

Estimating nPDF Uncertainties via Markov Chain Monte Carlo Methods

Nasim Derakhshanian

nCTEQ collaboration

IFJ PAN, Krakow, Poland

DIS 2024

KISD
KRAKOW SCHOOL
OF INTERDISCIPLINARY
PHD STUDIES



nCTEQ
nuclear parton distribution functions

Global analysis of nPDF

Input function at Q_0

Parameterize nuclear PDF at initial scale: 1.3GeV



DGLAP evolution

Compute theory predictions at scale Q by solving DGLAP equations



Construct χ^2 function

Calculate the goodness of fit in terms of theory predictions, data and uncertainties



Minimization

Minimize χ^2 function with respect to nPDF parameters

Experimental data

Choose experimental data (e.g. DIS, DY, W/Z, etc.) and apply kinematical cuts



$$\chi^2(\{a_i\}) = \sum_j^N \left(\frac{(\text{data}_j - \text{theory}_j(\{a_i\}))}{\sigma_j} \right)^2$$



Uncertainties estimation



nPDF uncertainties estimation

The **Hessian** method is widely used for error estimation in both proton and nuclear PDFs.



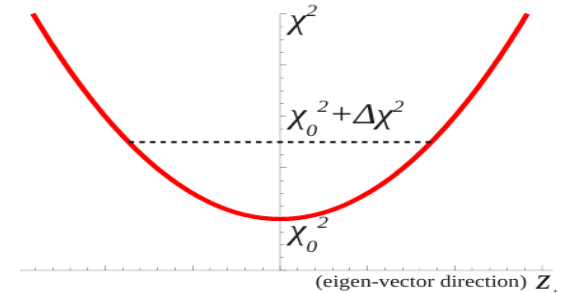
It relies on the quadratic behavior of the χ^2 function near the minimum.

Shortcomings:

- Non-gaussian errors
- Global minima judgment
- Choice of χ^2 tolerance

nPDF difficulties:

- Lacking data (range and precision of data for nuclei are generally lower than for proton)
- Complexity and nature of nuclear effects



Markov Chain Monte Carlo method

advanced statistical method as an alternative for Hessian

Global analysis of nPDF

Input function at Q_0

Parameterize bound proton PDF at initial scale: 1.3GeV

DGLAP evolution

Compute theory predictions at scale Q by solving DGLAP equations

Construct χ^2 function

Calculate the goodness of fit in terms of theory predictions, data and uncertainties

Minimization

Minimize χ^2 function with respect to nPDF parameters

Experimental data

Choose experimental data (e.g. DIS, DY, WZ production, etc.) and apply kinematical cuts

$$\chi^2(\{a_i\}) = \sum_j^N \left(\frac{(\text{data}_j - \text{theory}_j(\{a_i\}))}{\sigma_j} \right)^2$$

MCMC method

Uncertainties estimation

Markov Chain Monte Carlo (MCMC)

Markov Chain Monte Carlo (MCMC)

A sequence of random variables where the current value is dependent on the value of the prior variable (Memory-less property)

A technique for randomly sampling a probability distribution and approximating a desired quantity.

Bayes theorem:

$$p(\theta \mid \text{data}) = \frac{p(\text{data} \mid \theta) \cdot p(\theta)}{p(\text{data})}$$

Diagram illustrating Bayes' theorem with labels and arrows:

- Posterior: $p(\theta \mid \text{data})$
- Likelihood: $p(\text{data} \mid \theta)$
- Prior: $p(\theta)$
- Normalization: $p(\text{data})$

Prior: initial belief about the parameter before considering the data.

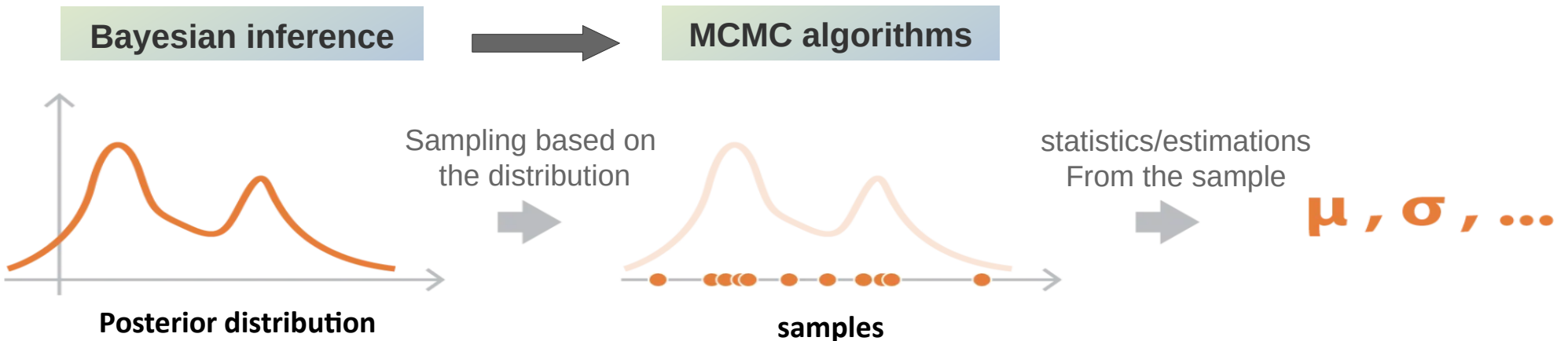
Likelihood: probability of observing the data given a specific value of the parameter.

Posterior: updated belief about the parameter given the data.

Markov Chain Monte Carlo (MCMC)

In this method, we aim to find the set of nPDF parameters that **maximizes the posterior** probability distribution given the experimental data.

Likelihood: $p(\text{data}|\theta) \propto \exp\left(-\frac{\chi^2}{2}\right)$



Markov Chain Monte Carlo (MCMC)

Metropolis algorithm:

Initialize parameters

for $i=1$ to $i=N$:

 multiplicity =1

 Proposing new parameters $\theta^* \sim q(\theta^*|\theta)$

 Compute acceptance probability

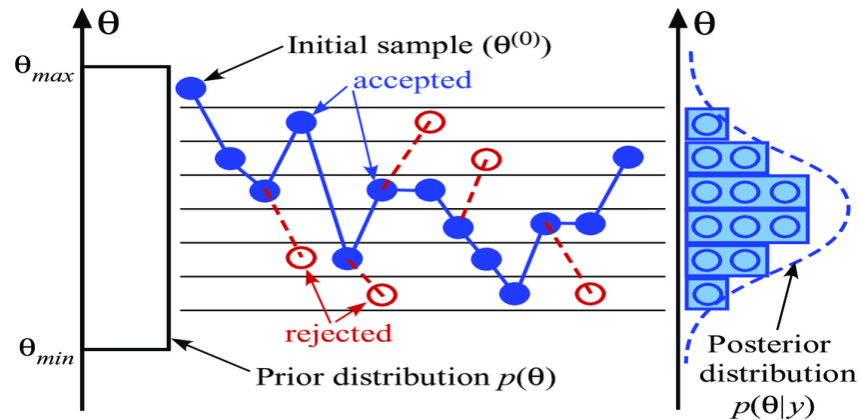
$$\alpha = \min(p(\theta^*|D)/p(\theta|D), 1)$$

 Sample from uniform distribution $u \sim \mathbf{U}(0, 1)$

 If $u < \min(1, \alpha)$ then $\theta_{i+1} = \theta^*$

 Else $\theta_{i+1} = \theta$ (multiplicity +=1)

- **Multiplicity**: the number of consecutive rejections of proposed points before an acceptance occurs.
- Each point in the chain represents a vector of the posterior parameter values.



nPDF fit setup

Fit properties:

- fit **NLO** QCD predictions
- using heavy quark effects (ACOT)
- Kinematic cuts: $Q > 2\text{GeV}$, $W > 3.5\text{GeV}$, $p_T > 3.0\text{ GeV}$
- NC & CC DIS, W/Z boson and Heavy Quark
- 10 free parameters: 2 gluon, 6 valence, 2 sea
- Parameterization:

- **Pb PDF fit**
- Multiple nuclei PDF fit

CJ15

Functional form for bound protons at Q_0 : $x f_i^{p/A}(x, Q_0) = c_0 x^{c_1} (1-x)^{c_2} (1 + c_3 \sqrt{x} + c_4 x)$

$$f_i^{(A,Z)} = \frac{Z}{A} f_i^{p/A} + \frac{A-Z}{A} f_i^{n/A}$$

Atomic number dependence:

$$c_k \rightarrow p_k + a_k \ln(A) + b_k \ln^2(A).$$

MCMC setup

Adaptive MH algorithm setup:

- ◆ The algorithm starts with a normal random-walk MH phase until N_0 samples have been generated

Proposal distribution: Multivariate Gaussian with fixed covariance C_0 $\mathbf{X}_{i+1} = \mathcal{N}(\mathbf{X}_i, C_0)$

- ◆ Then it switches to a self-learning proposal distribution

Adaptive proposal distribution: Multivariate Gaussian with self learned covariance C_i (covariance from collected samples so far)

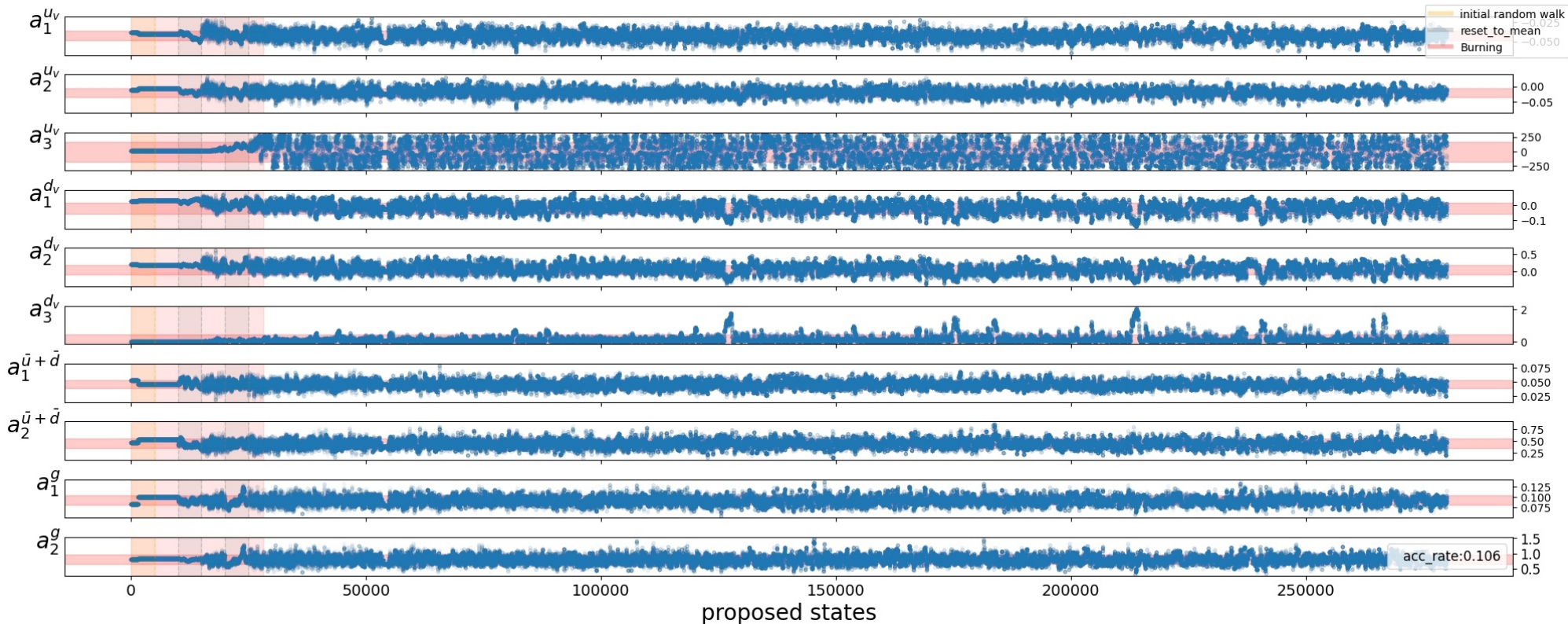
$$\mathbf{X}_{i+1} = (1 - \beta)\mathcal{N}(\mathbf{X}_i, \frac{(2.4)^2}{d} \cdot C_i) + \beta\mathcal{N}(\mathbf{X}_i, C_0)$$

- ◆ To boost the convergence, the algorithm restarts from its current mean value*

*The fixed covariance matrix is first given by a fraction of initial parameter values and then after restarting, it adjusts to the fraction of diagonal elements in the current self-learned covariance C_i

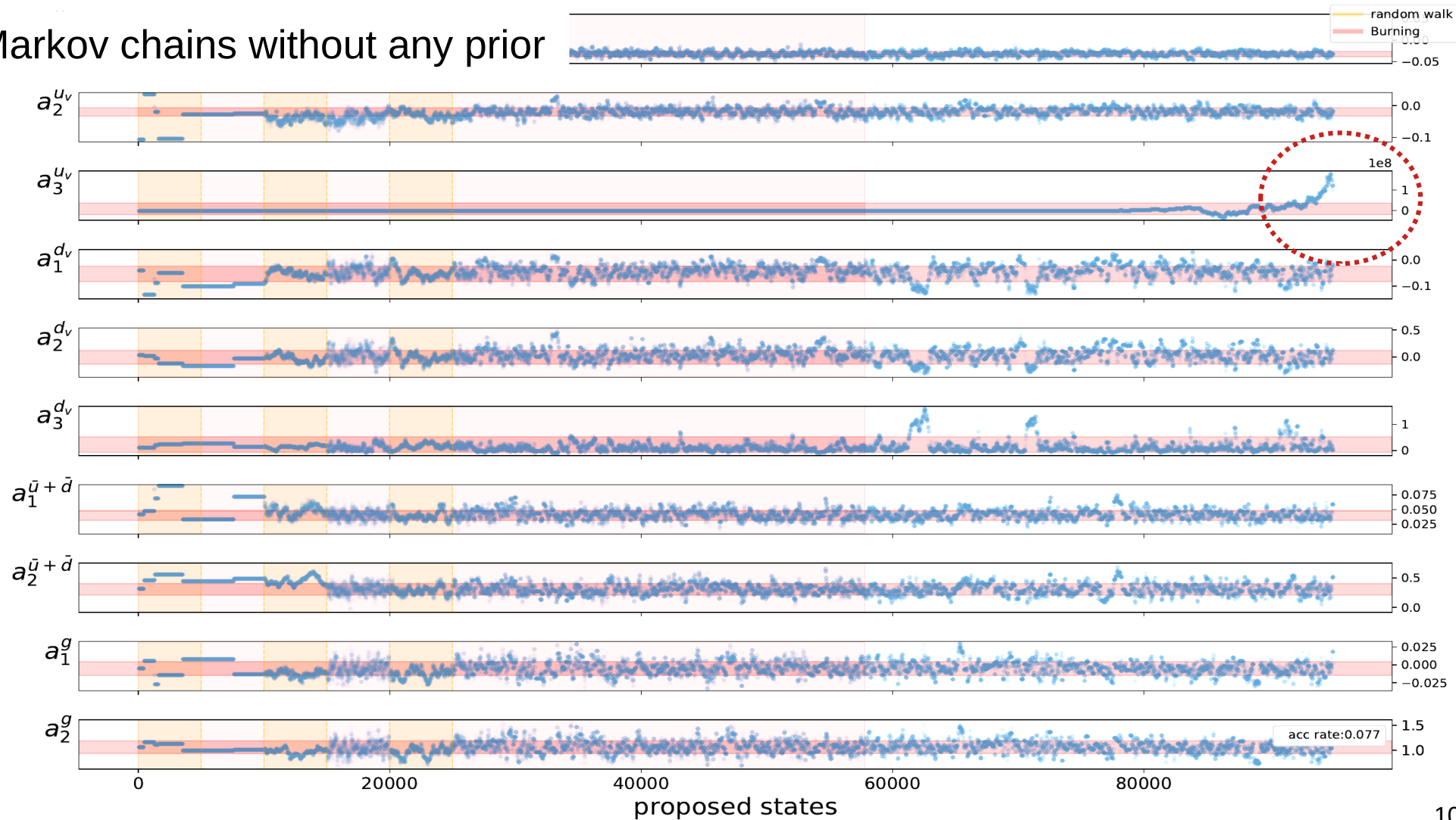
Preliminary results

Markov chain generated for Pb PDF parameters (W/Z and Heavy Quark and ν -DIS(chorus); 1448 data)



Generating this chain took about 20 days on 1 cpu

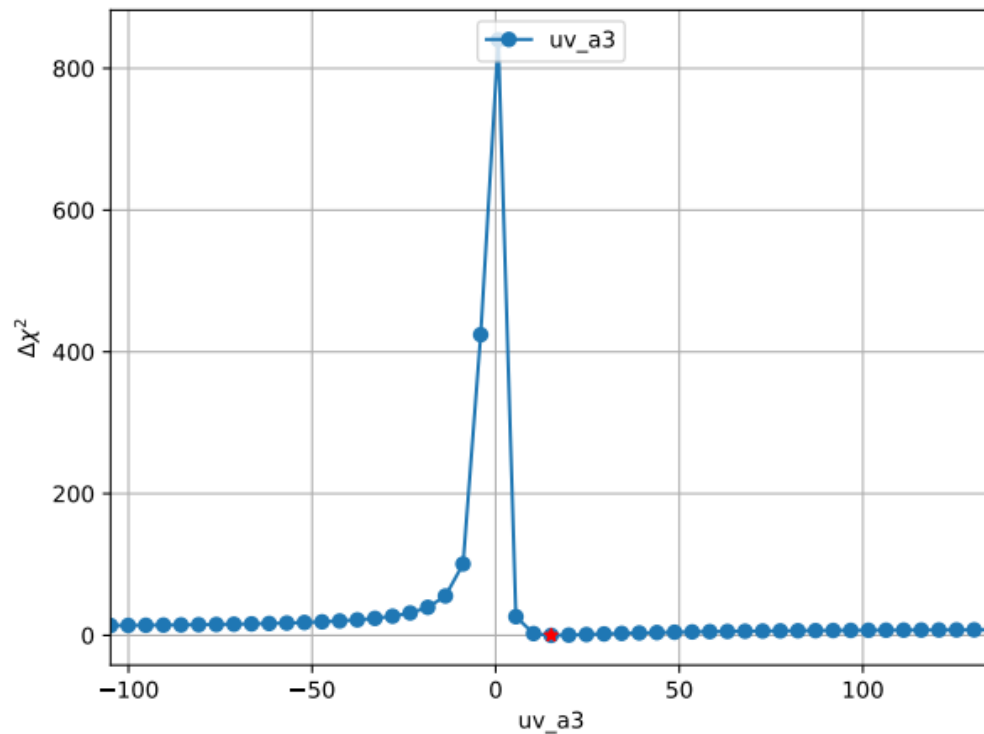
Markov chains without any prior



Prior setup

Prior: we just use a uniform prior for the parameter: $a_3^{uv} : U(-300, 300)$

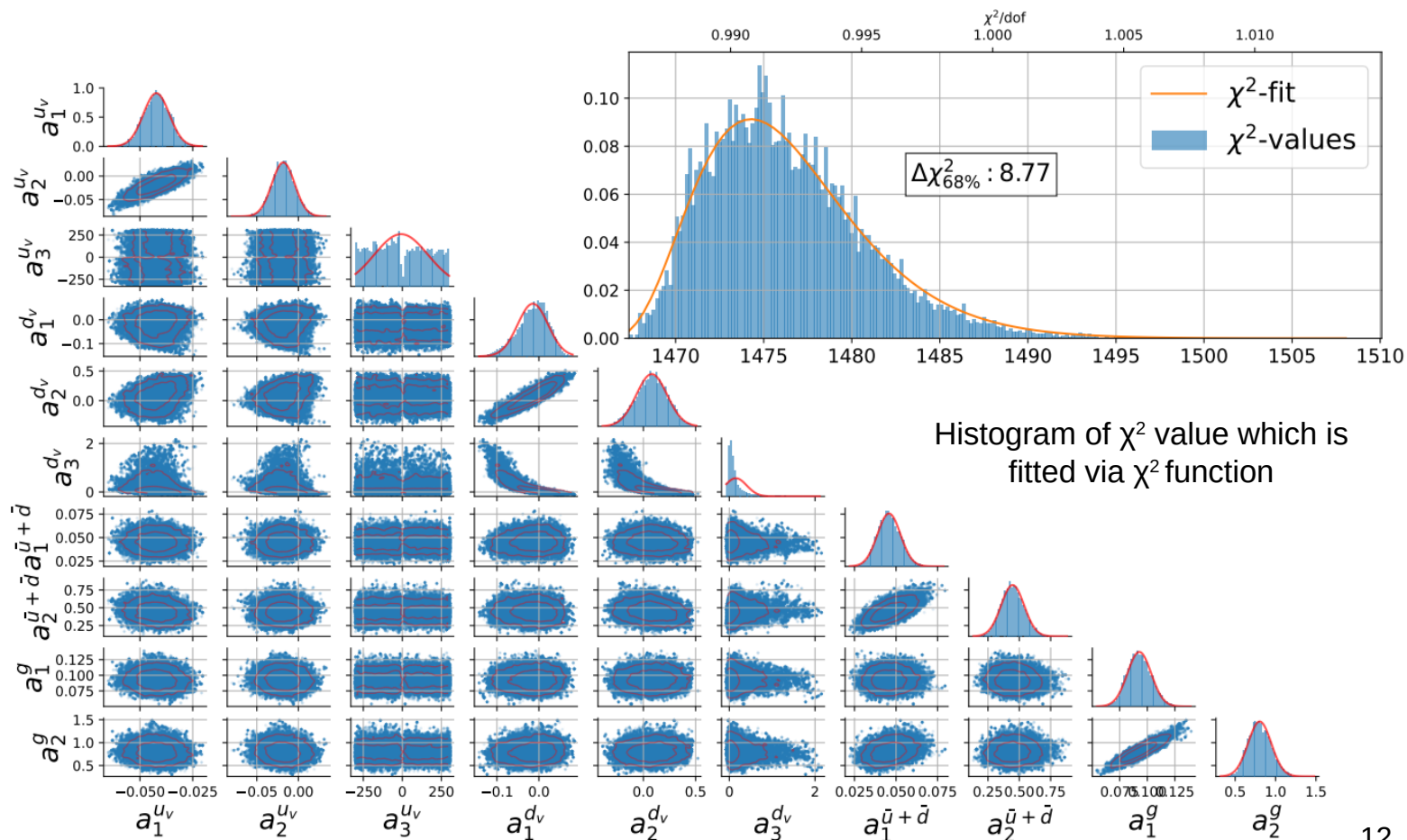
Scan of the χ^2 function along the nPDF parameters
(varying always one free parameter at a time while other parameters were left fixed at the global minimum)



Preliminary results

Pairwise plot:

diagonal: histogram of each parameter
off-diagonal: 2D correlation plots between parameters



MCMC error estimation

Autocorrelation function (ACF) $\rho(k) = \frac{\text{Cov}(k)}{\text{Cov}(0)}$ $\text{Cov}(k) = \frac{1}{n} \sum_{t=1}^{n-k} (x_{t+k} - \bar{x})(x_t - \bar{x}),$

Integrated autocorrelation time $\tau_{int} = \frac{1}{2} \sum_{-\infty}^{+\infty} \rho(k)$ \longrightarrow **Gamma-method**
Estimating by analyzing the sum of autocorrelation up to a certain lag W_{opt}

**Monte Carlo error estimation
(uncorrelated)**

$$\sigma_{MC}^2 = \frac{1}{n-1} \sum_{t=1}^n (X_t - \hat{\mu})^2$$

**MCMC error estimation
(correlated)**

$$\sigma_{MCMC}^2 = 2 \tau_{int} \sigma_{MC}^2$$

MCMC error estimation

Thinning method: keep only every k-th sample in the Markov chain and discard the rest

Why Thinning?

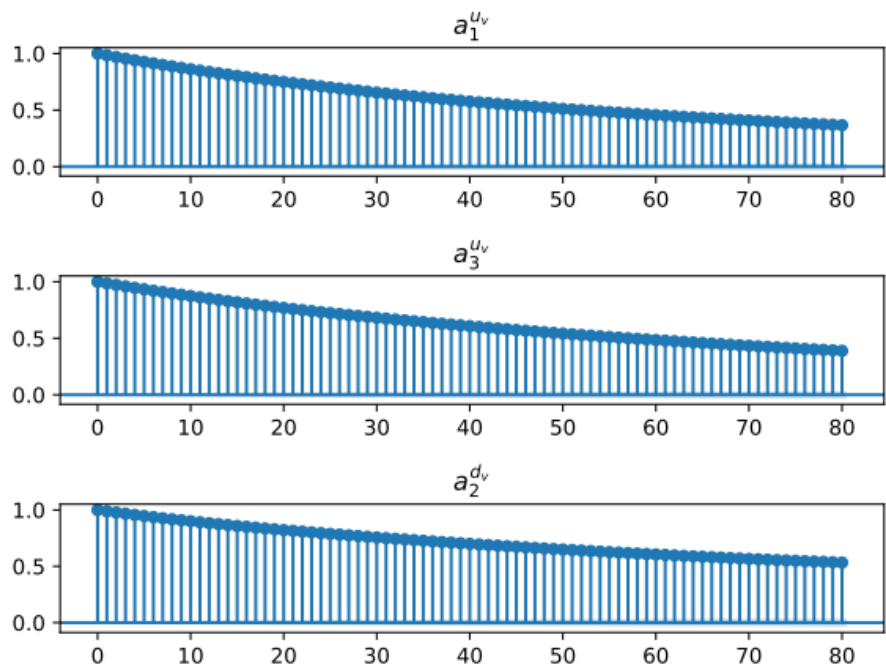
- It provides an **uncorrelated** chain so we can use Monte-Carlo error estimation:

$$\sigma_{MCMC}^2 = 2\tau_{int} \sigma_{MC}^2 \quad \longrightarrow \quad \sigma_{MC}^2 = \frac{1}{n-1} \sum_{t=1}^n (X_t - \hat{\mu})^2$$

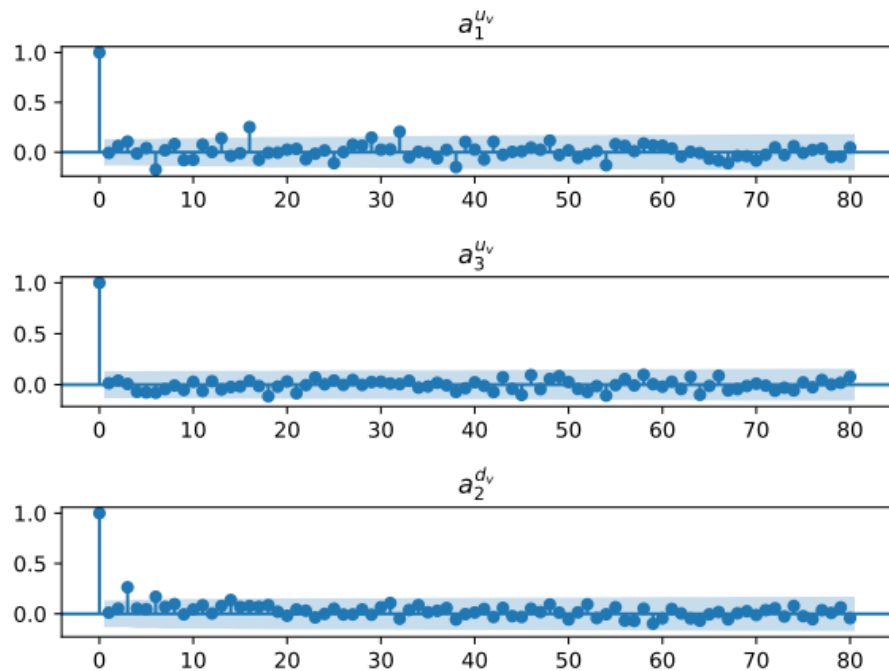
- We aim to generate a set of PDF grids corresponding chain's units. Thinning the chain makes it more applicable.

MCMC error estimation

Autocorrelation function versus time interval:



Thinning
by rate 100



Methodology

◆ **Generating Multiple Chains**

Each chain starts with random values from the Hessian fit results. Use different random seeds

◆ **Removing Burn-In Phase**

Discard the initial segment of each chain, known as the burn-in or thermalization phase, which represents the period before the chain converges to the target distribution

◆ **Thinning Each Chain**

Apply thinning to each chain to reduce the autocorrelation, aiming to retain only uncorrelated samples

◆ **Combining Uncorrelated Samples**

Merge all the thinned, uncorrelated samples from the different chains into a single chain

◆ **Estimating Parameters and Uncertainties**

Use the combined set of uncorrelated samples to estimate the values of nPDF parameters and their uncertainties.

◆ **generating an LHAPDF set**

Construct nPDF corresponding to each unit of the combined chain and perform error estimation in the level of nPDF (Saving them in the standard LHAPDF format so that anyone can use such nPDFs)

Uncertainty estimation

Percentile method [nNNPDF]

After generating nPDF corresponding to each point of thinned Markov Chain, then we perform “percentile method” to estimate the nPDF uncertainty:

The percentile method involves:

- generating a distribution of a statistic (in our case: **distribution of nPDFs**)
→ then determining confidence intervals by identifying the appropriate percentiles of this distribution.

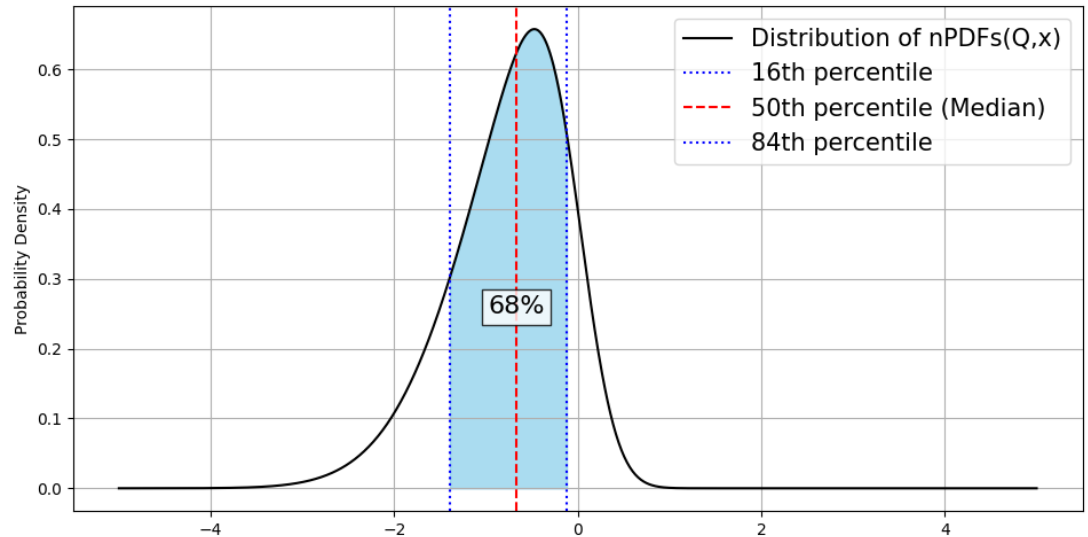
68% CI :

$$\alpha = 1 - 0.68 = 0.32$$

$$\text{Lower Confidence Limit: } P_{\alpha/2} = P_{16\text{th}}$$

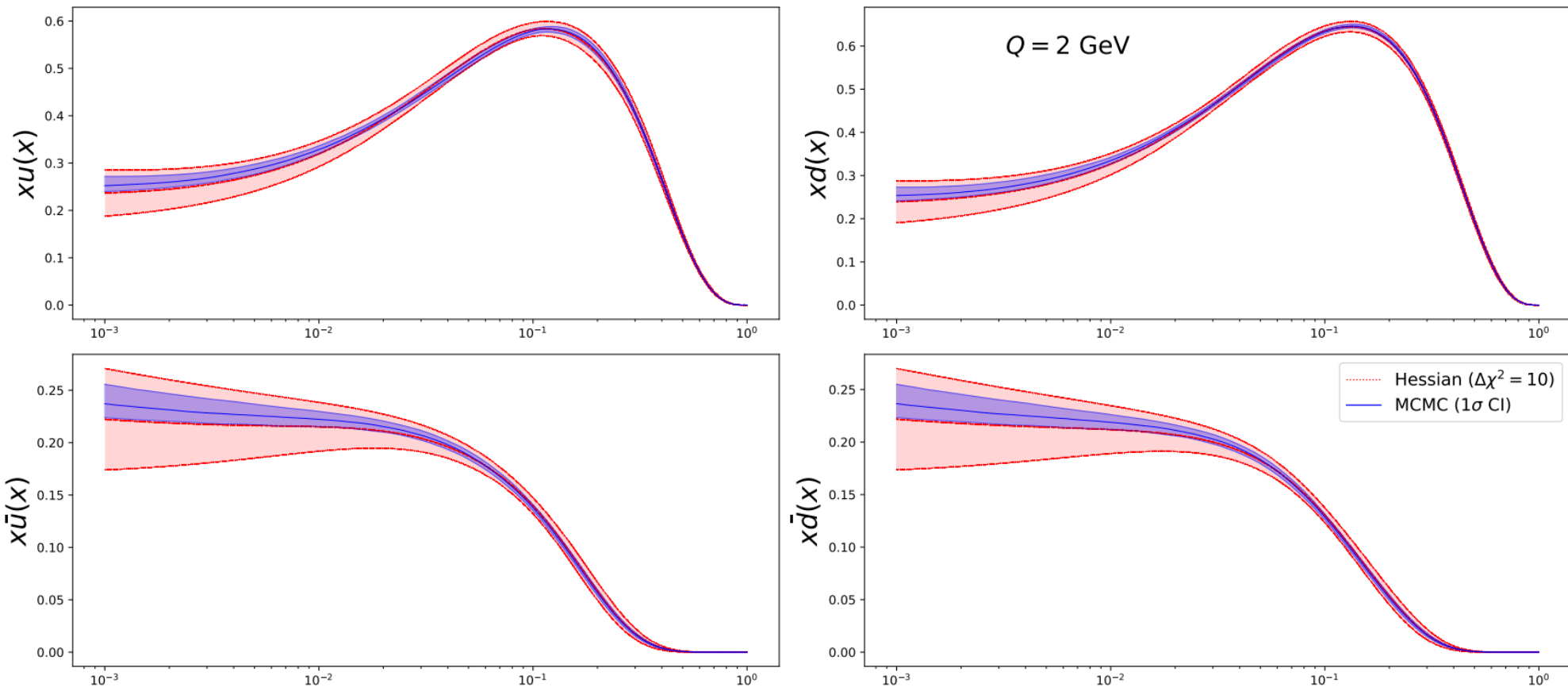
$$\text{Upper Confidence Limit: } P_{(1-\alpha/2)} = P_{84\text{th}}$$

$$\text{Central value: } P_{50\text{th}}$$



LHAPDF grids

Pb²⁰⁸ PDF resulting from **MCMC**(percentile method to for uncertainty estimation) and **Hessian** methods



Conclusion

Conclusion:

- Despite the MCMC challenges (mainly computational cost), this method has become a powerful tool for determining nPDFs and so far we have obtained promising results (comparing with Hessian) for Pb PDF fit
- We would like to extend this approach for multiple nuclei PDF fits and investigate additional statistical methods for estimating Markov Chains uncertainty.

Acknowledgment:

This work was supported by Narodowe Centrum Nauki under grant no.\ 2019/34/E/ST2/00186.

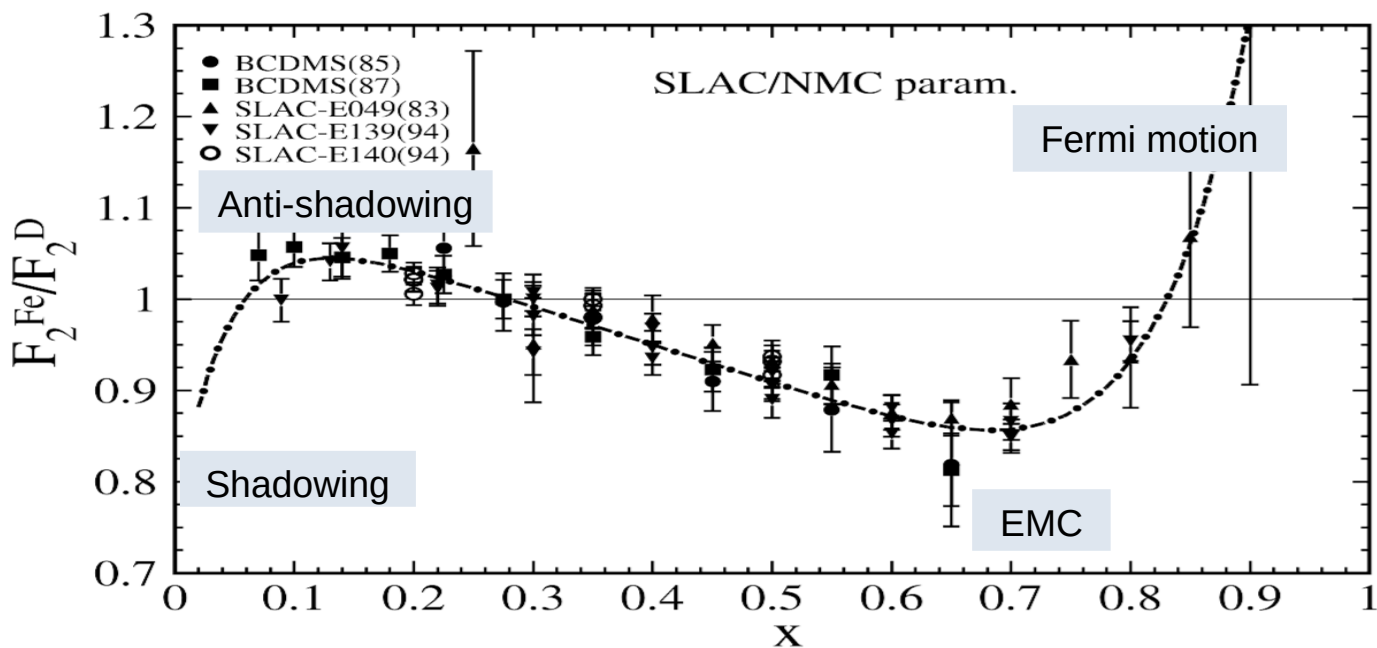
Back up

Nuclear Parton Distribution Function (nPDF)

nPDF describes the momentum distribution of partons (quarks and gluons) inside a nucleus

$$F_2^A(x) \neq ZF_2^p(x) + NF_2^n(x)$$

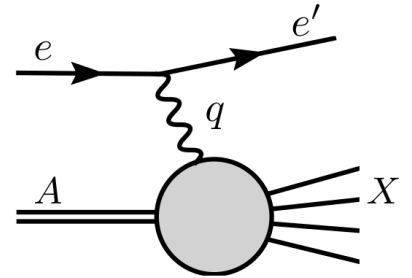
Nuclear correction ratio:



Theoretical Framework (nPDF)

DIS variables for **nucleus** $\left\{ \begin{array}{l} q \equiv k' - k, \quad Q^2 \equiv -q^2 \quad x_A \equiv \frac{Q^2}{2p_A \cdot q} \\ p_A : \text{nucleus momentum} \\ x_A \in (0, 1) : \text{fraction of the nucleus momentum} \\ \text{carried by a nucleon} \end{array} \right.$

DIS variables for **parton** $\left\{ \begin{array}{l} x_N = Ax_A : \text{parton momentum fraction with respect to the average} \\ \text{nucleon momentum } p_N \\ p_N = \frac{p_A}{A} \\ x_N \in (0, A) \end{array} \right.$



$$e(k) + A(p_A) \rightarrow e'(k') + X$$

Theoretical Framework (nPDF)

Nuclear modifications can be incorporated into the PDF framework:

1. Factorization: We assume that the nuclear effects can be absorbed into the universal nPDFs.

$$\sigma_{pA \rightarrow X} = \underbrace{f^p(x_1, \mu^2)}_{\text{free proton PDF}} \otimes \underbrace{f^A(x_2, \mu^2)}_{\text{nuclear PDF}} \otimes \hat{\sigma}(x_1, x_2, \mu^2)$$

2. Bound proton PDF $f^{p/A}$ satisfies the same **evolution equations** and **sum rules** as free proton PDF.

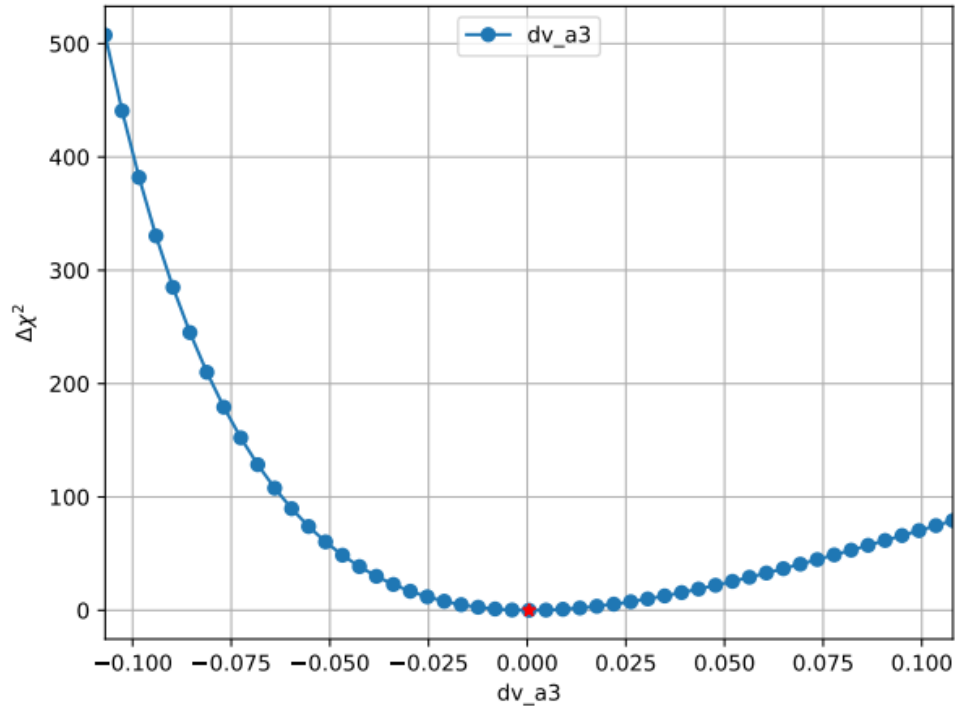
3. Isospin symmetry: $f_{d,u}^{n/A} = f_{u,d}^{p/A}$

nuclear PDF:

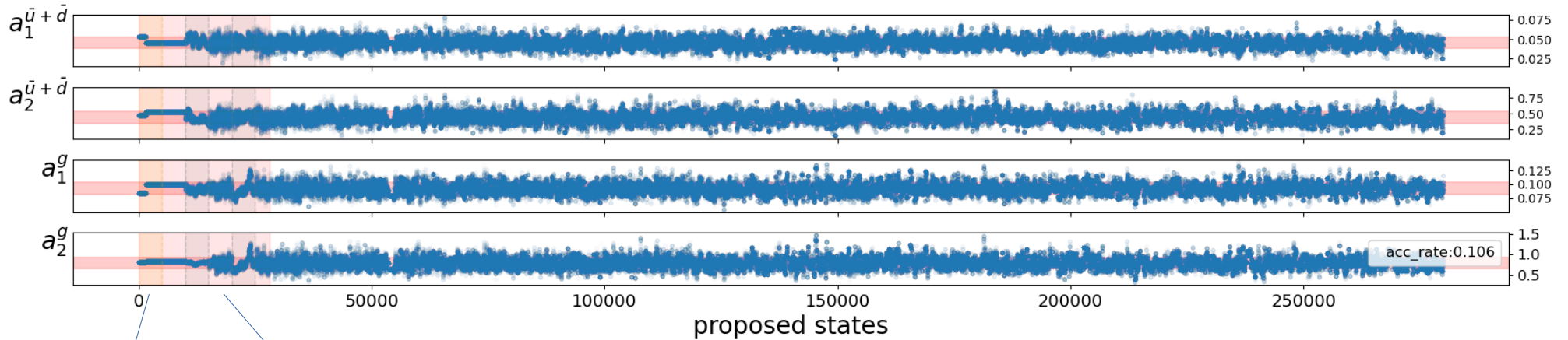
$$f_i^{(A,Z)} = \frac{Z}{A} f_i^{p/A} + \frac{A-Z}{A} f_i^{n/A}$$

MCMC setup

Scan of the χ^2 function along dv-a3 parameter



MCMC setup



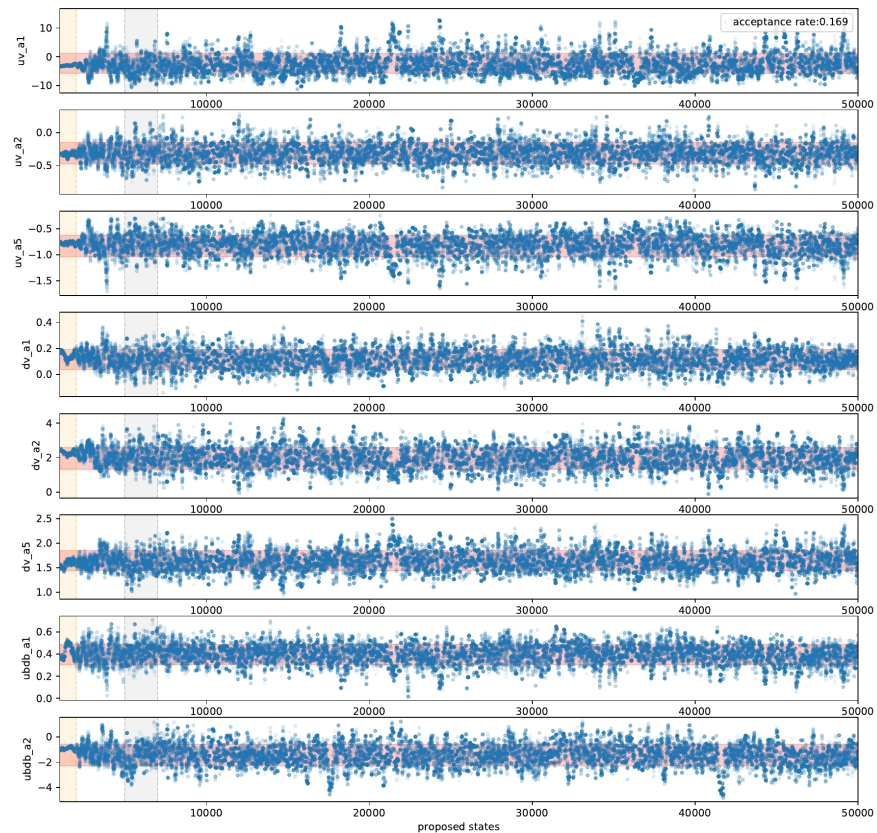
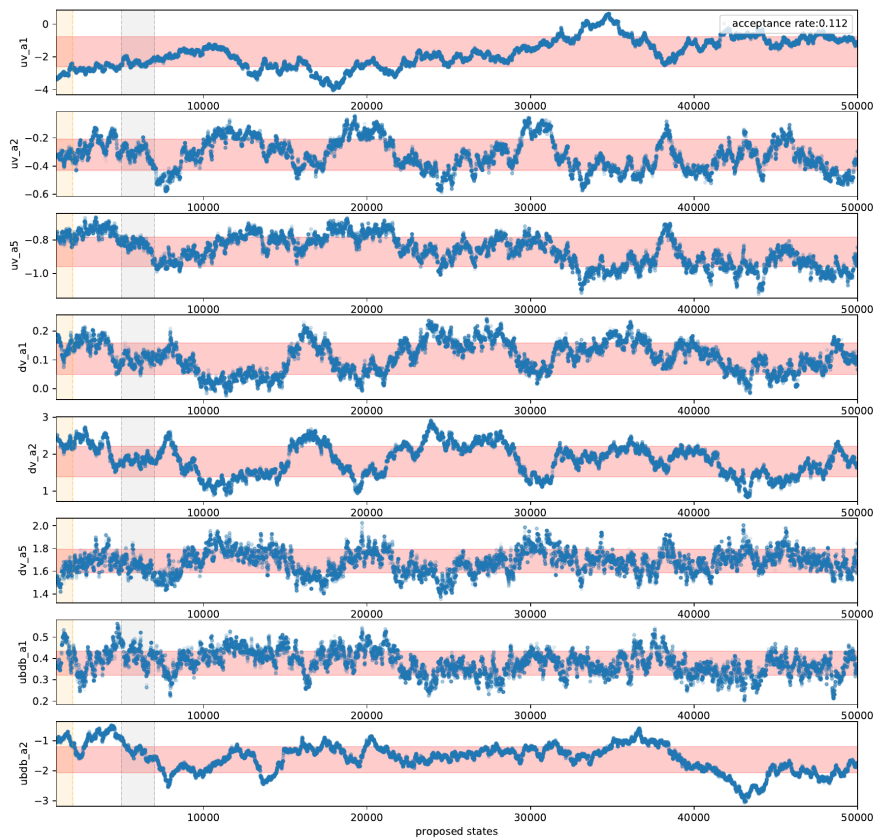
$N_0 = 5000$

Restarting the chain at
10,000 and 20,000

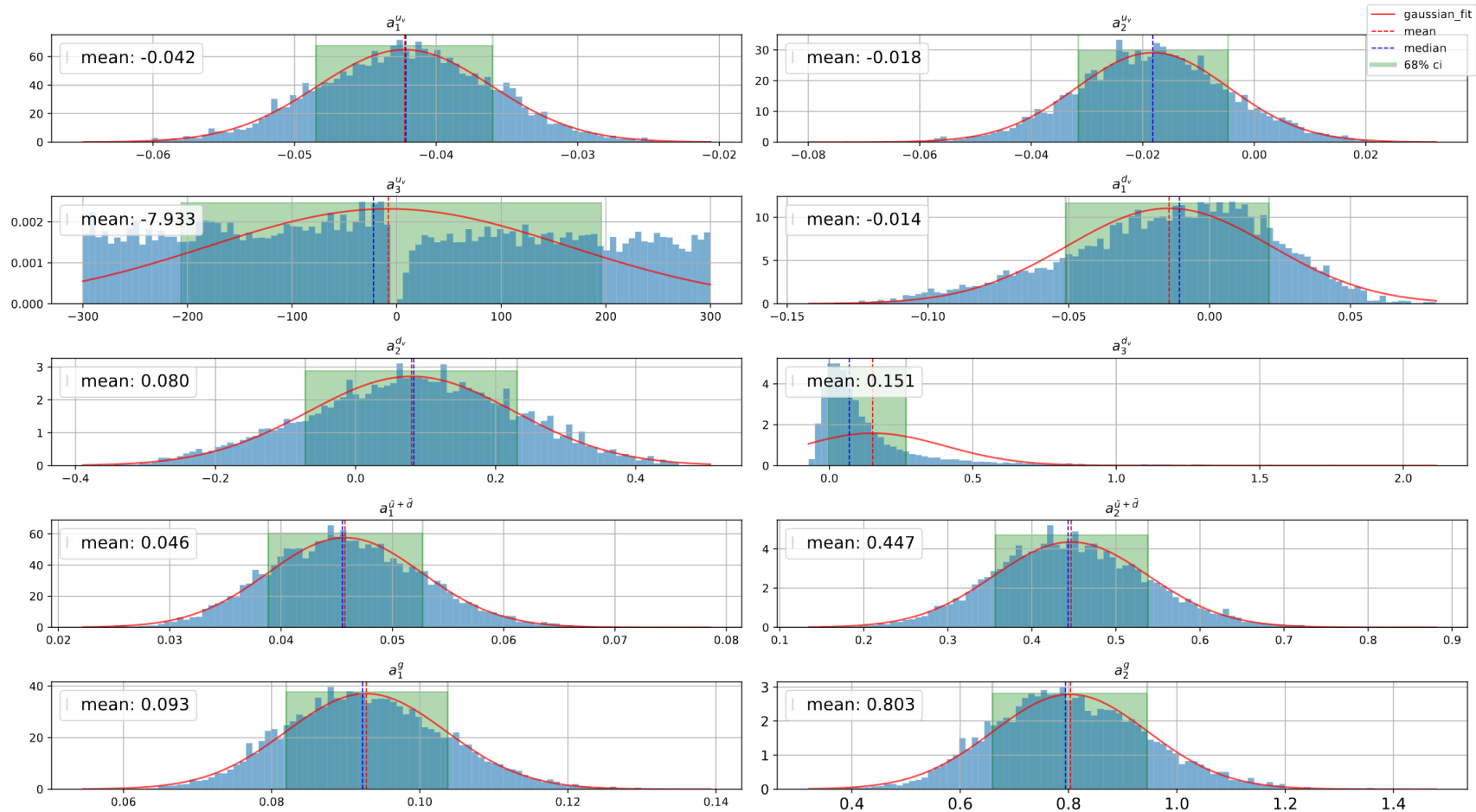
Starting point: global minimum from Hessian fit + Gaussian noise (width= 20 % of minimum value)
Thermalization (burn-in phase): removing first 8000 accepted points

MCMC setup

MH vs adaptive MH

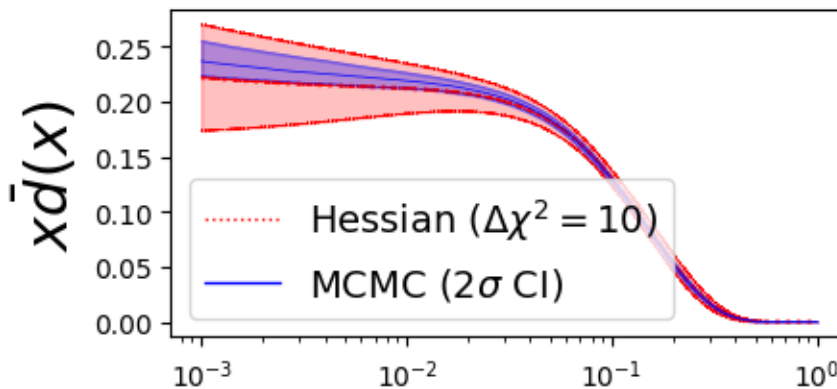
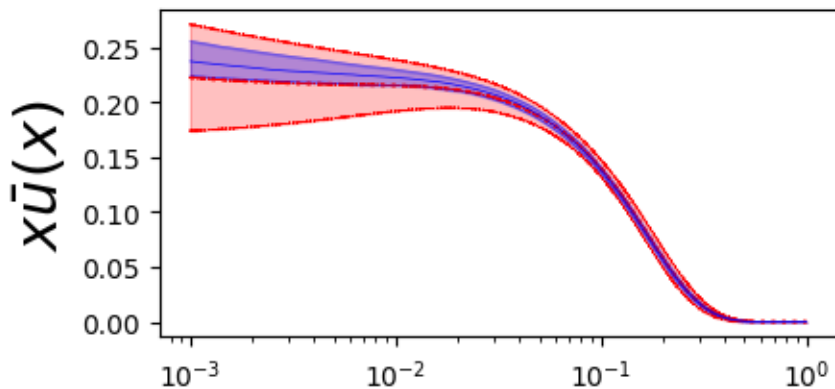
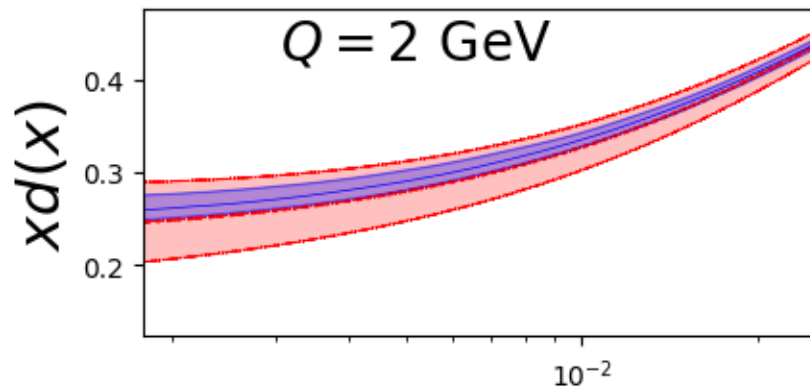
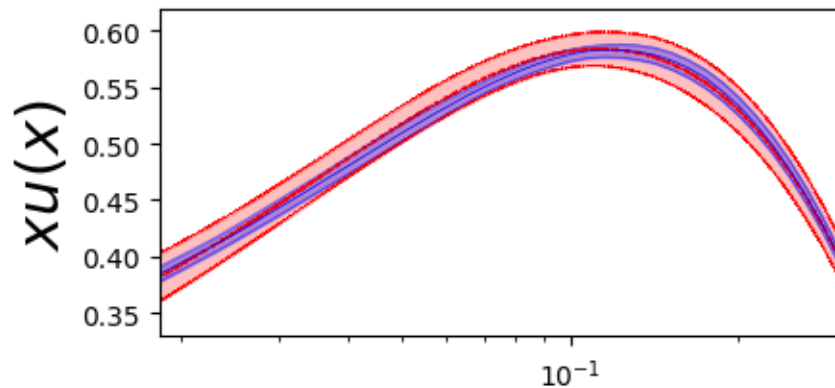


Preliminary results



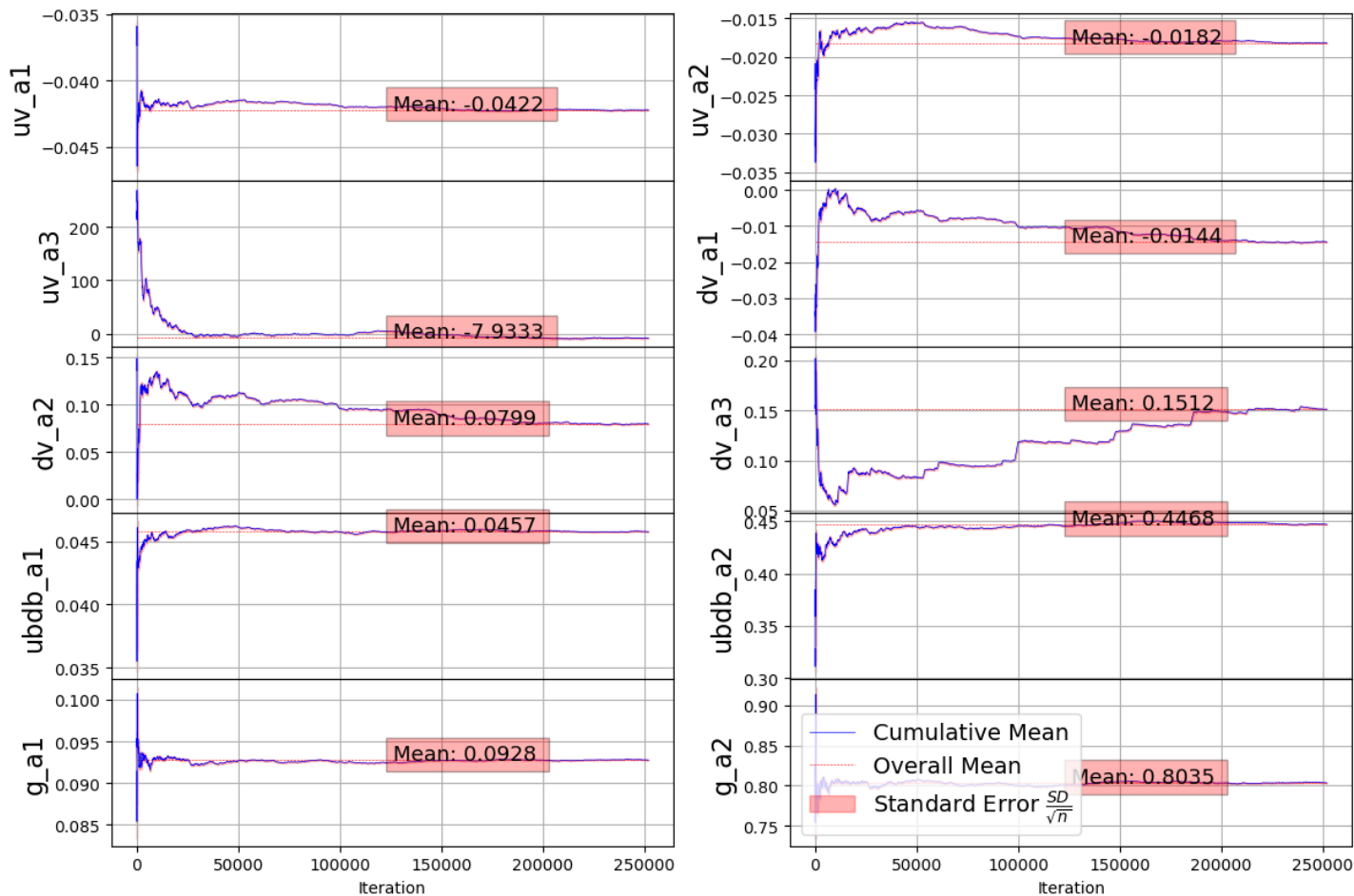
LHAPDF grids

Pb²⁰⁸ PDF resulting from **MCMC**(percentile method to for uncertainty estimation) and **Hessian** methods



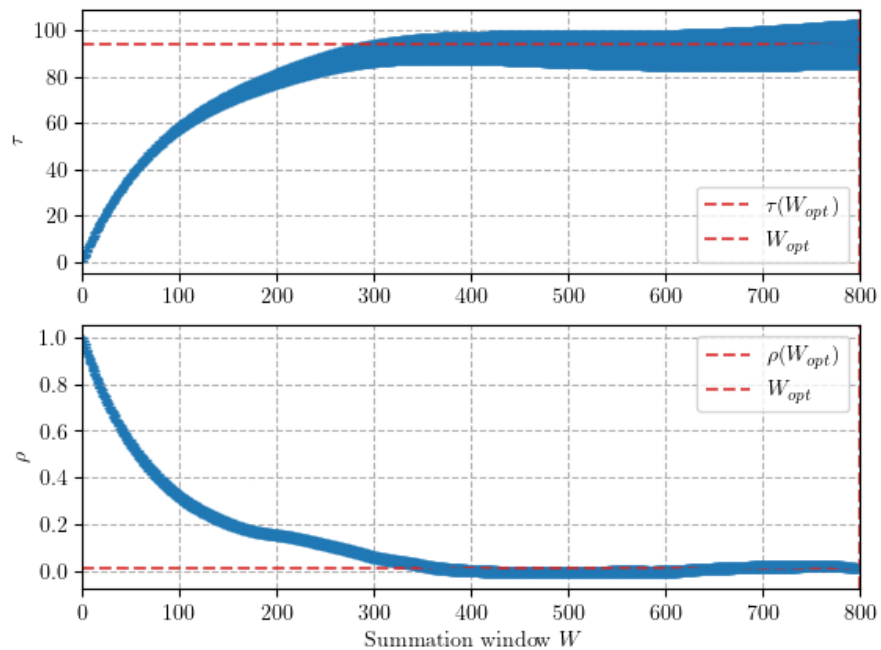
Preliminary results

Cumulative Mean:



MCMC setup

Autocorrelation $\rho(k)$ and autocorrelation time τ_{int} calculating via **Gamma-method**, before and after thinning



Thinning
by rate 100

