Rejection Sampling from Arbitrary Multivariate Distributions Using Generalized Fibonacci Lattices

Daniel Frisch and **Uwe D. Hanebeck**
Intelligent Sensor-Actuator-Systems Laboratory (ISAS)
Institute for Anthropomatics and Robotics
Karlsruhe Institute of Technology (KIT), Germany
daniel.frisch@ieee.org, uwe.hanebeck@ieee.org

***Abstract*—We present a quasi-Monte Carlo acceptance-rejection sampling method for arbitrary multivariate continuous probability density functions. The method employs either a uniform or a Gaussian proposal distribution. The proposal samples are provided by optimal deterministic sampling based on the generalized Fibonacci lattice. By using low-discrepancy samples from generalized Fibonacci lattices, we achieve a more locally homogeneous sample distribution than random sampling methods for arbitrary continuous densities such as the Metropolis-Hastings algorithm or slice sampling, or acceptance-rejection based on state-of-the-art quasi-random sampling methods like the Sobol or Halton sequence.**

I. Introduction

A. Context

Random, quasi-random, and deterministic samples, or particles, play an important role in, e.g., nonlinear filtering and control. In contrast to continuous density functions, samples can very easily be propagated through nonlinear functions like system dynamics or measurement models. Furthermore, expectations such as covariance matrices can be efficiently estimated using samples. Quasi-random numbers, instead of random numbers, are often used to cover the state space more uniformly and to avoid clusters of samples, and gaps in other places, that occur frequently in random samples, see Fig. 1.

B. Problem to be Considered

We consider the problem of obtaining unweighted samples of arbitrary multivariate continuous density functions. In order to obtain a point distribution that is locally homogeneous, and therefore exhibits faster convergence of expectation estimates, we employ low-discrepancy samples, preferably with the generalized Fibonacci lattice.

C. State-of-the-art

There are different ways to represent density functions with samples. The simplest way is a regular, equidistant grid, where the samples are weighted proportionally to the density value at the respective location [1]. However, since the entire state space must be covered with this grid, storage requirements are high. Furthermore, samples with small weights have only little contribution to the result but contribute equally to the computational load. Alternatively, there are non-equidistant unweighted samples that store information about the density in the sample locations instead of their weights. Three general variants exist. See Fig. 1 for a visual comparison between them.

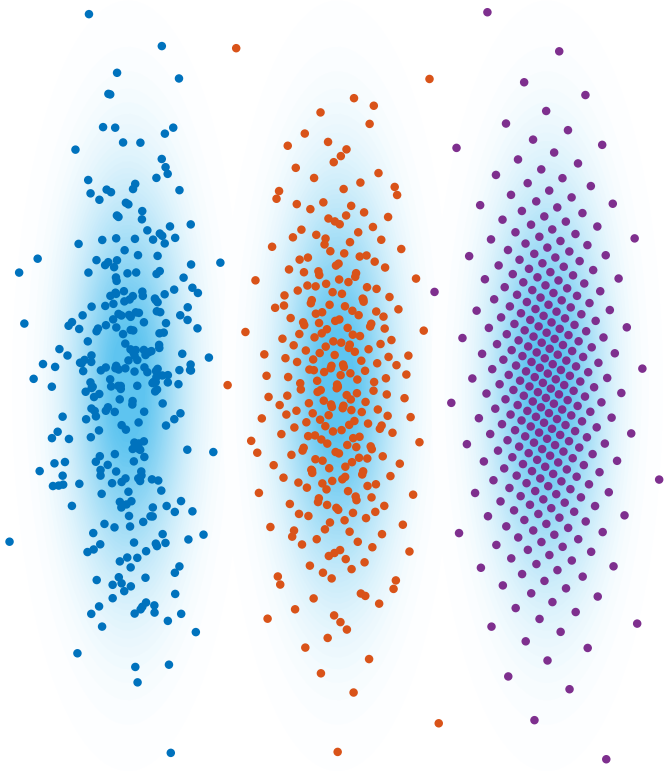


Fig. 1: Random (blue), quasi-random (red), and optimal deterministic (purple) Gaussian samples. Quasi-random: inversion method on uniform Sobol sequence. Optimal deterministic: inversion method on uniform Fibonacci grid. The figure shows 300 samples for each method.

Random: First and foremost, there are independent random samples. There are direct methods and fast inversion methods in place to generate random (or pseudo-random) samples efficiently for various density functions, like the uniform and standard normal random number generators being present in every statistical programming library. For arbitrary densities, the random acceptance-rejection method can be used [2], [3]. Furthermore, there are Markov chain-based

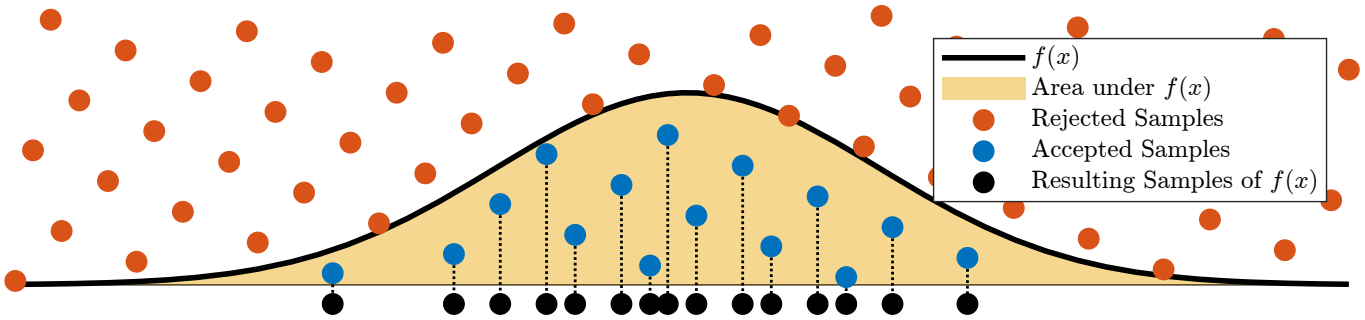


Fig. 2: Simple and intuitive example for acceptance-rejection sampling from a univariate Gaussian density (thick black line). Uniform proposal samples (union of red and blue dots) with added "extra dimension" (vertical axis). Samples "above" $f(x)$ are rejected (red dots), samples "below" are accepted (blue dots). Finally, the y-coordinate, corresponding to the "extra dimension", is discarded, and we obtain the wanted univariate samples of $f(x)$ (black dots).

samplers like the Metropolis-Hastings algorithm [4] and the slice sampler [5].

Quasi-Random $\equiv$ Suboptimal Deterministic: Second, there are quasi-random deterministic sequences that exhibit a low discrepancy. That is, they cover the state space more evenly by avoiding random clusters, and therefore allow for an improved convergence rate, i.e., better estimation results with fewer samples. Examples are the Halton and Sobol sequence [6]. These sequences generate multivariate uniform samples and can often be used as a drop-in replacement for random samples. With inversion techniques, some non-uniform densities such as the Gaussian can be obtained just as with random samples. Acceptance-rejection, on the other hand, cannot be transferred one-to-one from random samples to quasi-random ones. To retain the low discrepancy, the samples may be smoothed [7]. A more elegant way is to produce quasi-random proposal samples with one additional dimension, and have a deterministic acceptance-rejection based on the value of that additional coordinate [8], [9]. This is exactly what we propose, however, for proposal sampling, we employ the generalized Fibonacci sequence.

Optimal Deterministic: Third, there is optimal deterministic sampling. These methods place each individual sample optimally. Usually, they are the solution to an optimization problem and thus are expensive to compute. This holds for the Localized Cumulative Distribution (LCD), where a distance measure between continuous and discrete density functions is defined and minimized numerically [10], [11], [12]. The projected cumulative distribution (PCD) can be applied to more types of density functions and is somewhat easier to calculate, as sample placement involves a gradient-free iterative optimization similar to expectation-maximization [13], [14]. Recently, Purser's generalized Fibonacci grid [15] has made it possible for the first time to have sample sets with properties analogous to the well-known two-dimensional Fibonacci grid [16], [17] also in higher dimensions. It has been used to directly obtain optimal deterministic Gaussian samples [18] in dimensions N where $(2N+1)$ is prime and also for $N=4$.

Sampling techniques can also be classified into "open" methods, where samples can be added without changing the previously drawn samples, and "closed" methods, where all samples have to be discarded in order to add or remove samples. Furthermore, some methods produce samples that can be transformed, e.g., from standard normal to arbitrary Gaussian, while keeping their homogeneity properties.

II. Key Idea

Instead of state-of-the-art quasi-random proposal samples for acceptance-rejection [3], we propose to employ optimal deterministic proposal samples. More specifically, we use Purser's generalized Fibonacci grid. These point sets generalize the properties of the two-dimensional Fibonacci grid to higher dimensions. Note that constructions for the generalized Fibonacci grid are yet available for a limited number of dimensions only [18].

III. Inputs and Outputs

Here, we give a brief overview of what our method requires as input (I1, I2, I3, I4) and returns as output, depending on the type of proposal.

A. Uniform Proposal Density

I1 density function $f(\underline{x})$ as function handle
I2 bounding box of probability mass of $f(\underline{x})$
I3 scaling factor c such that $c \cdot f(\underline{x}) \leq 1 \; \forall \underline{x}$
I4 wanted number of samples

B. Gaussian Proposal Density

I1 density function $f(\underline{x})$ as function handle
I2 mean μ_p and covariance $\mathbf{C}_\mathrm{p}$ of proposal $f_\mathrm{p}(\underline{x})$
I3 scaling factor c such that $c \cdot f(\underline{x})/f_\mathrm{p}(\underline{x}) \leq 1 \; \forall \underline{x}$
I4 wanted number of samples

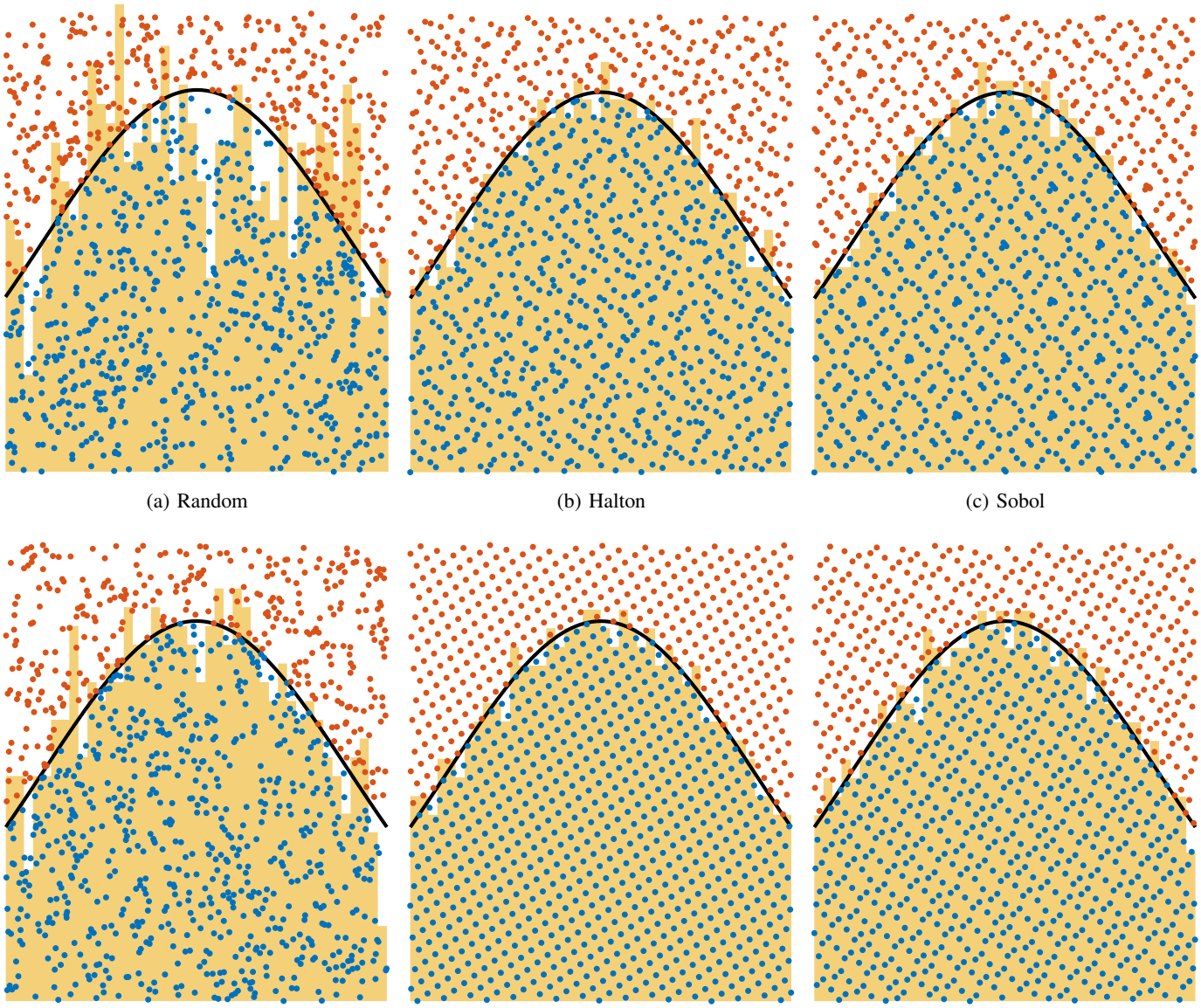
Based on this information, our proposed method calculates L unweighted quasi-random samples that approximate $f(\underline{x})$.

IV. Method Derivation

A. Prerequisites

We assume an arbitrary known probability density function (pdf) $f(\underline{x})$ with $\underline{x} \in \mathbb{R}^N$ to be given. The argument vector $\underline{x}$ can be written as

$$\underline{x} = \begin{bmatrix} x^{(1)} & x^{(2)} & \dots & x^{(N)} \end{bmatrix}^\top = \underline{x}^{(1:N)} \; . \tag{1}$$



(a) Random (b) Halton (c) Sobol

(d) Latin Hypercube (e) Fibonacci 2D (f) Fibonacci Ext. 3D

Fig. 3: Visualization of acceptance-rejection sampling from a given univariate density $f(x)$ (in black). The vertical axis represents the "extra dimension" used for the acceptance-rejection decision. Uniform samples are shown, where the red ones are rejected and the blue ones accepted. A histogram of the blue (accepted) samples with 43 bins is shown in yellow. Random samples give the worst result, closely followed by Latin hypercube sampling. Low-discrepancy sequences (Halton, Sobol) provide more uniform results but are outperformed by three-dimensional Fibonacci sampling ($N+2$, "open") and even more by two-dimensional ($N+1$, "closed") Fibonacci sampling. (Figure modified from [9].)

The pdf $f(\underline{x})$ is not necessarily normalized, i.e., the condition

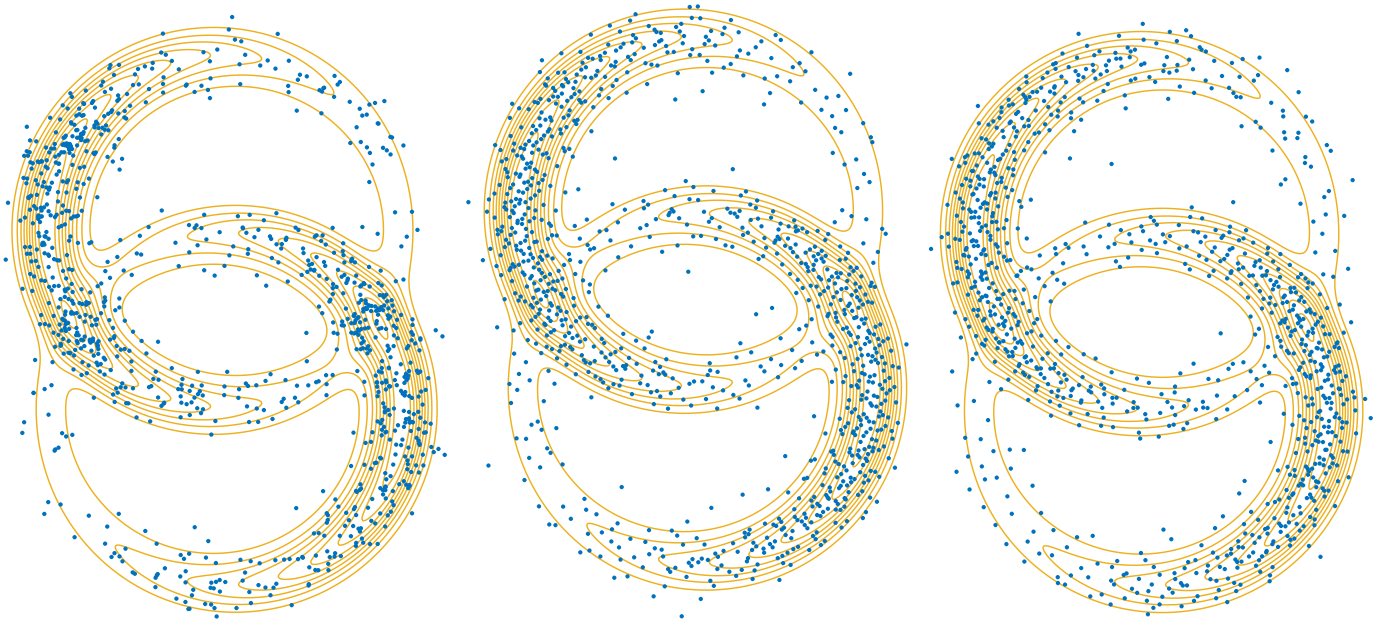
$$\int_{\mathbb{R}^N} f(\underline{x}) \, \mathrm{d}\underline{x} = 1 \tag{2}$$

does not necessarily hold. A typical reason for that is that the normalization integral cannot be calculated with a reasonable computational cost.

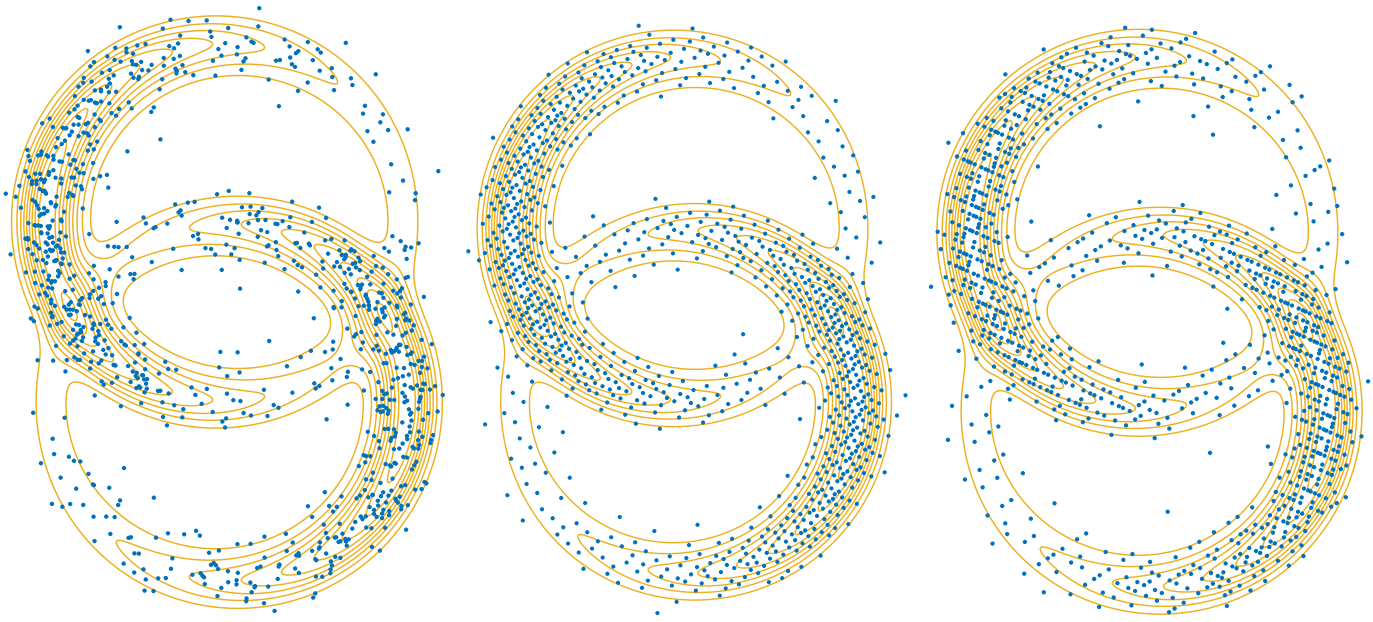
For this pdf $f(\underline{x})$, we want to find a Dirac mixture (DM) approximation of the form

$$\hat{f}(\underline{x}) = \frac{1}{L} \sum_{i=1}^{L} \delta(\underline{x} - \hat{\underline{x}}_i) \ , \tag{3}$$

i.e., a set of equally weighted (what we also call unweighted) samples $\hat{\underline{x}}_i$ that represent the continuous pdf $f(\underline{x})$ in an optimal way.



(a) Slice (b) Halton (c) Sobol



(d) Latin Hypercube (e) Fibonacci 3D (f) Fibonacci Ext. 4D

Fig. 4: Contour lines (yellow) of the non-Gaussian density function $f(\underline{x})$ used here for evaluation, the exact formula is given in (22). Overlaid about 2^{10} samples (blue) produced by slice sampling (a) and acceptance-rejection of various uniform sample sets (b-f). Note that the individual numbers of samples vary slightly due to rejection. For state-of-the-art methods (a-d), MATLAB standard parameters for the random slice sampler (a) and the quasi-random number generators (b-d) are used, i.e., no skip or leap. Compare our proposed methods (e, f) with a uniform generalized Fibonacci proposal: "closed" version, projected from a three-dimensional Fibonacci grid (e), and "open" version, projected from a four-dimensional Fibonacci grid (f). Quantitative results in Fig. 5.

B. Uniform Proposal Sampling

In the first step, we generate a uniform proposal point set $\underline{\hat{u}}_j$ with one "extra dimension"

$$\underline{\hat{u}}_j \in [0,1]^{N+1} \subset \mathbb{R}^{N+1} \ , \tag{4}$$

$$\underline{\hat{u}}_j = \begin{bmatrix} \underline{\hat{u}}_j^{(1:N)} \\ \hat{u}_j^{(N+1)} \end{bmatrix} \ , \qquad j \in \{1,2,\ldots,L_0\} \ , \tag{5}$$

where L_0 is chosen properly such that the we obtain approximately the desired number of samples after rejection. We propose to use the generalized Fibonacci grid [15], [18] here as an optimal deterministic proposal.

Second, we define a bounded region that contains a significant amount of the probability mass of $f(\underline{x})$. In this paper, we focus on a hyperrectangular bounding box $\mathcal{B} \subset \mathbb{R}^N$ for simplicity. We describe this bounding box by a matrix $\mathbf{B} \in \mathbb{R}^{N\times 2}$ that contains minimum and maximum values for each dimension. A practical criterion for defining such a bounding box in this context is

$$c \cdot f(\underline{x}) \leq \gamma \quad \forall\, \underline{x} \notin \mathcal{B} \ , \tag{6}$$

$$\mathcal{B} : \{x \mid x \in [B_{1,1}, B_{1,2}] \times \ldots \times [B_{N,1}, B_{N,2}]\} \ , \tag{7}$$

where $\gamma = \min_j \left\{ \hat{u}_j^{(N+1)} \right\}$, which is on average $1/L$ for uniform distributions, c is a constant such that $c \cdot f(\underline{x}) \leq 1 \ \forall \underline{x}$, and $B_{n,m}$ are the entries of $\mathbf{B}$ that describe $\mathcal{B}$. The proposal point set is then, via scaling and translation of the individual coordinates, transformed from $[0,1]^{N+1}$ to $\mathcal{B} \times [0,1]$.

C. Gaussian Proposal Sampling

For the Gaussian proposal $f_{\mathrm{p}}(\underline{x})$, we generate an optimal deterministic uniform Fibonacci grid in $[0,1]^{N+1}$ just as above. Subsequently, the first N dimensions are transformed to the Gaussian proposal distribution using the eigenvalues and eigenvectors of its covariance, as described in [18]. The coordinate of dimension $(N+1)$ stays uniformly distributed.

D. Rejection

The rejection decision can be best understood intuitively for the uniform proposal, see Fig. 2. As the height of the pdf is proportional to the desired local point density, and the area under pdf corresponds to the probability mass, we obtain the correct distribution of samples if we uniformly fill the area under the pdf with samples, and then project the samples down by marginalizing out the "extra dimension". For a more detailed explanation, refer to [3].

Regarding the formal description of the rejection process, at this point, we have a proposal sample set in $\mathbb{R}^{N+1}$ given as sample locations $\underline{\hat{u}}_j$, where $\underline{\hat{u}}_j^{(1:N)}$ is either uniform or normal, and $\hat{u}_j^{(N+1)}$ is uniform. In the next step, some samples must be rejected to obtain a representation of the wanted pdf $f(\underline{x})$.

We define a function $h(\underline{x})$ that serves as a reference for whether to accept or reject any sample

$$h(\underline{x}) = c \cdot f(\underline{x}) \tag{8}$$

for uniform and

$$h(\underline{x}) = c \cdot f(\underline{x}) / f_{\mathrm{p}}(\underline{x}) \tag{9}$$

for Gaussian proposals. Either way, it holds $h(\underline{x}) \in [0,1] \ \forall \underline{x}$. In the rejection phase, we remove those samples where the coordinate $\hat{u}_j^{(N+1)}$ is larger than the function value of $h(\underline{\hat{u}}_j^{(1:N)})$. In other words, the indicator function $I(\cdot)$

$$I(\underline{\hat{u}}) = \begin{cases} 1 \, , & \hat{u}^{(N+1)} \leq h(\underline{\hat{u}}^{(1:N)}) \\ 0 \, , & \hat{u}^{(N+1)} > h(\underline{\hat{u}}^{(1:N)}) \end{cases} \tag{10}$$

decides whether to keep $(I(\cdot) = 1)$ or to reject $(I(\cdot) = 0)$ a sample $\underline{\hat{u}}$. After this decision, the coordinate of dimension $(N+1)$ can be discarded and the coordinate values $\underline{\hat{u}}_i^{(1:N)}$ become the final sample locations $\underline{\hat{x}}_i$. See Fig. 3e and Fig. 4e for examples in $N = 1$ and $N = 2$, respectively.

Note that it is not known beforehand how many samples get rejected, and thus how many samples L one obtains for a given number of uniform samples L_0. Therefore, if a "closed" sequence is used, and one wants to obtain exactly a predetermined number of samples L, sampling and rejection must be done repeatedly if necessary. To avoid this, one should resort to "open" sequences in this case, or use the method described in the next section.

E. Sequential Sampling

A small generalization allows for reducing or increasing the number of samples even after rejection, while keeping the old samples in place. To do this, we add yet another "extra dimension" and generate uniform proposal samples

$$\underline{\hat{u}}_j \in \mathcal{B} \times [0,1] \times [\beta_1, \beta_2] \subset \mathbb{R}^{N+2} \ , \tag{11}$$

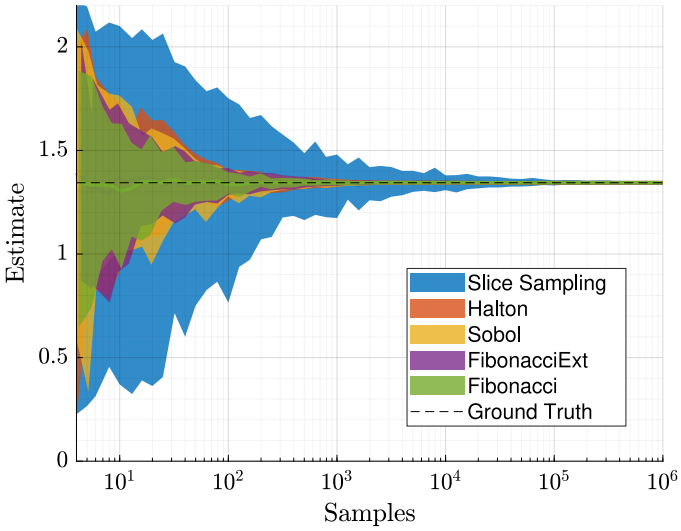
$$\underline{\hat{u}} = \begin{bmatrix} \underline{\hat{u}}^{(1:N)} \\ \hat{u}^{(N+1)} \\ \hat{u}^{(N+2)} \end{bmatrix} \ , \tag{12}$$

with $\beta_1 < \beta_2$, e.g., $[\beta_1, \beta_2] = [0,1]$. Rejection works just the same way as before, based on $\hat{u}_j^{(N+1)}$. See Fig. 3f and Fig. 4f for examples in $N = 1$ and $N = 2$, respectively. To remove samples from the final sample set, they are first sorted according to $\hat{u}_j^{(N+2)}$. Then, the desired number of samples with the largest, or smallest, value of $\hat{u}_j^{(N+2)}$ are removed. For the non-periodic Fibonacci lattice, it is also possible to add samples. One simply needs to take a connected set for $\hat{u}_j^{(N+2)}$ that follows without gap, e.g., $(\beta_2, \beta_3] = (1,2]$. Note that some state-of-the-art low-discrepancy sequences like Halton and Sobol are "open" by design, i.e., have their good-but-not-optimal properties for a wide range of the number of samples and can be used sequentially one by one without the extra dimension.

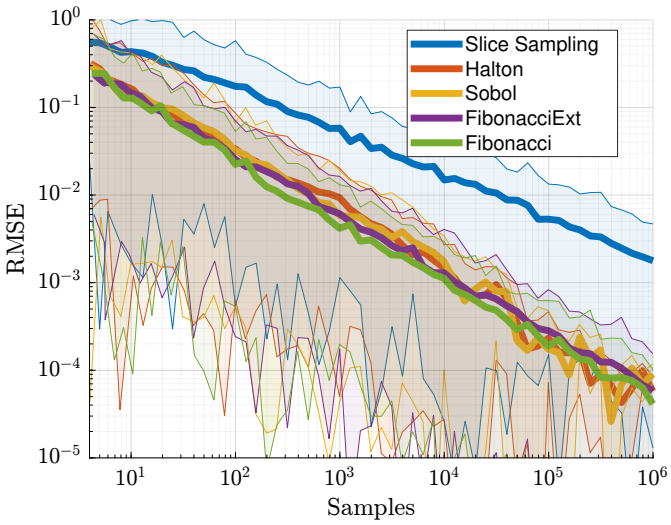
V. Evaluation

We consider calculating the expected value of a nonlinear function $g(\cdot)$ of the random vector $\underline{\boldsymbol{x}}$

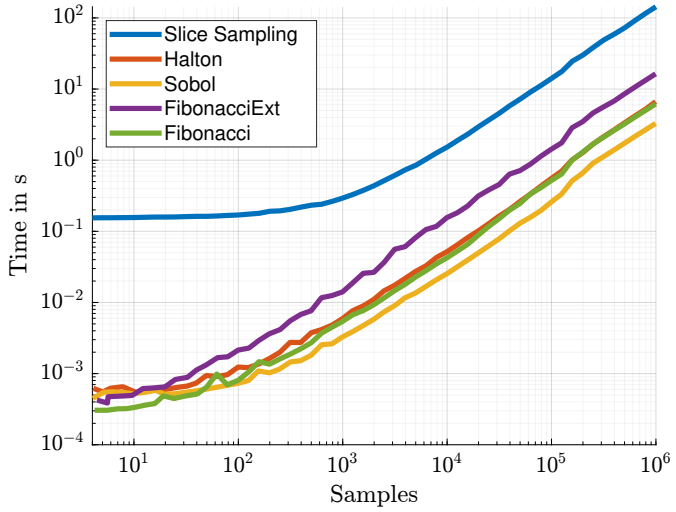
$$\mathrm{E}\{g(\underline{\boldsymbol{x}})\} = \int_{\mathbb{R}^N} g(\underline{x}) \cdot f(\underline{x}) \, \mathrm{d}\underline{x} \ , \tag{13}$$



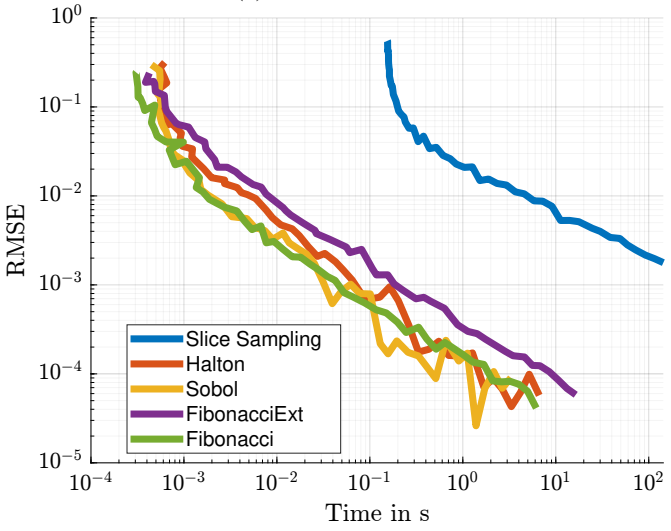
(a) Estimate and ground truth



(b) Estimation error



(c) Computation time



(d) Pareto front

Fig. 5: Quantitative nonlinear expectation estimation results and computation times with various sampling methods. In (a), the minimum and maximum value, and in (b), the best and worst result out of 100 trials is indicated by the shaded area. Solid lines show the mean of all trials. To obtain different results in individual trials also for the quasi-random sequences, parameters like "skip" or "burn-in" were randomly chosen. For the 3D Fibonacci grid and the "open" 4D version (FibonacciExt), random grid offsets and coordinate permutations were used. Geometric visualization of the setup is given in Fig. 4.

and obtain an estimate $\hat{\mathrm{E}}\{\cdot\}$ of the expected value by replacing the true pdf $f(\underline{x})$ by its Dirac mixture approximation $\hat{f}(\underline{x})$. This yields

$$\hat{\mathrm{E}}\{g(\underline{\boldsymbol{x}})\} = \int_{\mathbb{R}^N} g(\underline{x}) \cdot \hat{f}(\underline{x}) \,\mathrm{d}\underline{x} \tag{14}$$

$$= \int_{\mathbb{R}^N} g(\underline{x}) \cdot \frac{1}{L} \sum_{i=1}^{L} \delta(\underline{x} - \underline{\hat{x}}_i) \,\mathrm{d}\underline{x} \tag{15}$$

$$= \frac{1}{L} \sum_{i=1}^{L} g(\underline{\hat{x}}_i) \ . \tag{16}$$

Depending on the choice of $g(\cdot)$, the expected value $\mathrm{E}\{g(\underline{\boldsymbol{x}})\}$ can be the mean value of the pdf $f(\underline{x})$, the correlation, the covariance, and others. In our example, we choose the number of dimensions to $N = 2$ for ease of visualization of the setup and the function $g(\cdot)$ as

$$g(\underline{x}) = \left| \underline{x} - \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right|_2 = \sqrt{\left(x^{(1)} - 1\right)^2 + \left(x^{(2)}\right)^2} \ . \tag{17}$$

The non-Gaussian pdf $f(\underline{x})$ in our examples can be written as a product of a prior pdf $f^{\mathrm{p}}(\underline{x})$ and a likelihood $\Lambda(\underline{x})$, where

$$\underline{\mu}_M = \begin{bmatrix} 0 & 0.5 \end{bmatrix}^\top \ , \tag{18}$$

$$\mathcal{N}\left(\underline{x},\ \underline{\mu},\ \mathbf{C}\right) = |2\pi\mathbf{C}|^{-N/2} \exp\left\{-\frac{1}{2}\,\underline{x}^{\top}\mathbf{C}^{-1}\underline{x}\right\} \quad , \tag{19}$$

$$f^{\mathrm{p}}{}_{1,2}(\underline{x}) = \mathcal{N}\left(\underline{x},\ \pm\underline{\mu}_M,\ \mathbf{I}\right) \quad , \tag{20}$$

$$\Lambda_{1,2}(\underline{x}) = \mathcal{N}\left(\left|\underline{x} \mp \underline{\mu}_M\right|_2^2,\ 1,\ 0.2\right) \quad , \tag{21}$$

$$f(\underline{x}) = f^{\mathrm{p}}{}_1(\underline{x})\Lambda_1(\underline{x}) + f^{\mathrm{p}}{}_2(\underline{x})\Lambda_2(\underline{x}) \ . \tag{22}$$

See Fig. 4 for visualizations of the $f(\underline{x})$ as defined in (22), together with various sampling results. Note that it is not important that $f(\underline{x})$ is properly normalized. We only need to know the constant c, and a boundary $\mathcal{B}$ containing all the significant probability mass, or the Gaussian proposal. The mode $\underline{x}_{\mathrm{mode}}$ of $f(\underline{x})$ in (22) has been obtained by numerical optimization,

$$\underline{x}_{\mathrm{mode}} \approx \begin{bmatrix} 0.873 \\ -0.0220 \end{bmatrix} \quad , \tag{23}$$

$$1/c = f(\underline{x}_{\mathrm{mode}}) \approx 1.033 \ . \tag{24}$$

Quantitative evaluation results are shown in Fig. 5.

A MATLAB implementation of the proposed sampling method for arbitrary continuous pdfs is published along with this paper. You can find the source code in IEEE Xplore Code Ocean[1].

VI. Conclusions

The proposed method for rejection sampling is simple to implement, works with arbitrary multivariate pdfs, and does not require a normalized pdf.

In a first step, optimal deterministic uniform (or Gaussian) proposal samples are generated in higher-dimensional spaces (original dimensions plus one or two auxiliary dimensions). In a second step, rejection sampling simply selects those samples from "under" the pdf and projects them to the original space, which yields the desired samples.

The proposal samples in the higher-dimensional space are generated using generalized Fibonacci grids as optimal deterministic samples. With generalized Fibonacci grids, large numbers of optimal deterministic uniform and Gaussian samples can be generated quickly (without optimization required) for the first time. Also, nearly-optimal deterministic samples of arbitrary pdfs can be generated via acceptance-rejection as demonstrated in this work: by acceptance-rejection and projection onto the lower-dimensional original space, the optimal deterministic samples lose some of their optimality but are still superior to state-of-the-art quasi-random samples.

In the future, we will follow up with lattice rules and Kronecker rules for Purser's generalized Fibonacci grids. These will allow for optimal deterministic generated fast and sequentially one by one even in higher dimensions.

[1] https://codeocean.com/capsule/0257002/tree

References

[1] W. M. Wonham, "Some Applications of Stochastic Differential Equations to Optimal Nonlinear Filtering," *Journal of the Society for Industrial and Applied Mathematics Series A Control*, vol. 2, no. 3, pp. 347–369, 1964. [Online]. Available: https://doi.org/10.1137/0302028

[2] J. Von Neumann, "Various Techniques Used in Connection With Random Digits," *John von Neumann, Collected Works*, vol. 5, pp. 768–770, 1963.

[3] B. D. Flury, "Acceptance–Rejection Sampling Made Easy," *SIAM Review*, vol. 32, no. 3, pp. 474–476, 1990.

[4] W. K. Hastings, "Monte Carlo Sampling Methods Using Markov Chains and Their Applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970. [Online]. Available: http://www.jstor.org/stable/2334940

[5] R. M. Neal, "Slice Sampling," *The Annals of Statistics*, vol. 31, no. 3, pp. 705 – 767, 2003.

[6] H. Niederreiter, *Random Number Generation and Quasi-Monte Carlo Methods*. Society for Industrial and Applied Mathematics, 1992.

[7] X. Wang, "Improving the Rejection Sampling Method in Quasi-Monte Carlo Methods," *Journal of Computational and Applied Mathematics*, vol. 114, no. 2, pp. 231–246, 2000.

[8] N. Nguyen and G. Ökten, "The Acceptance-Rejection Method for Low-Discrepancy Sequences," 2014.

[9] H. Zhu and J. Dick, "Discrepancy Bounds for Deterministic Acceptance-Rejection Samplers," *Electronic Journal of Statistics*, vol. 8, no. 1, pp. 678 – 707, 2014.

[10] U. D. Hanebeck and V. Klumpp, "Localized Cumulative Distributions and a Multivariate Generalization of the Cramér-von Mises Distance," in *Proceedings of the 2008 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2008)*, Seoul, Republic of Korea, Aug. 2008, pp. 33–39.

[11] J. Steinbring, M. Pander, and U. D. Hanebeck, "The Smart Sampling Kalman Filter with Symmetric Samples," *Journal of Advances in Information Fusion*, vol. 11, no. 1, pp. 71–90, June 2016.

[12] D. Frisch and U. D. Hanebeck, "Efficient Deterministic Conditional Sampling of Multivariate Gaussian Densities," in *Proceedings of the 2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2020)*, Virtual, Sept. 2020.

[13] U. D. Hanebeck, "Deterministic Sampling of Multivariate Densities based on Projected Cumulative Distributions," in *Proceedings of the 54th Annual Conference on Information Sciences and Systems (CISS 2020)*, Princeton, New Jersey, USA, Mar. 2020.

[14] D. Prossel and U. D. Hanebeck, "Dirac Mixture Reduction Using Wasserstein Distances on Projected Cumulative Distributions," in *Proceedings of the 25th International Conference on Information Fusion (Fusion 2022)*, Linköping, Sweden, July 2022.

[15] R. J. Purser, "Generalized Fibonacci Grids; A New Class of Structured, Smoothly Adaptive Multi-Dimensional Computational Lattices," *Office note (National Centers for Environmental Prediction (U.S.))*, 2008.

[16] R. Swinbank and R. James Purser, "Fibonacci Grids: A Novel Approach to Global Modelling," *Quarterly Journal of the Royal Meteorological Society*, vol. 132, no. 619, pp. 1769–1793, 2006.

[17] R. Marques, C. Bouville, K. Bouatouch, and J. Blat, "Extensible Spherical Fibonacci Grids," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 4, pp. 2341–2354, 2021.

[18] D. Frisch and U. D. Hanebeck, "Deterministic Gaussian Sampling With Generalized Fibonacci Grids," in *Proceedings of the 24th International Conference on Information Fusion (Fusion 2021)*, Sun City, South Africa, Nov. 2021.