

Markov chain Monte Carlo techniques applied to parton distribution functions determination: Proof of concept

Y emalin Gabin Gbedo and Mariane Mangin-Brinet*

*Laboratoire de Physique Subatomique et de Cosmologie-Universit  Grenoble-Alpes,
CNRS/IN2P3, 53, avenue des Martyrs, 38026 Grenoble, France*

(Received 26 January 2017; published 21 July 2017)

We present a new procedure to determine parton distribution functions (PDFs), based on Markov chain Monte Carlo (MCMC) methods. The aim of this paper is to show that we can replace the standard χ^2 minimization by procedures grounded on statistical methods, and on Bayesian inference in particular, thus offering additional insight into the rich field of PDFs determination. After a basic introduction to these techniques, we introduce the algorithm we have chosen to implement—namely Hybrid (or Hamiltonian) Monte Carlo. This algorithm, initially developed for Lattice QCD, turns out to be very interesting when applied to PDFs determination by global analyses; we show that it allows us to circumvent the difficulties due to the high dimensionality of the problem, in particular concerning the acceptance. A first feasibility study is performed and presented, which indicates that Markov chain Monte Carlo can successfully be applied to the extraction of PDFs and of their uncertainties.

DOI: [10.1103/PhysRevD.96.014015](https://doi.org/10.1103/PhysRevD.96.014015)

I. INTRODUCTION AND MOTIVATION

Quantum chromodynamics (QCD) is the theory of strong interaction, whose ambition is to explain nuclei cohesion as well as neutron and proton structure, i.e. most of the visible matter in the Universe. Its application domain is even wider, since QCD controls the structure and interactions of all hadrons: proton, neutron, hyperons, pions, kaons, etc. It is one of the most elegant theories of science (with general relativity); it has only very few parameters and allows us to give a physical interpretation to a very broad range of phenomena using a well-defined and very compact formalism.

Among the fundamental ingredients of QCD, parton distribution functions (PDFs) are key elements and play an essential role to connect the QCD dynamics of quarks and gluons to the measured hard scattering cross sections of colliding hadron(s). They carry an invaluable source of information on the hadrons' partonic structure, and enormous theoretical and experimental effort has been devoted for years to the extraction of these distribution functions.

PDFs are all the more important nowadays; with the start of data taking at the LHC, they are essential for the computation of a large class of observables. Built for the discovery of the Higgs boson and the study of physics beyond the standard model, the LHC is indeed essentially a QCD factory, producing events in an unexplored energy range. The LHC potential of discovery crucially depends on the quality of predictions for QCD signals and backgrounds and, thus, on the PDFs quality.

PDFs are intrinsically nonperturbative objects and, thus, cannot be determined using only perturbative QCD tools. One of the most efficient methods to perform non-perturbative QCD calculations is Lattice QCD. However, although nucleon structure has been the subject of intense activity in Lattice QCD for years, and even if promising techniques to compute PDFs directly on the lattice have recently been proposed [1], *ab initio* calculations of PDFs are very challenging and still not a competitive alternative to global analyses. These latter, thus, remain the chosen method to obtain PDFs, which are parametrized by functional forms whose parameters are constrained by fits to the data.

Parton distribution function determination thus consists in extensive exploitation of data sets collected at colliders to constrain the parameters of the PDF functional forms given at a fixed scale in energy. Such analyses are usually based on a least square fit method, i.e. on the minimization of a χ^2 , which compares the input data and theory predictions. PDFs determined this way did not, for a long time, include any estimate of uncertainties, other than the mere comparison of results provided by different global analyses collaborations. However, with the advent, at the dawn of the 21st century, of the new generation of colliders and the active search for new physics, the need to assess the uncertainties of the PDFs became clear [2–5]. Many studies have since been devoted to the estimates of uncertainties on physical predictions due to the uncertainties of PDFs (see for instance [6–8] and references therein) and considerable progress has been made. Nevertheless, this task is far from being trivial, and many issues remain open [6].

Our current understanding of PDF uncertainties is mainly based—with the notable exception of neural

*Corresponding author.
mariane@lpsc.in2p3.fr

network techniques—on the Hessian or the Lagrange multiplier method [6,7]. The determination of the uncertainties then relies on an assumption on the permissible range of “acceptable” $\Delta\chi^2$ for the fit and the choice of a tolerance parameter. In order to improve on this point and get a deeper insight, we propose to use Markov chain Monte Carlo (MCMC) techniques to define the uncertainties in a way based as much as possible on robust statistical methods. Markov chain Monte Carlo algorithms have been an extremely popular tool in statistics. While these techniques are already widely used in many areas of physics (see for instance [9–11]) they have not yet been employed as a stand-alone method to compute PDF parameters and their errors—i.e. without resorting to a χ^2 minimization procedure.¹

The Markov chain Monte Carlo method allows us to estimate *a posteriori* probability densities for multidimensional models and provides reliable estimates of errors. MCMC consists in sequentially simulating a single Markov chain whose limit distribution is the chosen one.

The main challenge of the present paper is to demonstrate that Markov chain Monte Carlo techniques can be applied to PDF extraction. The higher the dimension of the chain (i.e. in our case, the more PDF free parameters to be determined), the more computing time is needed to generate the chain. The large number of parameters to be computed in a full PDF determination has led us to make use of a Monte Carlo algorithm based on molecular dynamics, initially developed for lattice field theory. We apply this algorithm to a realistic (though not full) extraction of PDFs, based on ten-parameter functional forms and four data sets, to demonstrate that Markov chain Monte Carlo can successfully be applied to PDF computation.

This paper is organized as follows: in Sec. II, we formulate the PDF determination problem in terms of Bayesian inference. In the next section, the basic principles of Markov chain Monte Carlo methods are recalled and illustrated utilizing the widely used Metropolis algorithm. The fourth section briefly presents the hybrid Monte Carlo algorithm and shows how it allows us to deal with the large number of PDF parameters to be determined. Section V details the MCMC analysis procedure and Sec. VI displays first results with a realistic run using ten parameters and four sets of data. Conclusions and outlook are discussed in the last section.

II. FORMULATION OF THE PROBLEM IN TERMS OF BAYESIAN INFERENCE

Parton distribution functions are usually (with the exception of neural network procedure [12]) parametrized at a given energy scale by functional forms, which are then evolved at any other scale thanks the DGLAP equations

¹Uncertainties estimations using pseudodata replicas are also based on Monte Carlo methods [8], but still rely on fits.

(this also excepts dipole models [13] and transverse-momentum dependent [14] and unintegrated PDFs [15], for instance, that we will not consider here). These PDFs are convoluted with partonic cross sections to obtain hadronic cross sections for various processes and a χ^2 function, constructed from these theoretical cross sections and corresponding experimental data, is then minimized to constrain the PDF parameters. Rather than using a minimization procedure and a Hessian method to estimate PDF uncertainties, we propose a Bayesian parameter inference approach. These techniques have already been successfully applied in many areas [9] and we only sketch the main principles in what follows. The interested reader can referred for instance to [16] for a more extensive review of the subject.

For compactness, we note \hat{q} the vector of PDF parameters to be determined: $\hat{q} = (q^{(1)}, q^{(2)}, \dots, q^{(m)})^T$ where m is typically, in the case of a full analysis, of the order of 25–30, and D the data. From a Bayesian perspective, both model parameters and observables are considered random quantities, and Bayesian inference aims formally to determine a joint probability distribution $P(D, \hat{q})$ over all random quantities. This joint distribution can be written as $P(D, \hat{q}) = P(D|\hat{q})P(\hat{q})$, where $P(\hat{q})$ is a *prior* distribution—quantifying the degree of belief one has *a priori* before observing the data—and $P(D|\hat{q})$ is the likelihood of the data: $\mathcal{L}(\hat{q}) \stackrel{\text{def}}{=} P(D|\hat{q})$. Bayes theorem is used to express the distribution of \hat{q} conditional on D , $P(\hat{q}|D)$, in terms of the likelihood $P(D|\hat{q})$:

$$P(\hat{q}|D) = \frac{P(D|\hat{q})P(\hat{q})}{\int d\hat{q}P(D|\hat{q})P(\hat{q})} \quad (1)$$

The denominator in (1) does not depend on the parameters and can be considered only as a normalization. This so-called “*posterior*” probability density $P(\hat{q}|D)$ quantifies the probability to have the model parameters \hat{q} given the observed data D and is the object we deal with in all Bayesian inference. To determine this conditional probability, we thus need to set a prior distribution for the parameters, and to compute the likelihood of the data. The probability density $P(\hat{q}|D)$ is then sampled using a Monte Carlo algorithm.

Assuming that the fluctuations of the n experimental data points under consideration around their corresponding theoretical values are uncorrelated and distributed according to a Gaussian law (assumption whose validity can be assessed *a posteriori*—see Sec. VI), the least square method and the maximum likelihood are equivalent and the logarithmic likelihood function can be written as

$$\log \mathcal{L}(\hat{q}) = -\frac{1}{2} \sum_{i=1}^n \frac{(D_i - T_i)^2}{\sigma_i^2} = -\frac{1}{2} \chi^2 \quad (2)$$

where D_i and T_i denote respectively the i th experimental point and the corresponding theoretical calculation, and

σ_i^2 is the uncertainty associated with the measured data i . The inclusive cross section T_i in hadron collision can be written as a convolution of PDFs with a partonic cross section, computed at a given order in perturbation theory. The likelihood function (2) thus contains the PDFs. Correlated experimental uncertainties can also be taken into account by introducing for instance a covariant matrix and properly modifying the χ^2 [17].

For this work, we have used the χ^2 function given by the default settings provided by the HeraFitter package [17] with 10 parameters (default HeraFitter steering file and minuit.in.txt.10pHERAPDF input file) and four data sets with an initial PDF parametrization set at $Q_0^2 = 1.9 \text{ GeV}^2$ and a lower cut on the data at $Q_{\min}^2 = 10 \text{ GeV}^2$. These settings provide already a computation of PDFs realistic enough for this feasibility study.

We thus apply Bayesian inference to the likelihood function defined in (2), that is, we compute the probability density function of the model parameters, based on selected experimental data. To this purpose, we use Monte Carlo Markov chain procedure, whose principles are briefly sketched in the next section. One of the crucial interest of these methods is the fact that the mean value and uncertainty in these parameters are by-products of the probability density functions computed.

III. PRINCIPLE OF MARKOV CHAIN MONTE CARLO

A. Basics of the method

The Markov chain Monte Carlo method allows us to estimate *a posteriori* probability densities for multi-dimensional models—which, as explained briefly in the previous section, is exactly what we want—and provides reliable estimates of errors. MCMC algorithms enable us to draw samples from a probability distribution known up to a multiplicative constant. They consist in sequentially simulating a single Markov chain whose limiting distribution is the chosen one (in our case, the maximum likelihood times a prior density). More precisely, a Markov chain is a stochastic process characterized by the fact that the conditional distribution of the random variable at iteration t , denoted \hat{q}_t , given the ensemble of random variables at all previous steps $\hat{q}_0, \dots, \hat{q}_{t-1}$, depends only on \hat{q}_{t-1} , and not on the previous history. Such a chain can be used to sample a probability density. To converge to a given stationary distribution, the chain needs to satisfy important properties: it has to be irreducible, aperiodic and positive recurrent. We will not expand further on Markov chain theory and we refer the reader interested by formal details to [9] and references therein.

Two ingredients are necessary to define a Markov chain: (i) the initial values (that is the marginal distribution) of parameters and (ii) the transition kernel between two sets of parameters: $T(\hat{q} \rightarrow \hat{q}')$, for going from a set \hat{q} to another

set \hat{q}' . There are several issues arising when implementing MCMC: the influence of the starting point of the chain (leading to the “burn-in” time), the choice of the transition kernel, the rate of convergence, the acceptance of the algorithm,... These questions will be illustrated in detail in the following sections.

B. Metropolis algorithm

The so-called “Metropolis-Hastings algorithm,” proposed in 1953 by Metropolis *et al.* [18] and generalized by Hastings in 1970 [19], is one of the simplest Monte Carlo algorithms. It is the standard computational workhorse of MCMC methods both for its simplicity and its versatility, and is in principle applicable to any system. It is extremely straightforward to implement and to sample a target density $P(\hat{q}|D)$ (see Sec. II), it proceeds as follows: at each Monte Carlo time $t - 1$, the next state \hat{q}_t is chosen by sampling a candidate point \hat{q}' from a proposal distribution $\pi(\cdot|\hat{q}_{t-1})$. The candidate point is then accepted with the probability

$$\alpha(\hat{q}_{t-1}, \hat{q}') = \min\left(1, \frac{P(\hat{q}'|D)\pi(\hat{q}_{t-1}|\hat{q}')}{P(\hat{q}_{t-1}|D)\pi(\hat{q}'|\hat{q}_{t-1})}\right),$$

and the Metropolis-Hastings transition kernel is thus

$$T(\hat{q}_{t-1} \rightarrow \hat{q}') = \pi(\hat{q}'|\hat{q}_{t-1})\alpha(\hat{q}_{t-1}, \hat{q}').$$

If the new set of parameters \hat{q}' is accepted, the next state of the chain becomes $\hat{q}_t = \hat{q}'$. If it is rejected, the chain does not move and the point at t is identical to the point at $t - 1$: $\hat{q}_t = \hat{q}_{t-1}$.

A special case of the Metropolis-Hastings algorithm is the random walk Metropolis, for which the proposal distribution is chosen to be such that $\pi(\hat{q}'|\hat{q}_{t-1}) = \pi(|\hat{q}_{t-1} - \hat{q}'|)$. The acceptance probability then reduces to $\alpha(\hat{q}_{t-1}, \hat{q}') = \min(1, \frac{P(\hat{q}'|D)}{P(\hat{q}_{t-1}|D)})$. Frequently, the proposal for the random walk jump has a form which depends on a scale parameter, giving the typical “size” of the leap from one site to the other. For instance the proposal distribution for \hat{q}' could be a normal distribution centered in \hat{q}_{t-1} with a standard deviation σ . A meticulous attention has to be taken when choosing this scale parameter. If it is too large, a very high percentage of the iterations will be rejected, leading to an inefficient algorithm. If it is too small, the random walk will explore the parameter space very slowly, leading again to inefficiency. This problem is all the more difficult to handle that the number of parameters (i.e. the dimension of the vector \hat{q}) to be sampled increases.

Ideally, to optimize the efficiency of the MCMC, the proposal distribution should be as close as possible to the target distribution. In practice, the performance of the algorithm is obviously highly dependent on the choice of proposal distribution $\pi(\cdot|\hat{q}_{t-1})$ and several options are

usually considered in the literature to explore the parameter space: one-dimensional Gaussian distributions, multivariate Gaussian distributions, a distribution obtained by binary space partitioning [20] However, even if adjustments of the proposal distributions improve the Metropolis efficiency, they are not effective enough to efficiently deal with several dozen parameters within a reasonable CPU and user time. To circumvent these problems—since in the case of PDF extraction, the number of free parameters to determine (that is the number of parameters in the PDF functional form) is of the order of $\sim 25\text{--}30$ —we have implemented a much more efficient algorithm, based on molecular dynamics, which has initially been developed for Lattice QCD and is widely used in this field.

IV. HYBRID (OR HAMILTONIAN) MONTE CARLO

As mentioned before, the main problem of Metropolis-type algorithms, relies on the choice of the candidate point at each move of the chain. Choosing a trial point far from the initial one will lead to a large change in the distribution to sample, and thus to a small acceptance probability, while choosing a point close to the initial one will not lead to an efficient exploration of the parameter space, and thus to a slow convergence of the chain.

Hamiltonian (or “hybrid”) dynamics [21], developed originally for lattice field theory, is used to produce candidate proposals for Metropolis algorithm, in a very elegant and efficient way. It is an exact algorithm which combines molecular dynamics evolution with a Metropolis accept/reject step. This latter is used to correct for discretization errors in the numerical integration of the corresponding equations of motion. Very good reviews and papers exist which detail the properties of this algorithm (see for instance [16]), and only the main ideas will be recalled here for completeness.

To implement hybrid Monte Carlo algorithm, one introduces for each set of parameters \hat{q} (see previous section) a set of conjugate momenta \hat{p} and associates to this joint state of “position” \hat{q} and “momentum” \hat{p} an Hamiltonian $H(\hat{q}, \hat{p}) = \hat{p}^T M^{-1} \hat{p} / 2 + \mathcal{U}(\hat{q})$, where M is a mass matrix, generally taken to be diagonal, and $\mathcal{U}(\hat{q})$ an arbitrary potential energy. This allows us to define a joint distribution as

$$P(\hat{q}, \hat{p}) = \frac{1}{Z} e^{-H(\hat{q}, \hat{p})} = \frac{1}{Z} e^{-\mathcal{K}(\hat{p})} e^{-\mathcal{U}(\hat{q})}$$

where Z is the normalizing constant. We use for the potential energy $\mathcal{U}(\hat{q}) = -\log[P(D|\hat{q})P(\hat{q})]$. Starting from a point \hat{q}_0 of the chain, the HMC procedure consists in selecting some initial momenta \hat{p}_0 normally distributed around zero and let the system evolve deterministically for some time according to Hamilton’s equations of motion for $H(\hat{q}, \hat{p})$. It reaches a candidate point (\hat{q}_1, \hat{p}_1)

which, according to Metropolis procedure described above, is accepted with probability $\min(1, e^{-\Delta H})$. Since the dynamics conserves energy, i.e. $\Delta H = 0$ along a trajectory, the acceptance rate is 100%, independently of the dimension of the vector \hat{q} .

In practice, this acceptance is degraded because of the numerical resolution of Hamilton equations, but remains still at a very high level (typically of the order of 70%–90%, independently of the dimension of the chain). HMC algorithm is thus very well suited to multiparameter determination. To discretize Hamilton’s equation, we use the LeapFrog method, a convenient second order integration method that gives the time reversal invariance needed for the Metropolis transition kernel.

We have implemented both Metropolis and hybrid Hamiltonian Monte Carlo algorithms in the open-source package HeraFitter and its successor xFitter [17]. This software provides a modular framework to determine PDFs by fitting a large ensemble of experimental data. In what follows, we focus on the proton PDFs, and we use for the PDF parametrization, the HERAPDF functional form, that we just recall here for the sake of clarity: the parametrized HERA PDFs are the valence distribution xu_v and xd_v , the gluon distribution xg , and the \bar{U} and \bar{D} distribution defined as $x\bar{U} = x\bar{u}$, $x\bar{D} = x\bar{d} + x\bar{s}$. Their functional form reads

$$xf_a(x) = A_a x^{B_a} (1-x)^{C_a} (1 + D_a x + E_a x^2), \quad (3)$$

where a labels a parton ($g, u_{\text{val}}, d_{\text{val}}, \dots$. See [17] for more details). The analysis procedure we apply to the Markov chain we have produced is explained in the next section.

V. MARKOV CHAIN ANALYSIS

Assessing statistical errors for observables in Monte Carlo simulations is a subtle task and requires a careful treatment of the Markov chain. This section presents the different stages of analysis and the checks we have performed.

The procedure to analyze a Markov chain consists of several steps. In particular, it is necessary to remove the thermalization (or burn-in) region, to verify the convergence of the chain and to properly examine correlations between neighboring points in the chain. We have also checked the chain reversibility and the fact that the distribution was correctly sampled.

A. Thermalization

The thermalization time (or burn-in length) b of a Markov chain corresponds to a number of states $\{\hat{q}_t\}_{t=1, \dots, b}$ to be discarded from the beginning so that the chain forgets its starting point. It can be estimated as being the first state of the random walk—that is the first set of parameters \hat{q} , denoted \hat{q}_b —reaching the median value

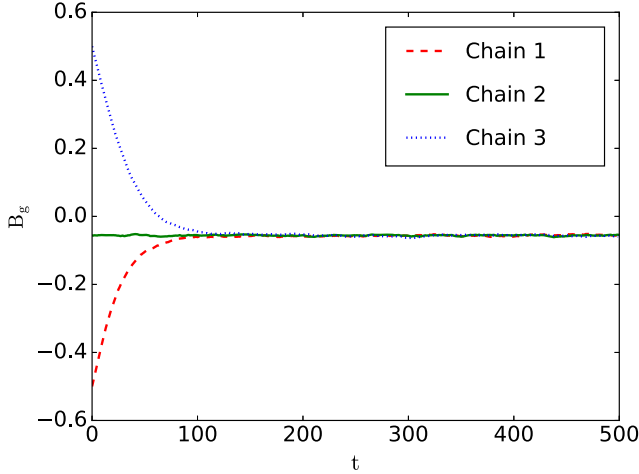


FIG. 1. Values of the parameter B_g as a function of the Monte Carlo time for three independent Markov chains. The green solid line represents a chain starting from the value given by MINUIT minimization, whereas the two other chains start from values much higher (blue dotted) or much lower (red dashed). The initial corresponding $\chi^2/\text{d.o.f.}$ values are, from chain 1 to chain 3, respectively, $\chi^2/\text{d.o.f.} = 67.44$, $\chi^2/\text{d.o.f.} = 0.87$ and $\chi^2/\text{d.o.f.} = 81.58$. We identify clearly on this plot the thermalization region, which is limited to the first ~ 100 – 210 iterations.

of its target distribution $P_{1/2}$ computed using the entire chain, i.e.

$$P(\hat{q}_b|D) > P_{1/2}. \quad (4)$$

To illustrate thermalization features, we have represented in Fig. 1 the Monte Carlo history of parameter B_g for three independent chains, each starting from a different point. For illustration purposes, we have fixed all parameters to their value given by the standard minimization procedure of xFitter, except this parameter B_g , which is obtained by a Monte Carlo procedure.

We can identify in Fig. 1 the thermalization region, whose extent depends on the starting point. The chain represented in solid green line has been started from the output value of a MINUIT minimization of the χ^2 with respect to parameter B_g and is thus thermalized very quickly, specifically after one iteration, whereas the other chains, started far from the minimum χ^2 , exhibits a thermalization of about 150 iterations for the chain represented in dashed red line and 210 iterations for the one represented by the dotted blue line; as expected, the farther from the minimum the starting point is, the longer the thermalization. Starting from a point far from its value given by the minimization procedure is useful to check that this latter did not get stuck in a local minimum and that simulations starting from different points converge to the same region (see also Sec. VC). In practice, we have generated several chains (36 chains, to be more precise),

starting from random points. We determined the thermalization time using Eq. (4) for each chain, and we removed from each chain its burn-in length.

B. Treatment of autocorrelations

By construction of a Markov chain, the state \hat{q}_t depends strongly on the state \hat{q}_{t-1} and quantities computed from this chain form themselves a Markov chain with inherent correlations from one member to the next. These type of correlations are often referred to as “autocorrelations” in simulation time.

Let us consider that we would like to extract an observable O from a Markov chain simulation with N points. For this estimation, we use the N successive Monte Carlo estimates O_t (we assume in what follows, that the thermalization region has already been discarded, i.e. that the chain has been equilibrated before recording data) and we compute the usual mean $\langle O \rangle$ where $\langle \cdot \rangle$ means averaging over the N data points. The usual estimate of root-mean-square deviation of this average can be computed as

$$\sigma_{\text{naive}}^2 = \frac{N}{N-1} (\langle O^2 \rangle - \langle O \rangle^2)$$

This “naïve” error relies on the assumption that the measurements performed on the Markov chain are not correlated, which is in general not true. In order to account for the correlations, one can introduce for the given observable O , the integrated autocorrelation time τ_{int} , which can be defined as follows,

$$\tau_{\text{int}} = \frac{1}{2} \sum_{-\infty}^{+\infty} \rho(s),$$

where $\rho(s)$ is the normalized autocorrelation function,

$$\rho(s) = \frac{(O_t - \langle O \rangle)(O_{t+s} - \langle O \rangle)}{(O_t - \langle O \rangle)^2}.$$

The dependence of $\rho(s)$ on the time separation s only is a consequence of the chain being in equilibrium. The integrated autocorrelation time controls the statistical error in Monte Carlo measurement of $\langle O \rangle$ and there are mostly two possibilities to incorporate this autocorrelation time in the assessment of the statistical errors. The first one consists in leaving out $2\tau_{\text{int}}$ points between two effective points, or in other terms, to do a subsampling by rejecting all states which are closer than $2\tau_{\text{int}}$ to each other, in order to get independent states. This approach has the disadvantage of requiring the *a priori* knowledge of τ_{int} . The second approach consists in keeping all measurements, but taking into account the autocorrelation time to estimate the statistical errors. The statistical error of correlated measurements can indeed be computed by [10,22]

$$\sigma_\tau^2 = 2\tau_{\text{int}}\sigma_{\text{naive}}^2$$

This means that the number of “effectively independent data points” in a run of length N is roughly $N/(2\tau_{\text{int}})$. If the integrated autocorrelation time is used to assess statistical errors, this means of course that a reliable estimate of τ_{int} and its error itself are needed. Such estimates require a delicate procedure. An efficient method—called Γ -method—has been developed in [22], which relies on the explicit determination of autocorrelation functions and autocorrelation times. This method provides not only numerical estimators of the integrated autocorrelation time, but also estimates for mean values and statistical errors for arbitrary functions of elementary observables in Monte Carlo simulations. We refer the interested reader to [22] and references therein for details. We have used the Γ -method both to obtain the autocorrelation time and to compute observables.

A further method to reliably estimate the error on uncorrelated measurements is the so-called “*jackknife binning*” [23]. It consists in building N subsets of data from the initial ensemble of size N , by removing one observation, leaving samples of size $N - 1$. Preaveraging over the blocks of data provides N estimates of the average:

$$\langle O \rangle_B = \frac{1}{N-1} \left(\sum_{t=1}^{(B-1)} O_t + \sum_{t=B+1}^N O_t \right), \quad B = 1, \dots, N.$$

The jackknife mean and variance for the observable O are then constructed from

$$\langle O \rangle_{\text{Jack}} = \frac{1}{N} \sum_{B=1}^N \langle O \rangle_B,$$

$$\sigma_{\text{Jack}}^2 = \frac{N-1}{N} \sum_{B=1}^N (\langle O \rangle_B - \langle O \rangle_{\text{Jack}})^2.$$

The jackknife method—and its extensions—is a widely used procedure, in particular in Lattice QCD. For cross-checks and comparison, we applied in our analysis both Γ -method and jackknife binning techniques. For this latter, to un-correlate the points of a given chain, we performed a subsampling of this chain using the value of the autocorrelation time provided by the Γ -method.

C. Reversibility and convergence

We have verified that our implementation of HMC algorithm satisfies reversibility with a very good precision (relative accuracy better than 10^{-6}) and that the average acceptance—computed using the jackknife method, after removing thermalization region and decorrelating the chain—is $\langle e^{-\Delta H} \rangle = 1.002 \pm 0.016$, thus insuring that our chains indeed converge towards a stationary distribution. In addition, to exclude the risk of a nonidentified lack of convergence, we have simulated several chains, with different (and random) starting points. This is illustrated in Fig. 2 in the case of one varying parameter, namely B_g (all others being fixed to their value obtained by standard χ^2 minimization procedure), where the chains are clearly seen to converge towards the same stationary distribution.

The results displayed in the next section have been obtained after skipping thermalization and properly taking

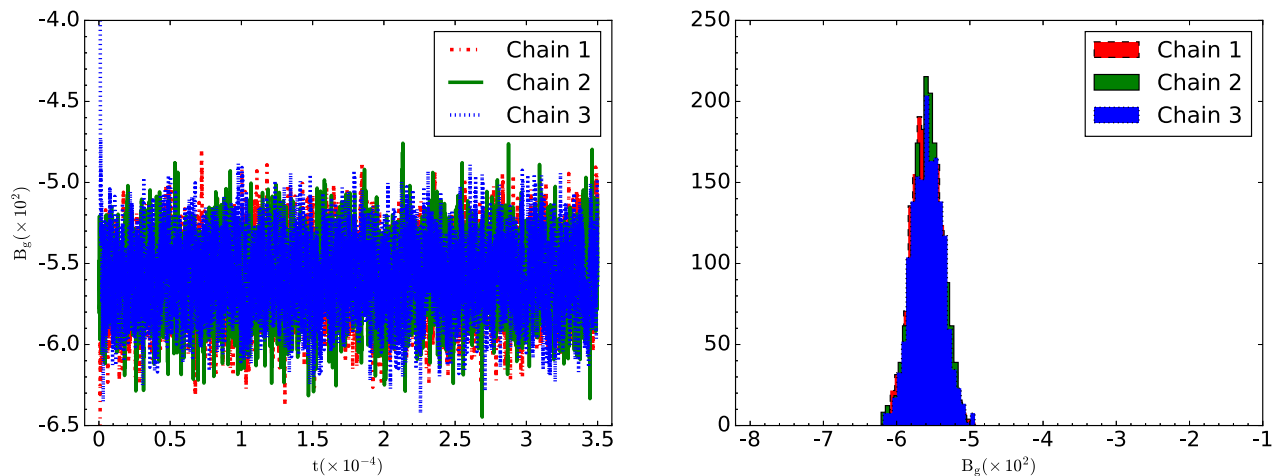


FIG. 2. Values of the parameter B_g as a function of the Monte Carlo time for three independent Markov chains (lhs). The starting points of the chains and the color code are the same as in Fig. 1. The chains are clearly converging towards the same stationary distribution, what is confirmed by plotting the parameter distribution for each chain (rhs), after removing thermalization points and taking into account autocorrelation.

into account the autocorrelation, using either the Γ -method or the jackknife binning procedure as explained above.

VI. PRELIMINARY RESULTS

A. Setup and simulation parameters

The results shown in this section are obtained from a Markov chain using the HERAPDF functional forms for initial PDFs at a scale $Q_0 = 1.9 \text{ GeV}^2$ with ten free parameters: $B_g, C_g, B_{u_{\text{val}}}, C_{u_{\text{val}}}, E_{u_{\text{val}}}, C_{d_{\text{val}}}, C_{\bar{u}}, A_{\bar{D}}, B_{\bar{D}}$ and $C_{\bar{D}}$ (see expression (3) for the definition of these parameters, and [17] for more details). We have used uniform priors for the parameters, and we consider the same data ensembles than the ones used to produce HERAPDF1.0 distributions.

These data are a combination of inclusive deep inelastic scattering cross sections measured by the H1 and ZEUS Collaborations in neutral and charged current unpolarized $e^\pm p$ scattering at HERA, during the period 1994–2000 [17]. Other settings and cuts—with the exception of the heavy flavor scheme, see below—were also identical to the ones of HERAPDF1.0 distributions. In particular, we do not rely of K factors, nor on grids techniques. The latter could, however, interestingly be used to speed up the computation.²

Since the aim of this work is to demonstrate the applicability of MCMC methods to PDF determination—rather than producing competitive PDF sets—we apply the ZMVFN scheme in order to speed up the computation of the χ^2 , and a lower cut on the data at $Q_{\text{min}}^2 = 10 \text{ GeV}^2$. These settings give a total number of data points of 537. We will denote by “HERAPDF1.0 ZMVFNs” the PDFs thus obtained by minimization.

As already mentioned in the previous section, we have generated 36 Monte Carlo chains, each chain starting from a different random point, using the HMC algorithm.

The HMC algorithm requires the tuning of essentially two parameters: the number of leapfrog steps L and the step size ε , this latter potentially depending on the direction in the parameter space. These two quantities are chosen such as to keep both the acceptance high (requiring small ε , to minimize the numerical errors in solving Hamilton equations), and the correlation between two successive Monte Carlo iterations small (thus requiring large trajectory length $L\varepsilon$). We have chosen $L = 100$ and one leapfrog step size for each parameter, depending on the parameter typical standard deviation. Namely we took $\varepsilon_i = 3 \cdot 10^{-2} \Delta q_i^{\text{typ}}$, where Δq_i^{typ} is, for each parameter, the value of its standard deviation provided by the minimization. With these HMC parameters, we obtain an acceptance of 80% and chains which have almost no correlation, since the integrated

²and we are exploring this possibility for more complete computations.

TABLE I. Comparison of mean values and their statistical errors obtained for PDF parameters using two different analysis methods. The jackknife binning has been applied after subsampling the chain, selecting points not closer than $2\tau_{\text{int}} = 4$ from each other. Since the number of points considered for the jackknife analysis is thus four times less than for the Γ -method, the errors are expected to be smaller by roughly a factor 2 for this latter procedure, which is indeed the case.

Parameter	Γ -method	Jackknife binning
B_g	-0.0537 ± 0.0001	-0.0537 ± 0.0002
C_g	5.9449 ± 0.0015	5.9483 ± 0.0025
$B_{u_{\text{val}}}$	0.6124 ± 0.0001	0.6125 ± 0.0002
$C_{u_{\text{val}}}$	4.7458 ± 0.0003	4.7455 ± 0.0006
$E_{u_{\text{val}}}$	14.965 ± 0.008	14.961 ± 0.012
$C_{d_{\text{val}}}$	3.2054 ± 0.0014	3.2077 ± 0.0016
$C_{\bar{u}}$	4.0917 ± 0.0038	4.0961 ± 0.0048
$A_{\bar{D}}$	0.3096 ± 0.0002	0.3098 ± 0.0002
$B_{\bar{D}}$	-0.0174 ± 0.0001	-0.0173 ± 0.0002
$C_{\bar{D}}$	6.2203 ± 0.0054	6.2096 ± 0.0076

autocorrelation time τ_{int} computed by the Γ -method is less than 2 for all parameters.

The HMC algorithm also requires the computation of the potential energy (that is in our case the χ^2) with respect to the parameters. These derivatives are computed numerically, using a symmetric derivative. We thus need, for 10 parameters, 20 evaluations of the χ^2 for each step of the Leapfrog algorithm. We have run 36 jobs in parallel, and collected the results after three days of running. We computed for each of our 36 chains the burn-in length, and we removed the maximum burn-in (namely 28) to all of them, to obtain a total of 4 400 points per chain. To analyze these chains with the Γ -method, we kept all 158 400 points, while the jackknife analysis was done considering one point every four (i.e. $2\tau_{\text{int}}$), that is 39 600 points.³

B. PDF parameter values, marginal distributions and correlations

Table I compares the mean value and its statistical error for each of the ten PDF parameters under consideration, using the two analysis procedures we have presented above, namely Γ -method and jackknife binning. The computation of the integrated autocorrelation time by the Γ -method gives values of τ_{int} less than 2 and we have used for the jackknife one point of the chain every four. As can be seen from Table I, both methods give very close results, showing that we have analyze our Markov chain in a

³For a fully realistic PDF determination, systematic uncertainties (factorization and renormalization scales, heavy quark treatment, ...) are much bigger than the sub-percent accuracy we obtain with such a long Markov chain. In realistic cases, a Markov chain with about a thousand decorrelated points will lead to results statistically accurate enough.

TABLE II. Comparison of results obtained for PDF parameter values extracted from an independent Markov chain of length 39 600, and the results provided from a MINUIT minimization. We compute from the MCMC the best fit value, the mean value and the standard deviation value for each of the 10 parameters considered, together with their statistical errors estimated by jackknife binning method. The standard deviation given by the minimization is the usual one-sigma deviation.

Parameter	Values	MCMC	MINUIT minimization
\mathbf{B}_g	Mean	-0.0537 ± 0.0002	
	Best fit	-0.0632 ± 0.0168	-0.0559
	Standard deviation	0.0299 ± 0.0001	0.0288
\mathbf{C}_g	Mean	5.9483 ± 0.0025	
	Best fit	5.8952 ± 0.0615	5.9274
	Standard deviation	0.5037 ± 0.0019	0.5078
$\mathbf{B}_{u_{val}}$	Mean	0.6125 ± 0.0002	
	Best fit	0.6092 ± 0.0121	0.6098
	Standard deviation	0.0371 ± 0.0001	0.0389
$\mathbf{C}_{u_{val}}$	Mean	4.7455 ± 0.0006	
	Best fit	4.7467 ± 0.0525	4.7122
	Standard deviation	0.1280 ± 0.0005	0.1332
$\mathbf{E}_{u_{val}}$	Mean	14.961 ± 0.012	
	Best fit	15.42 ± 0.94	14.76
	Standard deviation	2.494 ± 0.010	2.571
$\mathbf{C}_{d_{val}}$	Mean	3.2077 ± 0.0016	
	Best fit	3.084 ± 0.076	3.143
	Standard deviation	0.3183 ± 0.0016	0.2830
$\mathbf{C}_{\bar{U}}$	Mean	4.0961 ± 0.0048	
	Best fit	4.323 ± 0.814	4.052
	Standard deviation	0.9617 ± 0.0035	0.9782
$\mathbf{A}_{\bar{D}}$	Mean	0.3098 ± 0.0002	
	Best fit	0.320 ± 0.039	0.305
	Standard deviation	0.0485 ± 0.0002	0.0488
$\mathbf{B}_{\bar{D}}$	Mean	-0.0173 ± 0.0002	
	Best fit	-0.0097 ± 0.0237	-0.0178
	Standard deviation	0.0304 ± 0.0001	0.0306
$\mathbf{C}_{\bar{D}}$	Mean	6.2096 ± 0.0076	
	Best fit	5.888 ± 0.142	5.875
	Standard deviation	1.505 ± 0.009	1.290

consistent way. For the rest of this paper, we will thus display only the results obtained using jackknife binning techniques. We also notice that for the chain length considered ($\sim 40\,000$ thermalized and decorrelated points), the statistical errors on the mean values are tiny.

In Table II are displayed the results provided by MCMC method—using jackknife binning for error estimate—for the parameters mean, best fit⁴ and standard deviation,⁵ compared with the output of the standard MINUIT minimization. We notice, as already stated above, that the determination of parameters by Monte Carlo methods, gives much more information than a standard minimization. We can in particular extract the statistical errors on the quantities we are interested in and this error decreases with the length of the Markov chain. As seen from the

table, the best fit value extracted from the parameter probability distributions are compatible with the parameter values provided by the minimization procedure. For what concerns standard deviations however, although MCMC and minimization gives similar results, no precise comparison can be made since the minimization does not provide estimates of errors for the usual one-standard deviation of the parameters. In addition, both quantities (MCMC standard deviation and minimization deviation) should coincide only if the probability density of the parameter considered is Gaussian, which is not necessarily true (see below). This fact is already visible in Table II, where we can see that the mean and best fit values do not coincide within errors for some of the parameters ($C_{d_{val}}$, $C_{\bar{D}}$ for instance).

The probability distribution functions of the parameters, together with the two-dimensional correlation plots between parameters are displayed in Fig. 3. The marginal posterior parameter distributions are shown on the diagonal graphs, and 2D-correlations on the off-diagonal plots.

⁴The parameter best fit values are the parameters values that minimized the χ^2 function.

⁵we have computed here the corrected sample standard deviation

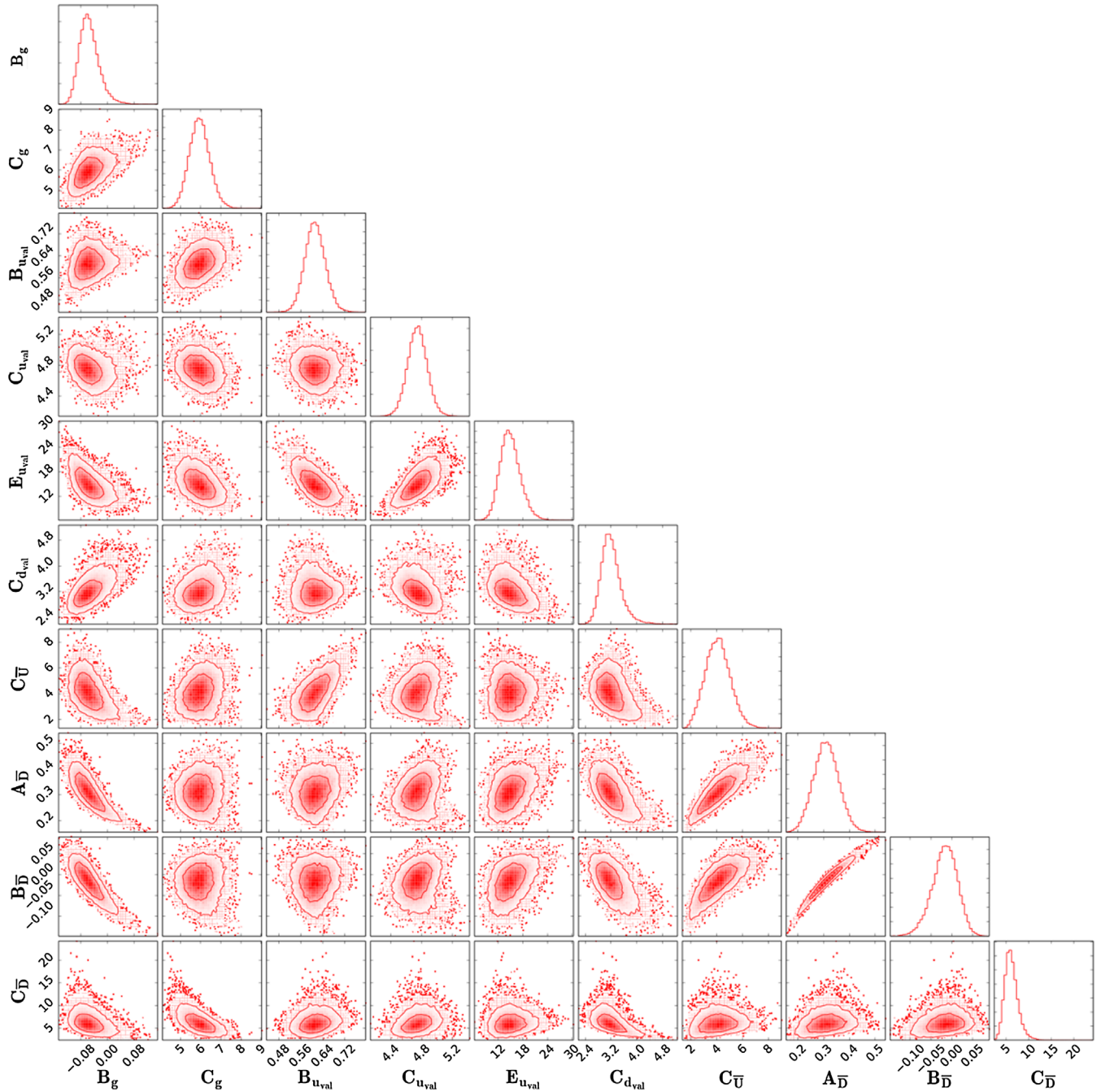


FIG. 3. Probability distribution functions of the PDF parameters (diagonal) and 2D correlation plots between parameters (off-diagonal).

The inner and outer contours of these latter are taken to be regions containing respectively 68% and 95% of the probability density. We note that the probability distribution of some parameters cannot properly be described by a Gaussian law, as illustrated in Fig. 4, and are even non-symmetric. We have also checked that our correlation plots are compatible with the values provided by the covariant matrix.

In Fig. 5 is shown the $\chi^2/\text{d.o.f.}$ distribution for our ten-dimensional Monte Carlo chain. The solid red line is an

adjustment with a χ^2 distribution law with 10 degrees of freedom, which perfectly describes our results. The fact that the χ^2 function built in Eq. (2) follows a χ^2 distribution law with the expected number of degrees of freedom is a strong indication (though not a formal proof) that our assumptions concerning the fluctuations of the experimental data points around their corresponding theoretical values, are justified.

Though this is not the case in the example we present, we would also like to note that potential flat directions in the

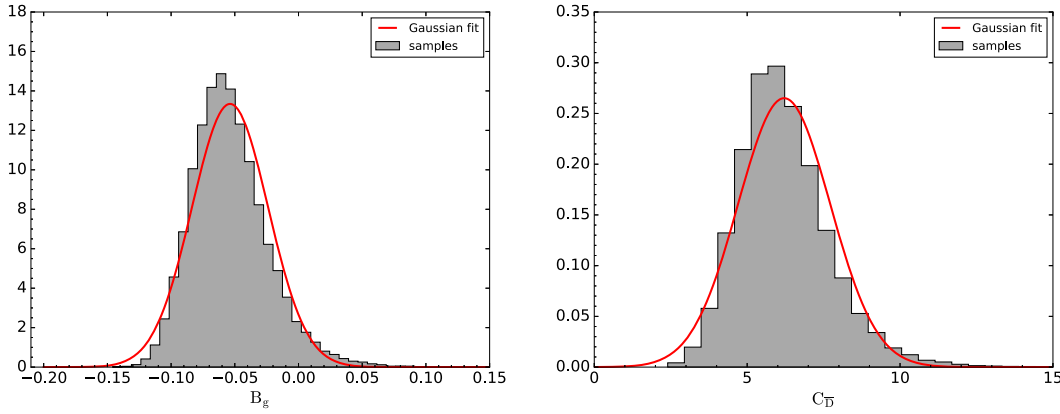


FIG. 4. Marginal probability distribution of parameters B_g (lhs) and $C_{\bar{D}}$ (rhs). They do not follow a Gaussian law, as can be seen from the gaussian fit (solid red line).

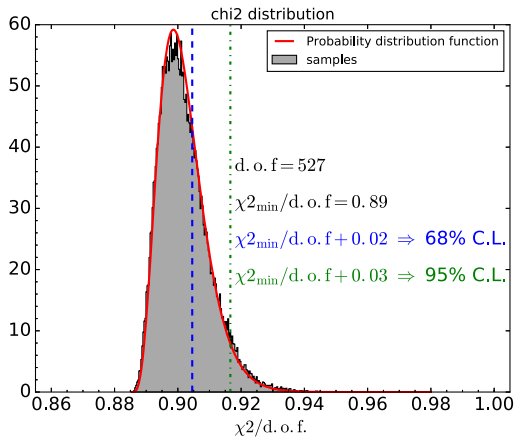


FIG. 5. χ^2 distribution for a 10D Monte Carlo chain. The solid red line is an adjustment of these data with a χ^2 distribution law with 10 degrees of freedom. The dashed (dashdot) vertical line indicates the 68% (95%) confidence limit.

parameter space are less problematic for MCMC method than for minimization techniques.

We need now to calculate, from the Markov chain of parameters, the parton distribution functions we are interested in. The procedure we use is explained in the next section and is more generally valid for any observable we want to compute from the MCMC.

C. PDF marginal distributions and confidence interval

To extract parton distribution functions from the Markov chain, we compute, from the set of 10 parameters obtained at each Monte Carlo iteration, the corresponding PDFs for a range of x and Q^2 values. This provides the marginal probability density functions of PDFs at fixed (x, Q^2) , as illustrated on Fig. 6 for the gluon, for two different x values.

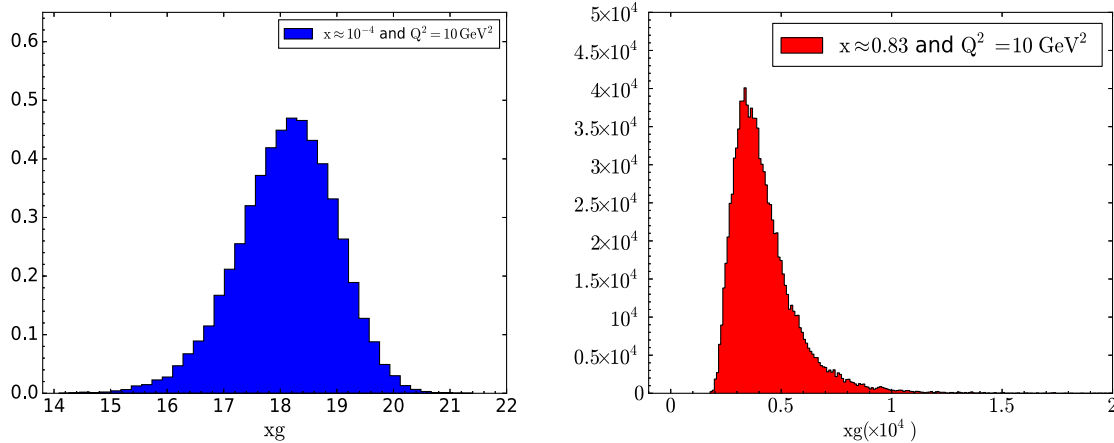


FIG. 6. Gluon PDF probability distribution function for $x \approx 10^{-4}$ (lhs) and $x \approx 0.83$ (rhs) at fixed $Q^2 = 10 \text{ GeV}^2$. The 68% confidence interval is obtained from this distribution, considering the region of the distribution containing 68% of the data remaining on each side of the best fit value.

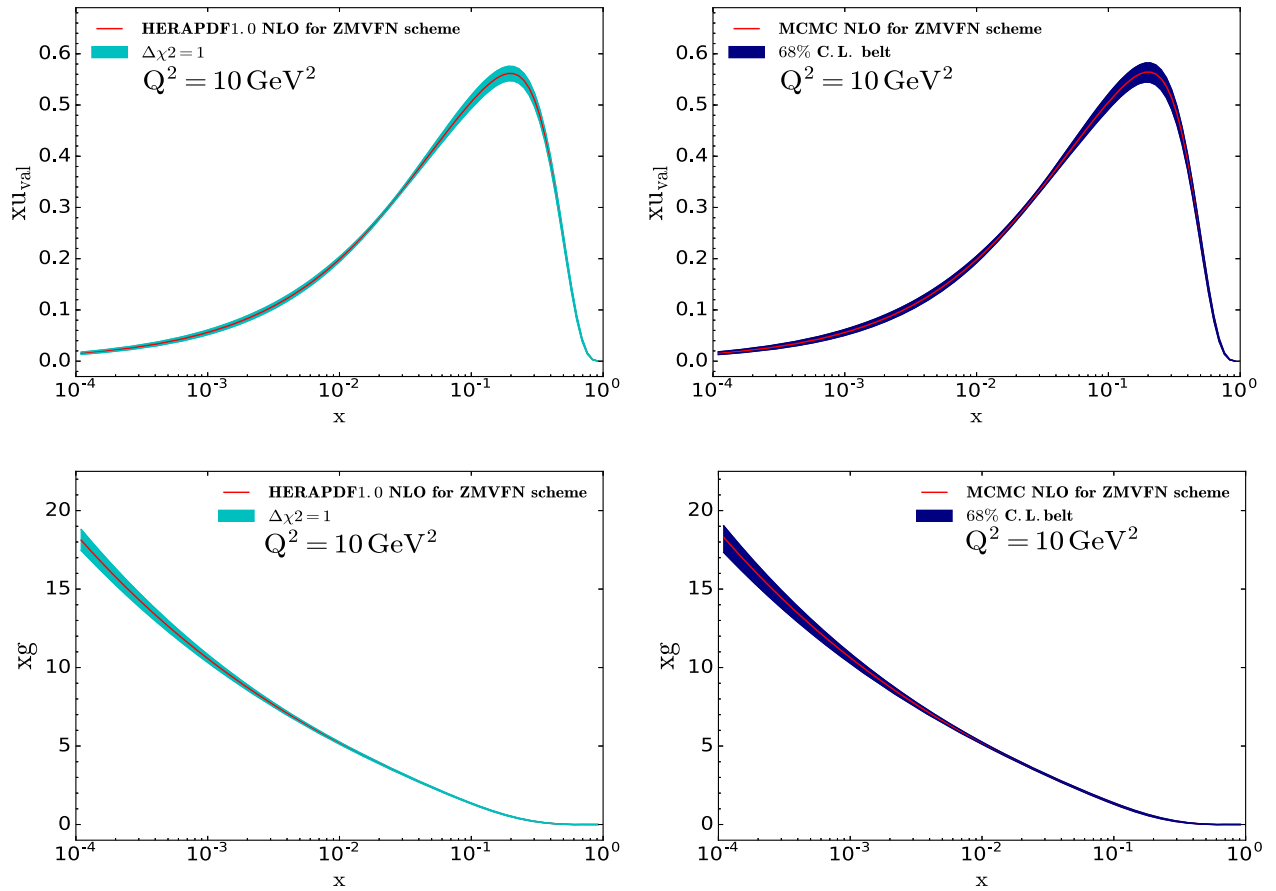


FIG. 7. The parton distribution functions obtained using MCMC (right) compared to HERAPDF1.0 (ZMVFN scheme) from xFitter output (left) for xu_{val} and xg , at $Q^2 = 10 \text{ GeV}^2$. The bands show the 68% confidence interval around the central value (in solid red line) for the MCMC PDFs, and the standard $\Delta\chi^2 = 1$ deviation for HERAPDF.

For each (x, Q^2) , we determine the $\alpha\%$ -confidence interval around the best fit value of the PDF (with typically $\alpha = 68$ or $\alpha = 95$) by considering the region of the distribution on each side of the best fit, and taking $\alpha\%$ of the data on each of these regions. This provides the (not necessarily symmetric) $\alpha\%$ -confidence limit intervals we show in Figs. 7 and 9 to 11.

The gluon and u_{val} parton distributions obtained this way are plotted as functions of x and for $Q^2 = 10 \text{ GeV}^2$ in Fig. 7. The central PDF is the best fit value. These MCMC PDFs are compared with the HERAPDF1.0 PDFs (ZMVFN scheme), with a direct comparison in Fig. 7, and a ratio plot in Fig. 8. They are, as expected, very close, both in central value and in confidence interval. Maximum likelihood estimator and least square method are indeed equivalent under Gaussian assumption, which in our case can be reasonably applied, as mentioned in the previous section.

The uncertainties obtained by the MCMC method and the Hessian method are also consistent within the kinematic range of HERA. This is demonstrated in Fig. 9 where experimental uncertainties—normalized by the best-fit

value—obtained for HERAPDF1.0 NLO and MCMC NLO, respectively, by the Hessian and MCMC methods are compared for the u_{val} and the gluon distributions.

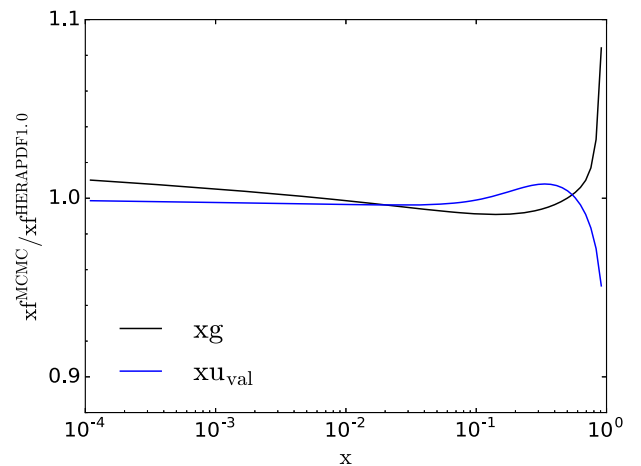


FIG. 8. Ratio of MCMC PDFs and HERAPDF1.0 (ZMVFN scheme) central values for xu_{val} and xg at $Q^2 = 10 \text{ GeV}^2$.

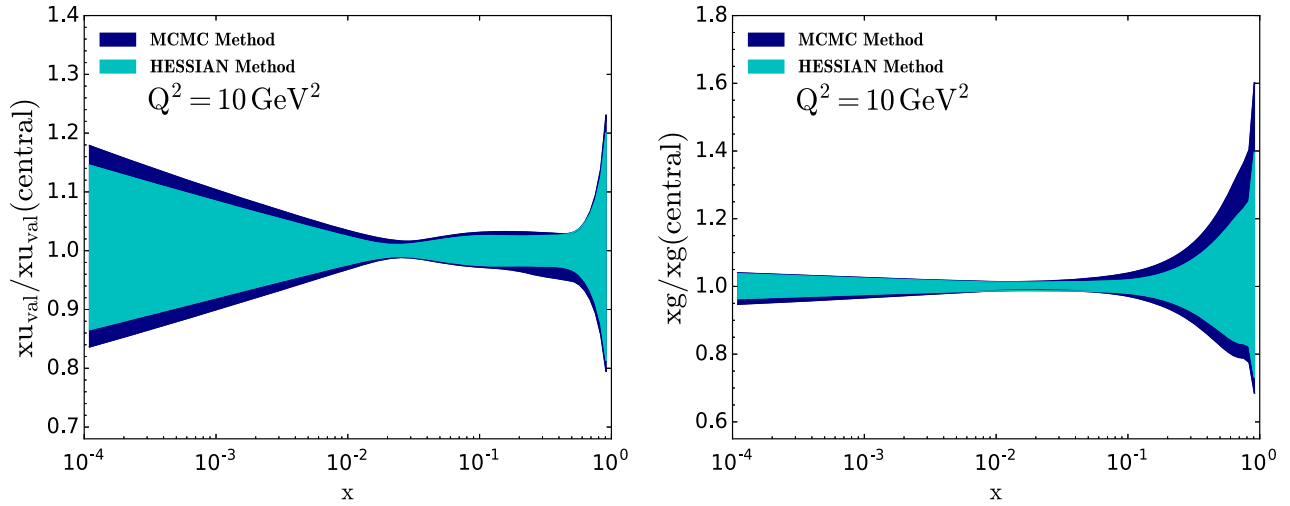


FIG. 9. Comparison of the PDF uncertainties, normalized by the best fit value, as determined by the Hessian and MCMC methods at NLO for the valence distribution xu_{val} and the gluon distribution xg , at a scale $Q^2 = 10 \text{ GeV}^2$.

The MCMC uncertainties tend to be slightly larger than the standard deviations obtained in the Hessian approach. For completeness, we also display in Fig. 10 the antiquark PDFs at $Q^2 = 10 \text{ GeV}^2$, and in Fig. 11 the central

value and 68% confidence limit interval for xu_{val} , xd_{val} , xg and $x\Sigma$ ($x\Sigma = x\bar{u} + x\bar{d} + x\bar{s} + x\bar{c}$, with $x\bar{c} = 0$ for $Q^2 < m_c^2$) on the same plot, at $Q^2 = 1.9 \text{ GeV}^2$ and $Q^2 = 10 \text{ GeV}^2$.

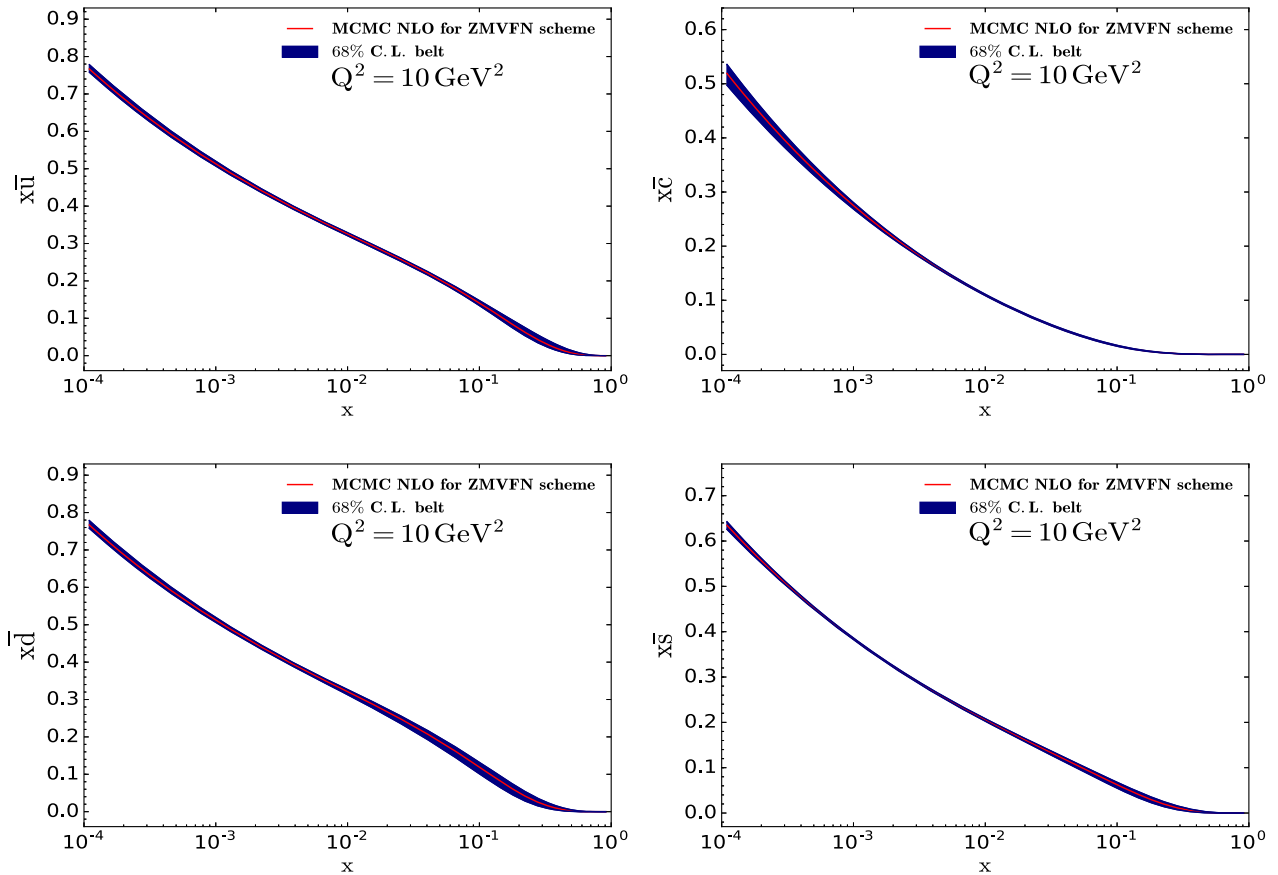


FIG. 10. The MCMC parton distribution functions $x\bar{u}$, $x\bar{d}$, $x\bar{s}$ and $x\bar{c}$ at $Q^2 = 10 \text{ GeV}^2$. The bands show the 68% confidence interval around the best fit value (in solid red line).

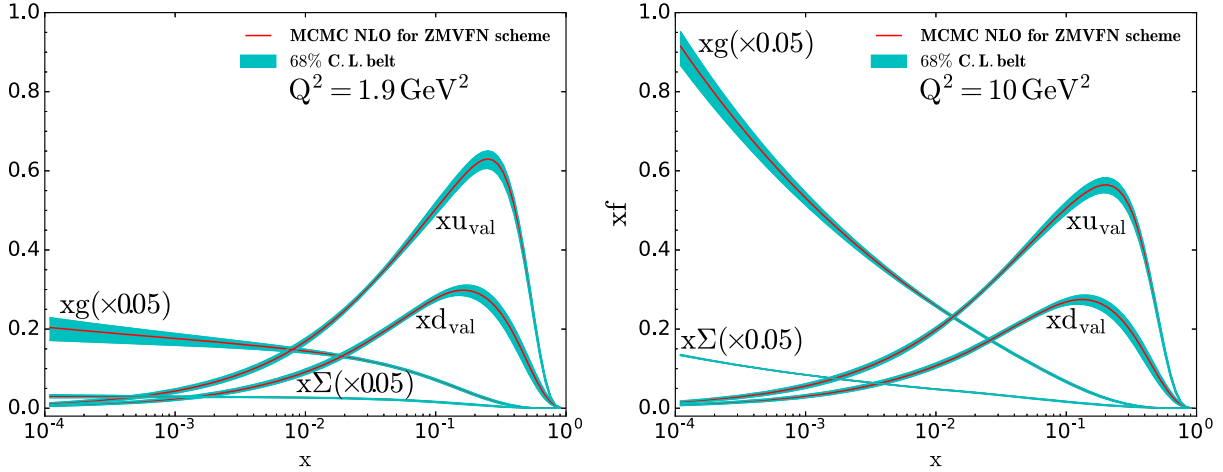


FIG. 11. PDFs of valence quarks ($x_{u_{val}}, x_{d_{val}}$), sea quarks ($x_{\Sigma} = x\bar{u} + x\bar{d} + x\bar{s} + x\bar{c}$, with $x\bar{c} = 0$ for $Q^2 < m_c^2$), and gluon (x_g) using MCMC at the scale $Q^2 = 1.9 \text{ GeV}^2$ (lhs) and $Q^2 = 10 \text{ GeV}^2$ (rhs) obtained with the ZMVFN scheme. The gluon and the sea distributions are scaled down by a factor of 20. The experimental uncertainties (68% confidence limit as defined in the text from the probability density of PDFs) are represented by the green-shaded region.

VII. CONCLUSION AND OUTLOOK

We have shown that MCMC, known to be well suited to multiparameter determination, is applicable to PDF determination and that Bayesian parameter inference approach applied to global PDF analysis can lead to a deeper insight into PDF uncertainties. The innovative procedure we implemented, which combines Monte Carlo techniques, lattice-developed algorithms and global PDF analysis is complementary to the existing methods. We have for the first time applied the hybrid Monte Carlo algorithm to PDF determination and computed marginal probability densities of PDF parameters, and the PDFs themselves. This allows us to study the probability distribution of these functions, to determine mean, best fit and median values, and to extract confidence intervals in a statistically controlled way.

This work will lead to an innovative PDF uncertainties determination, and thus to a reliable determination of uncertainties for many collider observables, in a way that is complementary to the existing methods. This will also open new ways to analyze the impact of new data sets being added to the analysis, to check which data set has outliers and if these latter can be tolerated.

This feasibility study paves the way for a more complete PDF determination by MCMC techniques, and our goal is to extend the present work to the full ensemble of PDF free parameters, including also as parameters, the strong coupling constant and c and b quark masses. We will consider more complex χ^2 functions including correlation and complete our analysis on a fully realistic case, studying, in particular, the impact of priors. No doubt that Markov chain Monte Carlo methods will give interesting and valuable information on PDFs and will contribute to our deeper understanding of these key elements of QCD.

ACKNOWLEDGMENTS

We thank L. Derome and D. Maurin for fruitful discussions and the xFITTER support team for their help. We are also grateful to the CFM Foundation for Research for its financial support through the Jean-Pierre Aguilar 2014 grant. Most of the computation has been done in Lyon-CCIN2P3, thanks to the CPUs resources provided. We are grateful to the staff members of this Computing Center for their constant support.

- [1] X. Ji, *Phys. Rev. Lett.* **110**, 262002 (2013).
 [2] H. L. Lai, J. Huston, S. Kuhlmann, F. Olness, J. Owens, D. Soper, W. K. Tung, and H. Weerts, *Phys. Rev. D* **55**, 1280 (1997).
 [3] A. D. Martin, R. G. Roberts, M. G. Ryskin, and W. J. Stirling, *Eur. Phys. J. C* **2**, 287 (1998); A. D. Martin,

- R. G. Roberts, W. J. Stirling, and R. S. Thorne, *Eur. Phys. J. C* **28**, 455 (2003); **35**, 325 (2004).
 [4] W. T. Giele and S. Keller, *Phys. Rev. D* **58**, 094023 (1998).
 [5] J. Huston, E. Kovacs, S. Kuhlmann, H. L. Lai, J. F. Owens, D. Soper, and W. K. Tung, *Phys. Rev. Lett.* **77**, 444 (1996).

- [6] J. Pumplín, D. Stump, R. Brock, D. Casey, J. Huston, J. Kalk, H. L. Lai, and W. K. Tung, *Phys. Rev. D* **65**, 014013 (2001); J. Pumplín, D. R. Stump, and W. K. Tung, *Phys. Rev. D* **65**, 014011 (2001); D. Stump, J. Pumplín, R. Brock, D. Casey, J. Huston, J. Kalk, H. L. Lai, and W. K. Tung, *Phys. Rev. D* **65**, 014012 (2001).
- [7] D. Stump, J. Pumplín, R. Brock, D. Casey, J. Huston, J. Kalk, H. L. Lai, and W. K. Tung, *Phys. Rev. D* **65**, 014012 (2001).
- [8] W. T. Giele and S. Keller, *Phys. Rev. D* **58**, 094023 (1998); Report No. FERMILAB-PUB-01-498-T.
- [9] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, *Markov Chain Monte Carlo In Practice* (Chapman and Hall/CRC, London, 1995).
- [10] A. D. Sokal, *Monte Carlo Methods in Statistical Mechanics: Foundations and new algorithms* (Cours de Troisième Cycle de la Physique en Suisse Romande, Lausanne, Switzerland, 1989).
- [11] A. Putze and L. Derome, *Phys. Dark Universe* **5–6**, 29 (2014); A. Putze, L. Derome, D. Maurin, L. Perotto, and R. Taillet, *Astron. Astrophys.* **497**, 991 (2009).
- [12] L. Del Debbio, S. Forte, J. I. Latorre, A. Piccione, and J. Rojo (NNPDF Collaboration), *J. High Energy Phys.* **03** (2007) 039; J. C. Rojo, PhD thesis, arXiv:hep-ph/0607122.
- [13] N. N. Nikolaev and B. Zakharov, *Z. Phys. C* **49**, 607 (1991); A. H. Mueller, *Nucl. Phys.* **B415**, 373 (1994).
- [14] J. Collins, *Foundations of Perturbative QCD*, vol. 32, Cambridge monographs on particle physics, nuclear physics and cosmology (Cambridge University Press, Cambridge, England, 2011).
- [15] S. M. Aybat and T. C. Rogers, *Phys. Rev. D* **83**, 114042 (2011); M. G. A. Buffing, A. Mukherjee, and P. J. Mulders, *Phys. Rev. D* **86**, 074030 (2012); P. J. Mulders, *Pramana* **72**, 83 (2009); S. Jadach and M. Skrzypek, *Acta Phys. Pol. B* **40**, 2071 (2009); F. Hautmann, *Acta Phys. Pol. B* **40**, 2139 (2009).
- [16] R. M. Neal, in *Handbook of Markov Chain Monte Carlo*, edited by S. Brooks, A. Gelman, G. L. Jones, and X. Meng (Chapman and Hall/CRC, London, 2011).
- [17] S. Alekhin *et al.*, *Eur. Phys. J. C* **75**, 304 (2015); F. D. Aaron *et al.* (H1 and ZEUS Collaboration), *J. High Energy Phys.* **01** (2010) 109; F. D. Aaron *et al.* (H1 Collaboration), *Eur. Phys. J.* **C64**, 561 (2009); M. Botje, *Comput. Phys. Commun.* **182**, 490 (2011); F. James, M. Roos (CERN), *Comput. Phys. Commun.* **10**, 343 (1975); C. Adloff *et al.* (H1 Collaboration), *Eur. Phys. J. C* **30**, 1 (2003); S. Chekanov *et al.* (ZEUS Collaboration), *Phys. Rev. D* **67**, 012007 (2003); F. D. Aaron *et al.*, *Eur. Phys. J. C* **63**, 625 (2009); S. Chekanov *et al.* (ZEUS Collaboration), *Eur. Phys. J. C* **42**, 1 (2005); J. Pumplín, D. R. Stump, J. Huston, H. L. Lai, P. M. Nadolsky, and W. K. Tung, *J. High Energy Phys.* **07** (2002) 012.
- [18] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, *J. Chem. Phys.* **21**, 1087 (1953).
- [19] W. K. Hastings, *Biometrika* **57**, 97 (1970).
- [20] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars, *Computational Geometry—Algorithms and Applications* (Springer-Verlag, Berlin, 2008), 2nd revised edition.
- [21] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth, *Phys. Lett. B* **195**, 216 (1987).
- [22] U. Wolff, *Comput. Phys. Commun.* **156**, 143 (2004).
- [23] M. H. Quenouille, *Biometrika* **43**, 353 (1956).