (2018)'s estimator shows a stronger sensitivity to the number of parameters used to fit the ACF. Therefore, in addition to the estimator obtained with 20 parameters, shown in magenta, the min/max envelope is also plotted. This envelope takes the minimum and maximum of the estimators obtained at each n for $[1, 2, 5, 10, 15, 20, 25, 30]$ parameters. Estimators 1, 2 and 4 evolve similarly as n increases, and all three tend to lie within the Beyhaghi et al (2018) envelope. Estimator 3 has a slightly different behavior, as we already noticed in Section 5.2. This estimator oscillates around the other three and seems to be less stable. The MBB estimator is again smaller than the other estimators, but the gap narrows as more samples are considered.

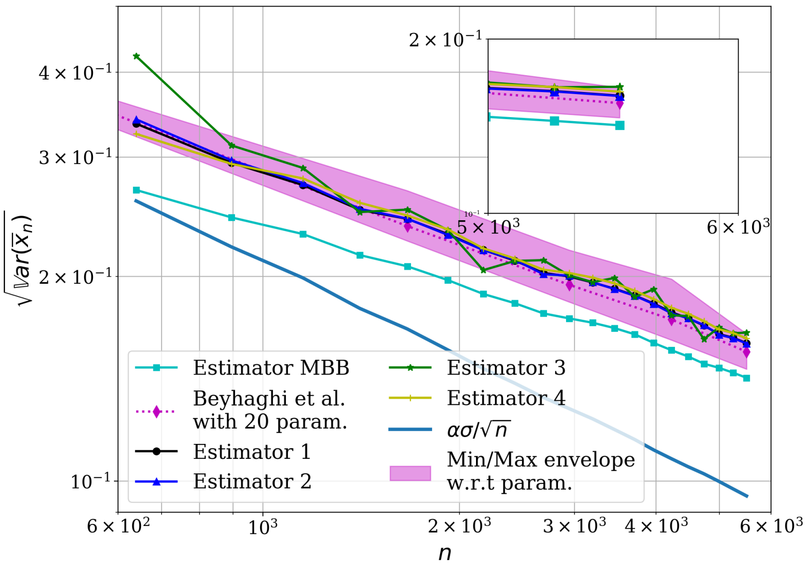

Fig. 7: Evolution of the four estimators (Table 1); estimator obtained with the MBB method (cyan line with square markers); Beyhaghi et al (2018)'s estimators obtained with 20 parameters for the fine-tuning of the ACF (magenta diamond dotted line); averaged over 192 realizations of the instantaneous wall-parallel velocity component taken in the z-direction. The basic estimator (Equation 1) is drawn in dark blue with a coefficient $\alpha = 4$ to make it visible.

5.4 LES of the two-dimensional periodic hill flow

This section deals with the estimation of the statistical error of flow data obtained by Large Eddy Simulation on the well-known two-dimensional periodic hill (Rapp et al, 2010) at the bulk Reynolds number Re_b of 10,595. The flow field is also computed using the in-house flow solver Argo-DG. The wrLES two-dimensional periodic hill data are also extracted from Argo-DG on structured probes. This test case is identical to the one employed in Boxho et al (2022) to evaluate space-time correlations to support the development of wall models. In this section, two particular locations (i.e. near the separation and after the recirculation bubble) on the periodic hill are carefully analyzed and compared with the two reference methods.

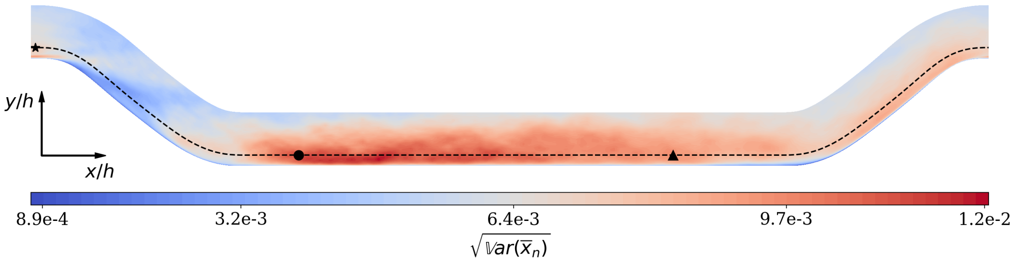

Fig. 8: Map of estimator 1 evaluated on the lower part of the two-dimensional periodic hill over 36.4 t_c and a subsampling of 1:50. The dash line represents the wall-normal height at which the instantaneous data are extracted ($\eta/h = 0.1$). The separation vicinity, recirculation and recovery regions are represented by $\star$, ●, and ▲ and symbols, respectively.

As for the channel, one of the main statistics of interest is the mean streamwise velocity profile plotted along the wall-normal direction. Two positions are targeted: points close to the flow separation ($x/h = 0.05$) and points located after the reattachment (i.e., the recovery region at $x/h = 6.0$). These locations are presented in Figure 8. The flow behavior is highly dependent of the positions because the streamwise direction is no more homogeneous. Therefore, various integral time scales are measured. For each case, the estimators are evaluated over $n \in [2^7, 2^{12.5}]$. Considering a time step $\Delta t = 5.10^{-2}\ h/u_b$ and a flow-through time of $t_c = L_x/u_b = 9\ h/u_b$, the period of averaging T/t_c extends approximately from 0.7 to 32, where u_b is the bulk velocity, h, the hill height, and L_x, the domain length in the streamwise direction. Knowing that the z-direction is homogeneous, an ensemble of 10 datasets is used for each (ξ, η)-position, equispaced along the z-direction, as new realizations of the stochastic process. As for the previous test case, no analytical expression exists for the mean, variance, and ACF. The number of auto-correlation terms m is evaluated with the algorithm described in 1.

Estimators evaluated in the separation vicinity at $x/h \simeq 0.05$

This position is located slightly downstream of the hill crest. At this location, the flow arrives at a relatively high velocity due to the strong acceleration generated by the windward slope of the hill. This location is characterized by a very thin boundary layer, as indicated by the high peak of the streamwise velocity in the near-wall region. This location is also characterized by a very high near-wall level of the turbulence component $\overline{u'u'}$ generated by intense streamwise fluctuations, which in turn are associated with rapid and random displacements of the separation over a substantial part of the hill crest. At this position and at a wall-normal height of $\eta/h \simeq 0.1$, the instantaneous wall-parallel velocity u_ξ component is extracted.

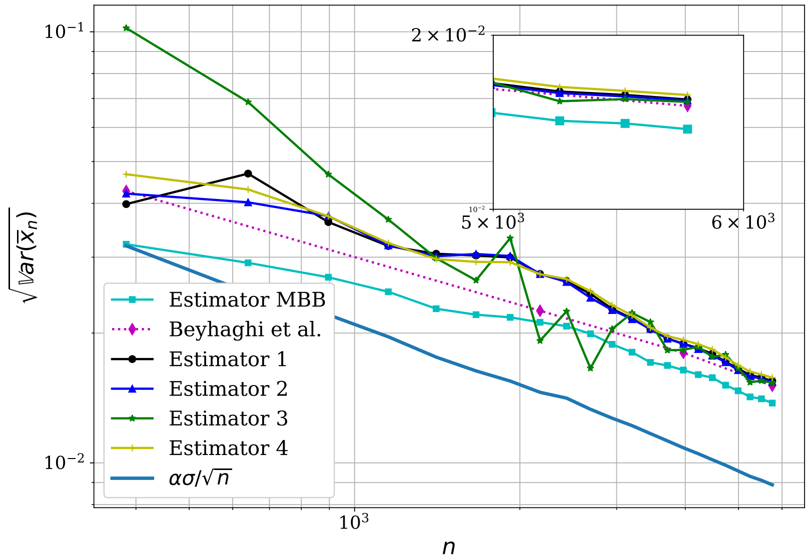

Fig. 9: Evolution of the four estimators (Table 1), the estimator obtained with the MBB method (dotted cyan line), and Beyhaghi et al (2018)'s estimator using 20 parameters for the fine-tuning of the ACF; averaged over 10 realizations of the instantaneous wall-parallel velocity component taken in the z-direction and extracted at $(\xi, \eta)/h = (0.05, 0.1)$. The basic estimator 1 is drawn in blue with a coefficient $\alpha = 4$ to make it visible.

Figure 9 shows the evolution of four estimators as n increases. Estimators 1, 2, and 4 have similar behavior and mainly follow Beyhaghi et al (2018)'s estimator. For this location, as well as for the recovery region, Beyhaghi et al (2018)'s estimator is less sensitive to the number of parameters used to fit the ACF. Therefore, no envelope is drawn as in

Section 5.3. Estimator 3 again shows an oscillatory behavior for small n values and converges to similar predictions as the other three estimators for larger n values. The MBB estimator is again below our and Beyhaghi et al (2018)'s estimators for this range of samples. Recall that MBB takes more samples to converge properly (see Section 5.1).

Estimators evaluated after the reattachment at $x/h \simeq 6.0$

Estimators are computed in the post-reattachment region (see Figure 10), halfway between the reattachment location and the foot of the next hill. This zone is also called the recovery region because the flow is characterized by a developing boundary layer from the reattachment point. Above this boundary layer, the flow consists of a wake emanating from the separated shear layer. The recovery region is, hence, composed of various scales and histories that interact with each other. The four estimators are in agreement with Beyhaghi et al (2018)'s estimator, converging to the same confidence interval as n increases. Note that for this particular location, estimator 3 is more stable and does not exhibit any oscillations. Again, the MBB estimator is lower and the gap with our estimators decreases as n increases. A deviation of 0.6% and 16% is measured for the Beyhaghi et al (2018) and MBB estimators respectively at the largest n.

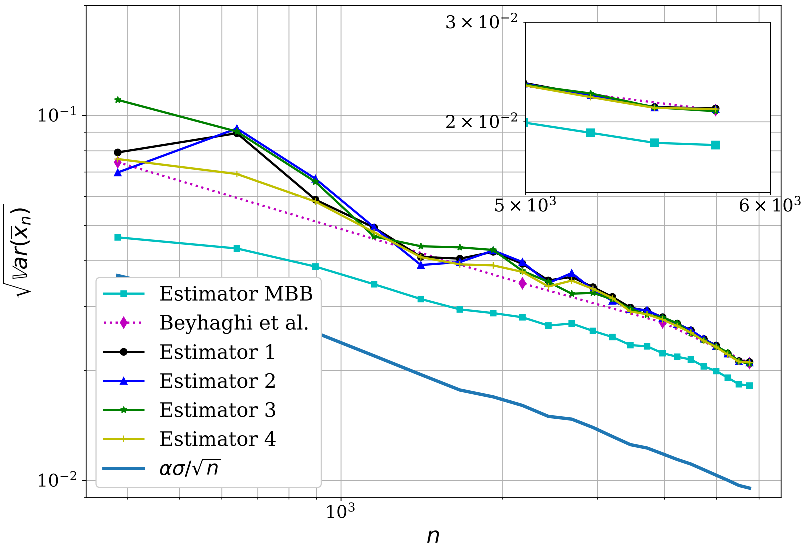

Fig. 10: Same caption as Figure 9 but extracted at $(\xi, \eta)/h = (6.0, 0.1)$.

5.5 Undersampling strategies applied to the two-dimensional periodic hill

Each estimator defined in Table 1 contains a truncated sum of the m first terms of the ACF. The parameter m is an image of the integral length scale $\mathcal{T}$ and is computed with the algorithm 1. However, for some physical processes, this characteristic time scale can be large, which means that a large number of terms must be stored in memory. The example of the two-dimensional periodic hill, fully described in Subsection 5.4, is considered to illustrate the issue. Depending on the location along the lower wall of the two-dimensional periodic hill, $\mathcal{T}$ can vary significantly. In the recirculation zone, large structures are convected at low speed generating a high integral time scale. Table 2 summarizes the quantities derived from the integral time scale, including the size m of the truncated sum, for different locations along the periodic hill. It is important to note that the parameter m is case-dependent (e.g., periodic hill, channel flow, blade, etc.), location-dependent, varies with the statistical quantity of interest (e.g., velocity field, Mach number, density, etc.), and depends on the temporal discretization (e.g., implicit, explicit).

Table 2: Evaluation of the integral time scale at different locations along the lower wall of the two-dimensional periodic hill measured for the wall-parallel velocity component u_ξ at $\eta/h = 0.1$

	ξ/h	$l^\star/t_c$	$\gamma^\star$	$\mathcal{T}/t_c$	m
Separation	0.05	0.118	0.673	0.156	≈ 544
Recirculation bubble	2.50	0.234	0.692	0.299	≈ 1027
Reattachment	6.00	0.212	0.927	0.220	≈ 625

The primary objective is to generate statistics of interest and, at the same time, a confidence interval size map (as shown in Figure 8) through co-processing in a flow solver. In the case of estimator 2, as defined in Table 1, it is necessary to store m coefficients of the ACF estimator, m terms of the time series used to accumulate the ACF estimator, the sample mean and the sample variance. Consequently, a total of $2(m + 1)$ doubles should be stored at each interpolation point. To illustrate this, consider the example of our flow solver, Argo-DG. A general guideline is to have about $1,000$ elements per partition. Assuming a third polynomial order ($p = 3$), the number of degrees of freedom per partition is $64,000$. The estimator is evaluated not only on the streamwise velocity, but also on the wall normal and spanwise components of the velocity, and on the Reynolds stress tensor, which is of great interest to validate the simulation. In total, nine statistics have to be kept. Assuming that 1,000 correlation coefficients have to be kept in memory to compute the estimator at each interpolation point, and that the velocity is stored as a double, the memory requirement to store the estimator for these nine statistics is 8.6 GiB. The direct approach is completely forbidden. However, undersampling can be used to reduce the memory requirement. The idea is to subsample the time series without losing the precision of the estimator. This approach also affects the integral time scale approximation described in equations 17, and 18. Furthermore, as a co-processing technique, this scale is updated with a dynamic procedure until a converged value is obtained.

This undersampling technique of the time series is tested for the recirculation bubble (i.e., $x/h = 2.50$). The number of coefficients selected to fit the l and γ parameters of the GEF is again automatically set using the procedure described in the algorithm 1. If the undersampling ratio is not larger than the integral time scale then the ACF is barely affected by the undersampling procedure. Therefore, the optimization procedure for finding l^* and γ^* leads to similar results because the same part of the ACF is encapsulated in the K coefficients. Regarding m, the optimization procedure leads to 742, 195, 114, and 57 for the sampling ratios of 1:1, 1:3, 1:5, and 1:10, respectively, highlighting the drastic reduction in the number of terms required to evaluate the estimators.

Figure 11 shows the evolution of the four estimators for different subsampling ratios against the number of samples n. The n values are scaled for better comparison with the reference (i.e. 1:1). Except for the smaller n, the estimators are not much affected by the subsampling. Moreover, for the largest n, the relative error of the $1 : 10$ subsampling compared to the prediction of the confidence interval on the original time series is less than 1%. Assuming a subsampling of $1 : 10$, the memory requirement for the nine statistics of interest is reduced from 11 GiB (i.e. for a set of $1,000$ correlation coefficients) to 225 MB (i.e. for a set of 50 correlation coefficients), which represents an acceptable memory overhead in a large-scale CFD simulation.

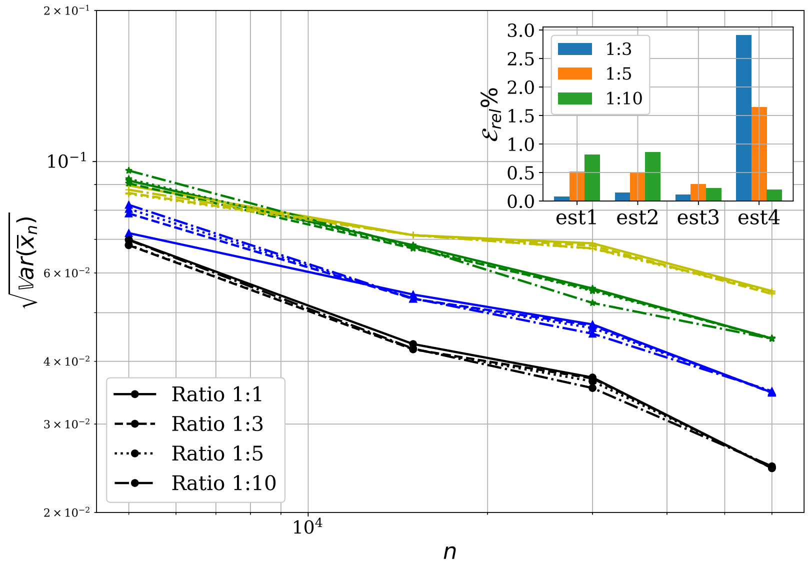

Fig. 11: Prediction of the confidence interval size of the three estimators based on subsampled time series. Black (blue, green, and yellow) lines represent estimator 1 (2, 3, and 4, respectively). Plain lines '—' are estimators computed using the original time series, dashed '− −' (dotted ':', and dashdotdotted '−.') lines are estimators computed using a subsampled time series with a ratio of 1:3 (1:5, and 1:10). The estimators are shifted from each other by 10^{-2} for better visibility. Note that the black curve is at zero offset.

5.6 Recommendations

Based on the results obtained on the different test cases, we advise the reader, who would like to implement our estimators in his favourite flow solver, to choose the second estimator $\hat{\mathcal{V}}_n^{(2)}$, as it gives similar results as estimator 1, but has a simpler expression of the accumulation. Although the determination of the parameter m is based on the standard estimator of the ACF, the expression of the quadratic variation estimator can be adjusted to mimic the standard estimator of the ACF. Using the expression developed in the expectation of the quadratic variation estimation in Section 4.3, the following adjustment expression is obtained,

$$\hat{\gamma}_k^n \approx \hat{\gamma}_{k,adjust}^n = s_n^2 - \frac{n}{2(n-k)}\hat{\delta}_k^n \,.$$

This last expression is an approximation of the standard ACF estimator using the quadratic variation estimator. Figure 12 shows the standard ACF estimator and the adjusted expression of the quadratic variation estimator for one realization of the AR process presented in Section 5.1 where $n \approx$ 16,000. The two functions overlap well each other up to $k \approx$ 3,000, which is also confirmed by the inserted plot showing the RMSE. Beyond this value of k, the error becomes larger. However, for the AR process, we have seen that only the first 60-80 terms are employed to evaluate the estimator. For these coefficients, the agreement between the two ACF estimators is almost perfect. With this simple adjustment of the quadratic variation estimator, no accumulation of the standard ACF estimator is needed to fit the parameters l and γ of the GEF and thus to determine m.

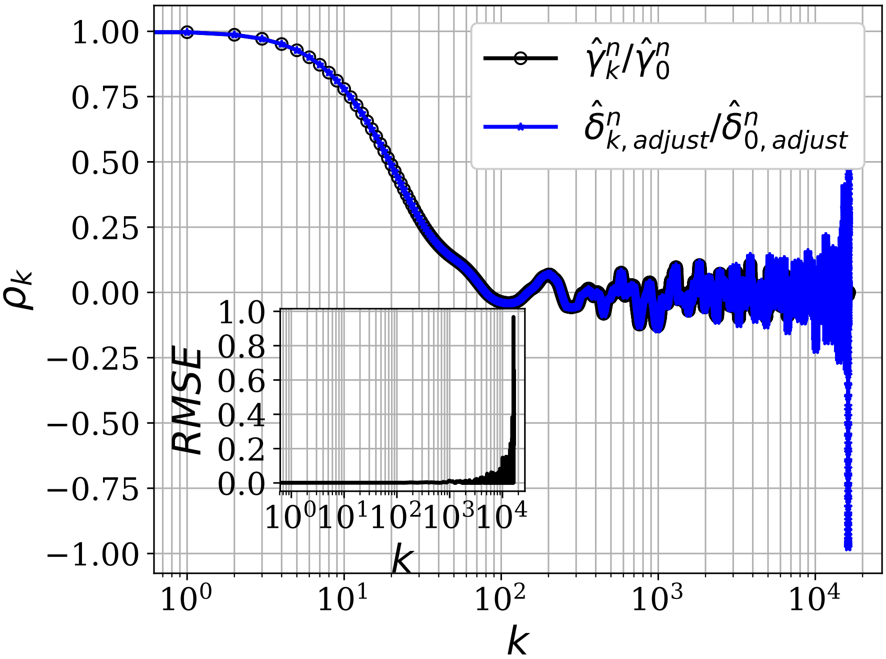

Fig. 12: The standard ACF estimator (black line) and the adjusted version of the quadratic variation estimator (blue line); the Root-Mean-Square-Error (RMSE) is plotted in the lower left corner.

6 Conclusion

Four estimators have been presented in this work for the prediction of the variance of the sample mean, i.e., to quantify the uncertainty linked to the approximation of infinite-time-average statistics of statistically stationary ergodic processes. The main goal is to create asymptotically unbiased estimators that can be computed with a **cumulative approach** and a **small set of correlation coefficients** to reduce the memory storage in scale-resolving flow solvers.

Our method is based on the mathematical development of the variance of the sample mean $\mathbb{V}\text{ar}\left(\overline{x}_n\right)$, where the sum over the correlation coefficients has been truncated up to m. This parameter m is an image of the integral time scale $\mathcal{T}$ and is determined by fitting a Gamma-Exponential function using a database composed of a certain number of ACF coefficients. Under this mathematical framework, our estimators are asymptotically unbiased by construction. As a result, the estimator is easy to implement in a scale-resolving flow solver as a co-processing tool. The long-term goal is to provide confidence interval maps of all computed statistics for a fair comparison with experimental data and other high-fidelity simulations.

The four estimators have been tested on four test cases: an auto-regressive process, the time evolution of the kinetic energy in the Kuramoto-Sivashinsky equation, the turbulent channel flow at a friction Reynolds number of 950, and the two-dimensional periodic hill at a bulk Reynolds number of 10,595. They have been compared with the Moving Block Bootstrap method and the estimator developed in the paper of Beyhaghi et al (2018), which both require all samples to evaluate the variance of the sample mean, which is intractable in high-fidelity simulations. For the four test cases, three of our estimators show similar behavior as the estimator developed by Beyhaghi et al (2018). Estimators 1, 2 and 4 have shown equivalent results for the three test cases, while estimator 3 oscillates more in the more realistic test cases (i.e., the turbulent channel and the periodic hill), and it is quite sensitive to the value of m.

To address the memory storage issue, a subsampling approach has been successfully applied to our four estimators. The subsampling of the original time series has a negligible impact on the precision of the estimators if the subsampling ratio is not greater than the integral time scale. This subsampling approach allows us to drastically reduce the memory storage and hence increase the feasibility of our approach for massively parallel high-fidelity simulations.

As discussed in Section 5.6, estimator 2 is recommended to the readers for implementation in their favorite flow solver.

The next steps of this work will be devoted to the implementation of estimator 2 (i.e., $\hat{\mathcal{V}}_n^{(2)}$) in a higher-order flow solver according to the cumulative formula given in Table 1. The notion of the initial transient also needs to be treated with a feasible criterion since the one proposed by Beyhaghi et al (2018) requires the complete original signal. The current approach needs to be validated for higher statistical moments, such as the Reynolds stress $\overline{u_i'u_j'}$.

Acknowledgments. The funding of M. Boxho by Safran Tech is gratefully acknowledged. The present research benefited from computational resources made available on the Tier-1 supercomputer Zenobe of the Fédération Wallonie-Bruxelles, infrastructure funded by the Walloon Region under the grant agreement n°1117545. The authors thank Maarten Arnst (University of Liege) for his constructive comments on the original manuscript.

Declarations

Some journals require declarations to be submitted in a standardised format. Please check the Instructions for Authors of the journal to which you are submitting to see if you need to complete this section. If yes, your manuscript must contain the following sections under the heading 'Declarations':

- Funding: Safran Tech and computational resources made available on the Tier-1 supercomputer of the Fédération Wallonie-Bruxelles, infrastructure funded by the Walloon Region under the grant agreement n°1117545.
- Conflict of interest/Competing interests: Not applicable
- Ethics approval: Not applicable
- Consent to participate: Not applicable
- Consent for publication: Not applicable
- Availability of data and materials: No but can be made available if needed
- Code availability: No but can be made available if needed
- Authors' contributions:
  - Conceptualization: [Koen Hillewaert];
  - Data curation: [Margaux Boxho];
  - Formal analysis: [Margaux Boxho, Koen Hillewaert, Michel Rasquin];
  - Funding acquisition: [Koen Hillewaert];
  - Investigation: [Margaux Boxho];
  - Methodology: [Margaux Boxho, Koen Hillewaert];
  - Project administration: [Koen Hillewaert, Thomas Toulorge, Grégory Dergham];
  - Resources: [Cenaero, Safran Tech];
  - Software: [Margaux Boxho];
  - Supervision: [Koen Hillewaert, Thomas Tourlorge, Michel Rasquin, Grégory Dergham];
  - Validation: [Margaux Boxho, Koen Hillewaert];
  - Visualization: [Margaux Boxho];
  - Writing - original draft preparation: [Margaux Boxho, Koen Hillewaert];
  - Writing - review and editing: [Margaux Boxho, Koen Hillewaert, Thomas Tourlorge, Michel Rasquin, Grégory Dergham];

References

Alimohammadi S, He D (2016) Multi-stage algorithm for uncertainty analysis of solar power forecasting. In: 2016 IEEE Power and Energy Society General Meeting (PESGM), pp 1–5, https://doi.org/10.1109/PESGM.2016.7741199

Andersen TG, Bollerslev T, Diebold FX, et al (2001) The distribution of realized exchange rate volatility. Journal of the American Statistical Association 96(453):42–55. https://doi.org/10.1198/016214501750332965

Bernardes M, Dias N (2010) The alignment of the mean wind and stress vectors in the unstable surface layer. Boundary-layer meteorology 134:41—-59. https://doi.org/10.1007/s10546-009-9429-8

Beyhaghi P, Alimohammadi S, Bewley T (2018) Uncertainty quantification of the time averaging of a statistics computed from numerical simulation of turbulent flow. arXiv:180201056 [physics, stat]

Boufidi E (2021) The characterization of turbulence in high speed compresisble and complex industrial flows. PhD thesis, Université catholique de Louvain

Boufidi E, Lavagnoli S, Fontaneto F (2020) A probabilistic uncertainty estimation method for turbulence parameters measured by hot-wire anemometry in short-duration wind tunnels. Journal of Engineering for Gas Turbines and Power 142:031007. https://doi.org/10.1115/1.4044780

Box G, Jenkins G, Reinsel G, et al (2015) Time Series Analysis: Forecasting and Control. 5th edition, Wiley, Prentice–Hall

Box G, Jenkins G, Reinsel G, et al (2016) Time Series Analysis: Forecasting and Control - Fifth Edition. Wiley, Hoboken, New Jersey

Boxho M, Rasquin M, Toulorge T, et al (2022) Analysis of Space-Time Correlations to Support the Development of Wall-Modeled LES. Flow, Turbulence and Combustion 109(4):1081–1109. https://doi.org/10.1007/s10494-022-00365-3, URL https://doi.org/10.1007/s10494-022-00365-3

Broersen PMT (2002) Automatic spectral analysis with time series models. IEEE Transactions on Instrumentation and Measurement 51(2):211—-216. https://doi.org/10.1109/19.997814

Broersen PMT (2006) Automatic autocorrelation and spectral analysis. Springer, https://doi.org/10.1007/1-84628-329-9

Brunton SL, Noack BR, Koumoutsakos P (2020) Machine learning for fluid mechanics. Annual Review of Fluid Mechanics 52(1):477–508. https://doi.org/10.1146/annurev-fluid-010719-060214, URL https://www.annualreviews.org/doi/10.1146/annurev-fluid-010719-060214

Conway R (1963) Some tactical problems in digital simulation. Manag Sci 10(1):47–61. https://doi.org/10.1287/mnsc.10.1.47

Conway R, Johnson B, Maxwell W (1959) Some problems of digital systems simulation. Manag Sci 6(1):92–110. https://doi.org/10.1287/mnsc.6.1.92

Donzis D, Yeung P, Sreenivasan KR (2008) Dissipation and enstrophy in isotropic turbulence: Resolution effects and scaling in direct numerical simulations. Physics of Fluids 20:045108. https://doi.org/10.1063/1.2907227

Duraisamy K, Iaccarino G, Xiao H (2019) Turbulence modeling in the age of data. Annual Review of Fluid Mechanics 51(1):357–377. https://doi.org/10.1146/annurev-fluid-010518-040547, URL http://arxiv.org/abs/1804.00183, 1804.00183

Dürre A, Fried R, Liboschik T (2015) Robust estimation of (partial) autocorrelation. Wiley Interdisciplinary Reviews: Computational Statistics 7(3):205–222. https://doi.org/10.1002/wics.1351, URL https://onlinelibrary.wiley.com/doi/10.1002/wics.1351

Efron B (1979) Bootstrap methods: Another look at the jackknife. Ann Statist 7(1):1–26. https://doi.org/10.1214/aos/1176344552

Favier D (2010) The role of wind tunnel experiments in CFD validation. In: Encyclopedia of Aerospace Engineering. John Wiley and Sons, Ltd., https://doi.org/10.1002/9780470686652.eae034

Fonseca E, Rangel V, Brener B (2022) Pre-processing dns data to improve statistical convergence and accuracy of mean velocity fields in invariant data-driven turbulence models. Theor Comput Fluid Dyn 36:435—463. https://doi.org/10.1007/

s00162-022-00603-4

Hayashi T, Yoshida N (2005) On covariance estimation of non-synchronously observed diffusion processes. Bernoulli 11(2):359–379. https://doi.org/10.3150/bj/1116340299

Hillewaert K (2013) Development of the Discontinuous Galerkin method for high-resolution, large scale CFD and acoustics in industrial geometries. PhD thesis, Université catholique de Louvain

Hosking JR (1981) Fractional differencing. Biometrika 68(1):165–176. https://doi.org/10.2307/2335817

Hoyas S, Jiménez J (2008) Reynolds number effects on the reynolds-stress budgets in turbulent channels. Physics of Fluids 20(10):101511. https://doi.org/10.1063/1.3005862

Kunsch HR (1989) The jackknife and the bootstrap for general stationary observations. Ann Statist 17(3):1217–1241. URL https://www.jstor.org/stable/2241719

Kuramoto Y (1978) Diffusion-induced chaos in reaction systems. Progress of Theoretical Physics Supplement 64:346–367. https://doi.org/10.1143/PTPS.64.346, URL https://doi.org/10.1143/PTPS.64.346

Law A, Kelton W (2000) Simulation Modeling and Analysis. 3rd Edition, McGraw–Hill, Boston

Liao L, Park C, Hannig J, et al (2016) Autocovariance function estimation via penalized regression. Journal of Computational and Graphical Statistics 25(4):1041–1056. https://doi.org/http://www.jstor.org/stable/44861908, publisher: [American Statistical Association, Taylor & Francis, Ltd., Institute of Mathematical Statistics, Interface Foundation of America]

Lumley J, Panofsky H (1964) The structure of atmospheric turbulence. Interscience

Ma Y, Genton MG (2000) Highly robust estimation of the autocovariance function. Journal of Time Series Analysis 21(6):663–684. https://doi.org/10.1111/1467-9892.00203, URL http://doi.wiley.com/10.1111/1467-9892.00203

Meketon M, Schmeiser B (1984) Overlapping batch means: something for nothing? in: WSC '84 Proceedings of the 16th Conference on Winter Simulation 27:226–230

Oliver TA, Malaya N, Ulerich R, et al (2014) Estimating uncertainties in statistics computed from direct numerical simulation. Physics of Fluids 26(3):035101. https://doi.org/10.1063/1.4866813

Politis DN, White H (2004) Automatic block-length selection for the dependent bootstrap. Econometric Reviews 23(1):53–70. https://doi.org/10.1081/ETC-120028836

Pope SB (2000) Turbulent Flows. Cambridge University Press, Cambridge

Rapp C, Breuer M, Manhart M, et al (2010) Underlying Flow Regime 3-30: 2D Periodic Hill Flow. ERCOFTAC KB Wiki, URL https://kbwiki.ercoftac.org/w/index.php/Abstr:2D_Periodic_Hill_Flow

Rezaeiravesh S, Gscheidle C, Peplinski A, et al (2023) In-situ estimation of time-averaging uncertainties in turbulent flow simulations. arXiv https://doi.org/10.48550/arXiv.2310.08676

Russo S, Luchini P (2017) A fast algorithm for the estimation of statistical error in dns (or experimental) time averages. Journal of Computational Physics 347:328–340

Schmeiser B (1982) Batch size effects in the analysis of simulation output. Oper Res 30(3):556–568. https://doi.org/10.1287/opre.30.3.556

Trenberth KE (1984) Some effects of finite sample size and persistence on meteorological statistics. part i: Autocorrelations. Monthly Weather Review 112(12):2359–2368. https://doi.org/10.1175/1520-0493(1984)112⟨2359:SEOFSS⟩2.0.CO;2

Vogelsang TJ, Yang J (2016) Exactly/nearly unbiased estimation of autocovariances of a univariate time series with unknown mean. Journal of Time Series Analysis 37(6):723–740. https://doi.org/10.1111/jtsa.12184, URL https://onlinelibrary.wiley.com/doi/abs/10.1111/jtsa.12184

Wei WWS (2006) Time Series Analysis: Univariate and Multivariate Methods. 2nd Edition, Pearson Addison Wesley

Welford BP (1962) Note on a method for calculating corrected sums of squares and products. Technometrics 4(3):419–420

Carton de Wiart C, Hillewaert K, Bricteux L, et al (2014) Implicit les of free and wall bounded turbulent flows based on the discontinuous galerkin/symmetric interior penalty method. International Journal of Numerical Methods in Fluids 78(6):335—-354. https://doi.org/10.1002/fld.4021