

RESEARCH ARTICLE

Practical sampling of constraint-based models: Optimized thinning boosts CHRR performance

Johann F. Jadebeck^{1,2}, Wolfgang Wiechert^{1,2}, Katharina Nöh^{1*}

1 Institute of Bio- and Geosciences, IBG-1: Biotechnology, Forschungszentrum Jülich, Jülich, Germany, **2** Computational Systems Biotechnology (AVT.CSB), RWTH Aachen University, Aachen, Germany

* k.noeh@fz-juelich.de



Abstract

Thinning is a sub-sampling technique to reduce the memory footprint of Markov chain Monte Carlo. Despite being commonly used, thinning is rarely considered efficient. For sampling constraint-based models, a highly relevant use-case in systems biology, we here demonstrate that thinning boosts computational and, thereby, sampling efficiencies of the widely used Coordinate Hit-and-Run with Rounding (CHRR) algorithm. By benchmarking CHRR with thinning with simplices and genome-scale metabolic networks of up to thousands of dimensions, we find a substantial increase in computational efficiency compared to unthinned CHRR, in our examples by orders of magnitude, as measured by the effective sample size per time (ESS/t), with performance gains growing with polytope (effective network) dimension. Using a set of benchmark models we derive a ready-to-apply guideline for tuning thinning to efficient and effective use of compute resources without requiring additional coding effort. Our guideline is validated using three (out-of-sample) large-scale networks and we show that it allows sampling convex polytopes uniformly to convergence in a fraction of time, thereby unlocking the rigorous investigation of hitherto intractable models. The derivation of our guideline is explained in detail, allowing future researchers to update it as needed as new model classes and more training data becomes available. CHRR with deliberate utilization of thinning thereby paves the way to keep pace with progressing model sizes derived with the constraint-based reconstruction and analysis (COBRA) tool set. Sampling and evaluation pipelines are available at <https://jugit.fz-juelich.de/IBG-1/ModSim/fluxomics/chrrt>.

OPEN ACCESS

Citation: Jadebeck JF, Wiechert W, Nöh K (2023) Practical sampling of constraint-based models: Optimized thinning boosts CHRR performance. PLoS Comput Biol 19(8): e1011378. <https://doi.org/10.1371/journal.pcbi.1011378>

Editor: Christos A. Ouzounis, CPERI, GREECE

Received: January 4, 2023

Accepted: July 21, 2023

Published: August 11, 2023

Copyright: © 2023 Jadebeck et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: The source code and data used to produce the results and analyses presented in this manuscript are available at jugit repository: <https://jugit.fz-juelich.de/IBG-1/ModSim/fluxomics/chrrt>.

Funding: JFJ received funding from the Helmholtz Association of German Research Centres and performed this work as part of the Helmholtz School for Data Science in Life, Earth and Energy (HDS-LEE). The funder had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Author summary

Analyzing the parameter spaces of genome-scale metabolic models (GEM) by means of Markov chain Monte Carlo (MCMC) sampling has become a key method in systems biology. In this context, sub-sampling, or *thinning*, reduces storage requirements and post-processing efforts of the immense sample volumes. However, sub-sampling is typically applied without due consideration, despite statisticians arguing that, by increasing the

Competing interests: The authors have declared that no competing interests exist.

variance of the resulting estimate, thinning is almost always statistically inefficient. Considering synthetic and real sampling problems of widely varying complexity, we show that for the state-of-the-art uniform MCMC sampling algorithm, Coordinate Hit-and-Run with Rounding (CHRR), thinning has dramatic consequences on sampling performances: Our benchmarks reveal that sub-sampling boosts CHRR efficiencies, often by orders of magnitude, and the performance gain scales with problem dimension. For the studied problem classes, we provide simple rules of thumb for performance-optimized thinning, for which we give out-of-sample evidence. Utilization of our thinning guideline for GEMs, hence, keeps promise to solve hitherto inaccessible sampling problems, while for large scale networks its application can make the difference between immediate sampling success and repeated failures. In summary, on the one hand, optimized tuning of CHRR is now easily possible. On the other hand, thinning choice has to be registered when benchmarking CHRR implementations with regard to their running times.

1 Introduction

Constraint-based modelling of metabolism provides a versatile mathematical framework for integrating genomic and biochemical knowledge with multi-omics data and to interrogate network models [1–3]. As such, the constraint-based reconstruction and analysis (COBRA) methodology has established an indispensable tool set with numerous applications in metabolic engineering, biotechnology, microbiology, and systems as well as synthetic biology [4–7]. Nowadays, genome-scale models (GEMs) are routinely reconstructed from genomic data using automatized workflows [8]. Powerful reconstruction pipelines such as CaveMe [9], AGORA2 [10], or metaGEM [11] generate high-quality models with increasing sizes and at accelerated speed [12]. For instance, recently these tools enabled to generate more than 7,300 context-specific yeast GEMs [10]. Multi-organismal GEM repositories are publicly available, such as the BiGG database [13, 14] that contains 108 models, with the largest metabolic model, *Recon3D* [15], accounting for more than ten thousand reactions. Nowadays, modeling initiatives are starting to target the description of microbial communities, pan- and meta-genomes, and even the whole human body [16–18], therewith escalating model sizes further.

GEMs are particularly useful to predict phenotypes, to suggest genetic interventions, e.g., by applying bi-level strain design approaches [19], and to foresee the outcome of experiments [20]. Besides comprehensiveness, the capabilities of GEMs to deliver useful insights is intimately tied to the knowledge of the unknown model parameters, i.e., the intra- and extracellular metabolic reaction rates (fluxes). Here, in particular, enzyme constraints have been shown to effectively improve the models' prediction performances [21]. The stoichiometry-induced mass balances at steady-state and further constraints introduce linear dependencies among the fluxes, implying that all feasible flux configurations are located within a convex hyper-polytope [22, 23], with a dimension dictated by the nullspace of the stoichiometric matrix. Interestingly, such convex polytopes as solution spaces arise in many applied modeling contexts, including operations research [24], ecological modeling [25], computational finance [26], astronomy [27], physics [28], and everywhere where network-flow problems are to be solved.

The expansion in GEM sizes has raised the need for developing scalable tools to analyze the high-dimensional flux solution spaces. In this context, uniform convex polytope sampling (UCPS), i.e., drawing representative random numbers from a truncated uniform distribution defined over the bounded flux polytope, is routinely used for characterizing the solution spaces of metabolic [29, 30] and gene networks [31], identification of metabolic flux couplings [32],

design of experiments [20, 33], and assessing the effects of uncertainties in biochemical network formulations [34, 35]. A simple UCPS strategy is *rejection sampling*, a Monte Carlo technique that iteratively generates samples within easily accessible objects such as polytope-enclosing (hyper)cubes, where it keeps only those draws that are located interior to the polytope, until the truncated uniform target distribution is approximately covered [36]. While initial efforts in the field of constraint-based modeling used this technique [37], rejection sampling was quickly recognized to be computationally infeasible for more comprehensive network models, because for every feasible sample the number of rejections grows super-exponentially with polytope (effective model) dimension.

The class of Hit-and-Run (HR) algorithms has been developed to overcome this bottleneck [38]. HR is a Markov chain Monte Carlo (MCMC) technique, which, starting from an initial flux configuration within the polytope, generates a sequence of flux states (a Markov chain) by drawing randomly from inner chords directed in random directions [39]. As a result, the generated states (or a fraction of them) are called *MCMC samples*. These samples, albeit being correlated by construction, approximate the target distribution asymptotically [40].

How quickly the target distribution is approximated well in terms of wall-clock time, the so-called *sampling efficiency*, is critical for practical applicability of HR. Sampling efficiency is determined by two components: (1) the statistical efficiency, which specifies how well a set of samples represents a target distribution, and (2) the computational efficiency of the HR sampling algorithm, i.e., how many computational operations are required per generated state. Formally, the statistical efficiency of N samples produced by a Markov chain is given by the *effective sample size* (ESS) [36]:

$$ESS = \frac{N}{1 + 2 \cdot \sum_{t=1}^{\infty} \rho_t} \quad (1)$$

with ρ_t being the autocorrelation of the sequence at lag t . The sampling efficiency of the MCMC implementation is then measured by the ESS normalized to wall clock time t , i.e., ESS/t .

According to Eq (1), two strategies to raise MCMC sampling efficiencies are (1) reducing the computational cost per sample, and (2) decreasing the autocorrelation between subsequent states. For HR, the cost per sample is reduced by updating the states in a pure coordinate-wise fashion instead of drawing random directions, a technique also known as Gibbs sampling [41]. To decrease the autocorrelation, the average distance between subsequent states is to be increased. In UCPS, the achievable distance is limited by the geometry of the polytope. Intuitively, in isotropic geometries on average longer steps are possible than in anisotropic spaces. Since heterogeneous parameter scales in GEMs are the rule, an affine transformation is employed to homogenize the flux scales prior to sampling, the so-called *rounding transformation* [22]. After sampling, the states have to be mapped back to the original space by inverting the rounding transformation. Together, the two strategies culminated in the celebrated Coordinate Hit-and-Run with Rounding (CHRR) algorithm [42], the tried-and-tested workhorse for UCPS in the COBRA domain [4, 43, 44], with several implementations being available [42, 45–47].

CHRR is commonly applied together with *thinning*, a MCMC post-processing technique, to reduce the memory footprint of the sampling results. The simplest and most widely used form of thinning is *fixed frequency* thinning with thinning constant τ , which means that only every τ^{th} sample is kept. While thinning indeed reduces the autocorrelation of the samples generated by CHRR, the reduction comes at the cost of producing an approximation of the target distribution that is less accurate. Indeed, thinning is known to be statistically inefficient in all but very few cases [48, 49].

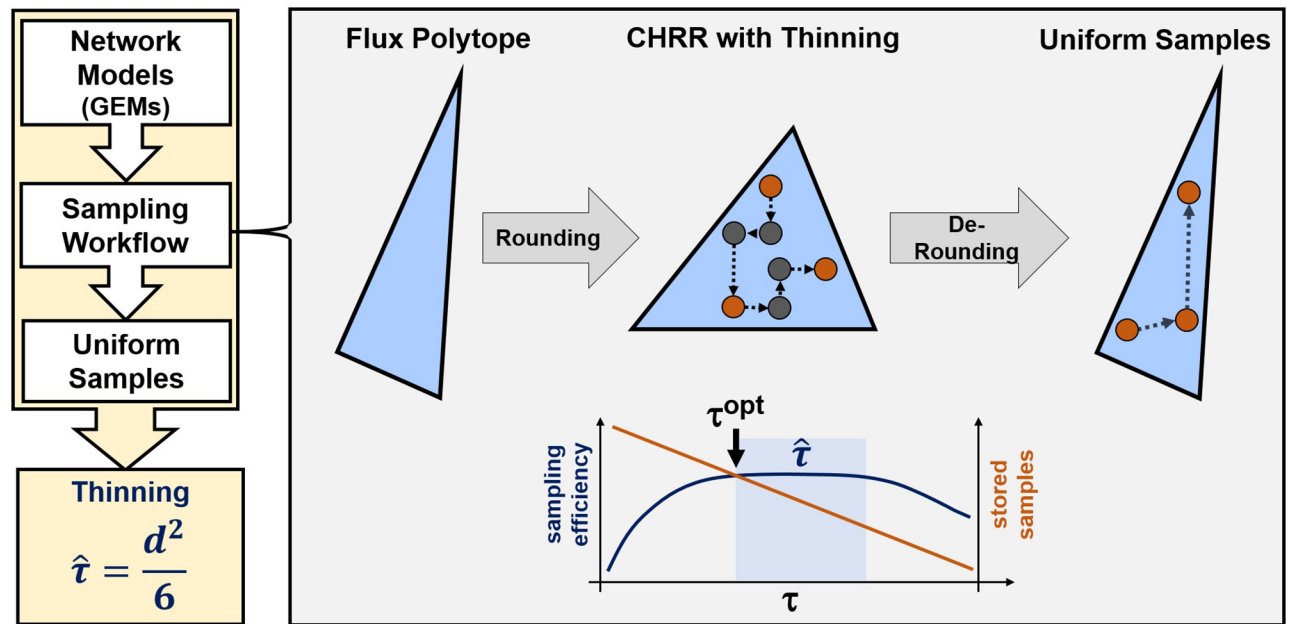


Fig 1. Overview of the CHRR tuning workflow. CHRR takes in a flux polytope that is sampled after bringing the polytope in a more favorable shape (rounding). By thinning, samples (grey) are deliberately dropped before transforming the remaining samples back to the original space (de-rounding). Finally, the samples (red) represent the uniform samples of the given GEM. Thinning is performance critical for CHRR, because de-rounding implies costs per-sample. Between the extremes of discarding too many samples and spending too much compute resources for de-rounding highly-correlated samples, an optimal thinning constant τ^{opt} exists. By running benchmarks for different GEMs and thinning constants, we derive a guideline for choosing $\tilde{\tau}$, close enough to τ^{opt} in efficiency, to sample unseen GEMs quickly.

<https://doi.org/10.1371/journal.pcbi.1011378.g001>

Recognizing the “need for speed” to sample large-scale GEMs, in this work, we study the impact of thinning for CHRR systematically from the perspective of practical sampling efficiency, based on principled statistical criteria (cf. Fig 1). By analyzing simplices and GEMs with widely varying dimensionality, we demonstrate that CHRR is one of the rare cases for which thinning boosts sampling efficiencies, measured in terms of ESS/t , and prove current thinning practices computationally wasteful. From our benchmarks, we derive simple, yet effective and statistically rigorous guidelines to effectively and efficiently tune CHRR, which we verify using large-scale GEMs such as *Recon3D*. Our findings thus guide practical sampling efforts of COBRA models towards achieving more effective samples with less compute resources, thereby updating previous wisdom [42].

2 Methods and models

We start with a brief background on UCPS, describe the pre- and post-processing of flux polytopes relevant for the CHRR sampling procedure, list test and validation problems, and give details about MCMC diagnostic checks along with implementation details.

2.1 Convex GEM flux polytopes and their pre- and post-processing

GEMs are stoichiometric network models that are compiled from genomic information and biochemical knowledge [50]. By mass balancing at steady-state, linear inequality systems are derived from these models for the D unknown metabolic reaction rates (fluxes) $v \in \mathbb{R}^D$:

$$A'_{eq} \cdot v = b'_{eq} \quad (2)$$

where $A'_{eq} \in \mathbb{R}^{n'_{eq} \times D}$ is the (extended) stoichiometric matrix, and $b'_{eq} \in \mathbb{R}^{n'_{eq}}$ contains the time derivatives of the (intra- and extracellular) metabolite concentrations. As a convention, we use the symbol “ $\cdot$ ” for the matrix-vector product. In addition, the fluxes v are subject to linear inequalities originating from physiologically motivated lower and upper limits on their values:

$$A'_{in} \cdot v < b'_{in} \quad (3)$$

with $A'_{in} \in \mathbb{R}^{n'_{in} \times D}$ the constraint matrix and $b'_{in} \in \mathbb{R}^{n'_{in}}$ contains the flux bounds. The convex polytope (Eqs 2 and 3) is constructed such that it is always bounded in all flux directions, otherwise the uniform sampling problem is ill-defined. Since A'_{in} typically contains redundant constraints that cause numerical instabilities in the sampling workflow, in a first pre-processing step, the redundant inequalities are eliminated from Eq (3), giving

$$A_{in} \cdot v < b_{in} \quad (4)$$

with $A_{in} \in \mathbb{R}^{n_{in} \times D}$ the constraint matrix and $b_{in} \in \mathbb{R}^{n_{in}}$. Then, inequalities that are effectively equalities (fluxes bound to ranges equal or smaller than $1e-7$) are identified and reformulated as equalities, which creates the modified stoichiometric system

$$A_{eq} \cdot v = b_{eq} \quad (5)$$

with $A_{eq} \in \mathbb{R}^{n_{eq} \times D}$ and $b_{eq} \in \mathbb{R}^{n_{eq}}$. Together, Eqs 4 and 5 define the D -dimensional effective convex-bounded flux polytope of the given GEM:

$$\mathcal{P}_D = \{v \in \mathbb{R}^D \mid A_{eq} \cdot v = b_{eq} \wedge A_{in} \cdot v < b_{in}\} \quad (6)$$

We note that by the pre-processing $\mathcal{P}_D$ is slightly smaller than its original counterpart defined by Eqs (2) and (3). However, because the threshold, below which inequalities are interpreted as equalities, is selected to be orders of magnitude smaller than the achievable flux precision, the impact of the shrinkage on the flux values is in fact marginal. Instead, the beneficial effect on the numerical stability of the sampling algorithm outweighs the risk of losing relevant flux information [44].

Next, the equalities (5) are exploited to reduce the dimension of the sampling problem, without changing the polytope volume, by expressing the polytope $\mathcal{P}_D$ in terms of independent, still interpretable flux coordinates $v^{\text{indep}} \in \mathbb{R}^d$, $d \leq D$ [51]. Therefore, we determine a basis $K \in \mathbb{R}^{D \times d}$ of the null space of the modified stoichiometric system (5), subject to the inequality constraints (4):

$$A_{in} \cdot (K \cdot v^{\text{indep}} + v_0) < b_{in} \quad (7)$$

where v_0 is a particular solution within the flux polytope $\mathcal{P}_D$. A typical choice for v_0 is the Chebyshev center of $\mathcal{P}_D$. By going from Eqs (4) and (5) to Eq (7), all equalities are eliminated from $\mathcal{P}_D$ in Eq (6), resulting in the dimension-reduced flux polytope $\mathcal{P}_d$:

$$\mathcal{P}_d = \{v^{\text{indep}} \in \mathbb{R}^d \mid A \cdot v^{\text{indep}} < b\} \quad (8)$$

where $A = A_{in} \cdot K \in \mathbb{R}^{n_{in} \times d}$ and $b = b_{in} - A_{in} \cdot v_0 \in \mathbb{R}^{n_{in}}$ with v_0 and $n_{in} > d$ in all but toy networks. Hitherto, we call d the *effective* model dimension. As an example, *Recon3D*, as downloaded from BiGG, has 10,600 bounded reactions. After dimension-reduction and flux coordinate transformation, $n_{in} = 11,195$ inequality constraints and $d = 4,861$ effective dimensions remain.

Flux polytopes are known to have anisotropic shapes, a characteristic that slows down the mixing of MCMC sampling schemes (see [22] for details). To curb for this, dimension-

preserving so-called rounding transformations have been proposed that transform the polytope $\mathcal{P}_d$ into a flux polytope $\mathcal{P}_d^{\text{round}}$ having a more isotropic shape [22]. For more technical discussions of the isotropy of convex polytopes, we refer to [39, 52, 53]. In this work, using the approach in [44], we determine the linear rounding transformation $R \in \mathbb{R}^{d \times d}$ by computing the Maximum Volume Ellipsoid (MVE) inscribed in $\mathcal{P}_d$ [54]. Specifically, the inverse mapping R^{-1} is determined such that it maps the MVE to the unit sphere. The rounded polytope $\mathcal{P}_d^{\text{round}}$ is then given by:

$$\mathcal{P}_d^{\text{round}} = \{v^{\text{round}} \in \mathbb{R}^d \mid A_R \cdot v^{\text{round}} < b\} \quad (9)$$

with $A_R = A \cdot R^{-1} \in \mathbb{R}^{n_{in} \times d}$. Finally, any d -dimensional independent flux vector v^{round} in the rounded space has to be mapped back to a flux vector v in the original D -dimensional flux space to be biologically interpretable, using the affine back-transformation $T: v^{\text{round}} \mapsto v$ given by:

$$v = K \cdot R^{-1} \cdot v^{\text{round}} + v_0 \quad (10)$$

2.2 Coordinate Hit-and-Run with Thinning

Thinning is an integral part of many MCMC algorithms. We hitherto denote our implementation of CHRR [42] that makes deliberate use of thinning by CHRRT. Algorithm 1 gives a pseudocode description of the CHRRT algorithm. With a UCPS task at hand, defined by Eqs (4) and (5), first the rounded polytope $\mathcal{P}_d^{\text{round}}$ is determined (L 1). Starting from an initial flux state, samples in $\mathcal{P}_d^{\text{round}}$ are iteratively generated by constructing chords along the coordinates (L 6–10). Fixed frequency thinning is applied with thinning constant τ (L 11). Lastly, every sample that is not discarded due to thinning is transformed back to the full dimensional flux space $\mathcal{P}_D$ (L 12).

Algorithm 1 Coordinate Hit-and-Run with Rounding and Thinning (CHRRT)

Input:

Equality constraints: $A_{\text{eq}} \cdot v = b_{\text{eq}}$
 Inequality constraints: $A_{\text{in}} \cdot v \leq b_{\text{in}}$
 Number of samples to store: N
 Thinning constant: τ
 Feasible starting state: $v_0 \in \mathcal{P}_D$

Result:

N uniformly distributed flux samples $v_1, \dots, v_N \in \mathcal{P}_D$

Procedure:

```

1: determine dimension-reduced polytope  $\mathcal{P}_d$ 
2: compute rounding transformation  $R$  and determine rounded polytope  $\mathcal{P}_d^{\text{round}}$ 
3: transform starting state:  $v_0^{\text{round}} \leftarrow T^{-1}(v_0)$ 
4:  $i \leftarrow 0$  ▷ Label iteration of CHRRT with  $i$ 
5: while number of stored samples  $< N$  do
6:    $k \leftarrow$  uniform random coordinate  $\in \{1, \dots, d\}$ 
7:   ▷ compute distances from  $v_i^{\text{round}}$  to borders of  $\mathcal{P}_d^{\text{round}}$ :
8:    $d_+, d_- \leftarrow$  distances from  $v_i^{\text{round}}$  to  $\partial \mathcal{P}_d^{\text{round}}$  along  $k^{\text{th}}$  coordinate
9:    $\lambda \sim \mathcal{U}(d_-, d_+)$  ▷ Sample step size  $\lambda$  uniformly from interval
10:   $v_{i+1}^{\text{round}} \leftarrow v_i^{\text{round}} + \lambda \cdot e_k$  ▷ Update in direction of unit vector  $e_k$ 
11:  if  $i$  modulo  $\tau$  is 0 then
12:    store result of  $T(v_{i+1}^{\text{round}})$  ▷ Back-transform sample to original flux space  $\mathcal{P}_D$ 
13:  end if
```

```

14:    $i \leftarrow i + 1$ 
15: end while

```

Concerning the computational costs of CHRR, determining an internal point v_0 and computing the rounded polytope $\mathcal{P}_d^{\text{round}}$ are one-time costs, both being independent of the numbers of samples. We therefore do not further consider these costs in this work. The most expensive operation per sample is the back-transformation T that maps the generated sample to the original polytope (Algorithm 1, L 12). The back-transformation amounts to a dense matrix-vector multiplication, with worst case costs of $\mathcal{O}(n_{in} \cdot d)$ per sample. All other operations are element-wise additions, multiplications, or divisions with per-sample costs of $\mathcal{O}(d)$.

2.3 Test problems

We consider two types of test problems, simplices representing well-defined and easy to scale problems, and GEMs being our main concern. A d -dimensional simplex $\mathcal{S}_d$ is defined by:

$$\mathcal{S}_d = \{x \in \mathbb{R}^d \mid \sum_{i=1}^d x_i \leq 1, x_i \geq 0 \quad \forall i \leq d\} \quad (11)$$

Being easily scalable to different dimensions, simplices are ideal to study the dependency of sampling complexity and polytope dimension. Different to GEMs, simplices are constructed by non-redundant constraints and, therefore, do not need to be dimension-reduced. However, as for GEMs, polytope rounding improves the geometric isotropy of simplices. Thus, Eq (10) remains valid, with the matrix K set to identity. In this work, simplices in 64, 256, 1,024, and 2,048 dimensions were benchmarked.

Besides simplices, we selected eleven GEMs of varying sizes for this study, eight for training our thinning guideline, and three (*Yeast8*, *ecYeast8*, *Recon3D*) for validation purposes. Effective polytope dimensions range from 24 to 4,861. In contrast to simplices, for GEMs there is no consistent relationship between the number of reactions, the number of constraints, and the effective polytope dimension d . We refer to Table A in S1 Appendix for more details on the test problems.

2.4 MCMC convergence diagnostics

Assessing whether a (sub)sequence of MCMC samples has converged to the target distribution is not only vitally important in practical inference, but also critical for reliable benchmarking sampling performances. The *ESS* and rank-normalized $\hat{R}$ values are computed for each of the D fluxes separately and set to the minimum/maximum value of these, i.e., considering the worst-case. We compute the *ESS* in the original flux space, because this is the space, for which we are interested in statistics about estimates, such as the mean flux. Particular care must be taken here, since the estimated *ESS* is typically noisy and may therefore be unreliable for sample sets that are far from being converged. We here follow the advice given by [55] to use four independent chains and compute the rank-normalized $\hat{R}$ diagnostic along with the *ESS* to test for convergence. Vehtari et al. [55] argue that the estimation of $\hat{R}$ and *ESS* requires reliable estimates of the variances and autocorrelations of the samples, that is, the estimation of $\hat{R}$ and *ESS* is only reliable, if the *ESS* of a set of samples is sufficiently large. Therefore, in our study, the threshold for convergence is set to $\hat{R} < 1.01$ and $ESS > 400$, which is suitable to a setup with four independent Markov chains. The limit of 400 for the *ESS* is derived by considering that for a single chain, each of the parts of the split chain (50/50 split) is required to contain $ESS > 50$, which amounts to a total of at least 400 when considering multiple, independent

chains [55]. Exemplary *ESS* and rank normalized $\hat{R}$ values are shown in Figs B and C in [S1 Appendix](#).

2.5 Implementation details

GEMs were downloaded (Table A in [S1 Appendix](#) for details) in SBML (Systems Biology Markup Language) format [56], and plugged into PolyRound v0.1.8, the state-of-the-art python package for polytope rounding [44]. The highly optimized polytope sampling library HOPS v3.0.0 was used for MCMC sampling [46]. CHRRT, as implemented in HOPS, was used for all benchmark runs to ensure their comparability. In each sampling run, four parallel chains were used according to recommendations by Vehtari et al. [55]. Samples, the rounded polytopes, and the accompanying transformations were stored. For each flux, trace plots were visually examined, the maximum rank-normalized $\hat{R}$ metric, and the minimal *ESS* over all fluxes were computed to check that MCMC convergence criteria are met as described in Sec. 2.4. Calculations are performed using arviz v0.12.1 [57]. All numerical experiments were run on an AMD Ryzen 9 5950X 16-Core Processor.

3 Results and discussion

After motivating the idea of using thinning in CHRR to reallocate computational resources, we quantify achievable sampling efficiencies for the test problems. From that, we derive simple and useful guidelines for optimized thinning choice, which we compare to previous advice using three validation problems.

3.1 The role of thinning for CHRR

In constraint-based metabolic modeling, some studies using CHRR report thinning constants of 100, 1000, and 10,000 [4, 43]. However, these works neither do provide guidance for a suitable selection of τ for a sampling problem at hand, nor do they discuss the implications of the different choices of τ . On the other hand, Haraldsdóttir et al [42] pointed out that τ should be selected depending on the problem dimension. Specifically, the authors suggest setting τ , as a rule of thumb, to eight times the squared polytope dimension, stating that this choice ensures statistical independence of the thinned samples. However, in this work no derivation or their rule or further underpinning of their argument is given.

Spurred by the seminal theoretic work of Geyer [48], we set out to investigate whether the sampling efficiency of CHRR benefits from thinning. The question is whether, for a fixed computational budget, CHRR with $\tau > 1$ converges faster than unthinned CHRR with $\tau = 1$. To this end, we consider the *ESS* of a set of N samples in relation to the time cost of producing the samples, $t_{N,\tau}$. The time cost $t_{N,\tau}$ is given by the sum of the time costs t_{update} for producing $N \cdot \tau$ states v_i^{round} in Algorithm 1 (L 6–10), and the time cost $t_{\text{transform}}$ of transforming the N samples back to the original flux space (Algorithm 1, L 12):

$$t_{N,\tau} = N \cdot (\tau \cdot t_{\text{update}} + t_{\text{transform}}) \quad (12)$$

Geyer argued that for thinning to be beneficial, the time cost $t_{N,\tau}$ has to grow slower with τ than the *ESS*, indicating that thinning is not advantageous for every combination of sampling problem and MCMC algorithm [48]. In this vein, Link et al. criticize the routine application of thinning for applications in ecology, where thinning is often detrimental to sampling efficiency [49].

To investigate the role of thinning for UCPS using our CHRRT implementation, we measured t_{update} and $t_{\text{transform}}$ for a set of synthetic and real test problems (Table A in [S1](#)

Appendix). For both time costs, we took the wall-clock times and averaged them over the number of times each operation was applied. The resulting time ratios $t_{\text{transform}}/t_{\text{update}}$ are shown in Fig 2. Among all test problems, the time ratios for only the two models with the fewest dimensions, i.e., the 64-dimensional simplex and the 24 dimensional *E. coli* core model, are below ten. For the remaining investigated models, $t_{\text{transform}}$ is up to three orders of magnitude larger than t_{update} . Precisely, for the simplices, the time ratio increases quadratically with dimension. We also observed a stark increase in the ratio for GEMs. Different to simplices, however, with GEMs additional factors contribute to the time ratios, such as the number of fluxes D (more fluxes increase $t_{\text{transform}}$) and the number of constraints n_{in} (more constraints increase t_{update}), resulting in a less succinct relation between effective model dimensions and time ratios.

Across all test problems, $t_{\text{transform}}$ is generally considerably larger than t_{update} , with a time ratio of at least four for the smallest tested models, but which starkly increases with the dimension of the sampling problem. Given such large time ratios, it is plausible to expect thinning to be beneficial for the sampling efficiency of CHRR in the context of UCPS. However, only by considering the relation between the time ratios and the autocorrelation mediated by the ESS, which is specific to the combination of model characteristic and CHRR, we are able to find out if thinning actually is practically beneficial. In conclusion, low autocorrelation should not be the only deciding factor when suggesting rules of thumb for thinning constants.

3.2 Benchmarking CHRR from the perspective of thinning

To study how the performance of CHRR depends on thinning for the two classes of test problems, we benchmarked the sampling efficiency, in terms of ESS/t , for a wide variety of thinning constants. To find thinning constants that yield high ESS/t values, pre-runs with $\tau = d$ were performed for each test problem. From the pre-runs, we estimated appropriate ranges of problem-specific thinning constants and selected a set of at least five τ 's distributed over that range.

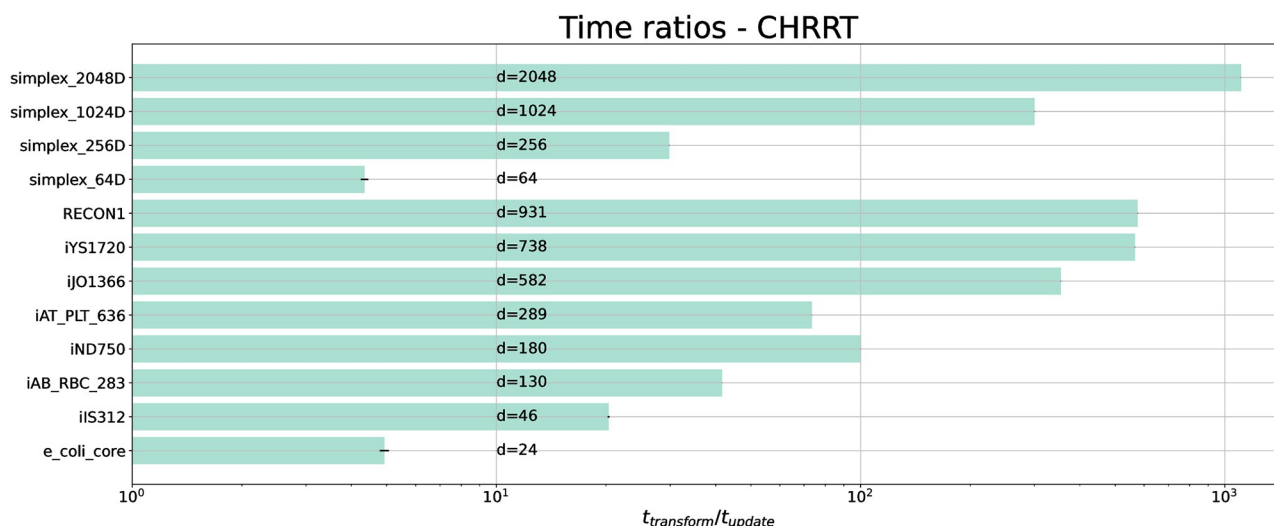


Fig 2. Mean time ratios of one back-transform $t_{\text{transform}}$ vs. one CHRR update t_{update} for simplices and GEMs (horizontal axis scaled logarithmically). Ratios indicate that in all cases, the cost to transform a sample back to the original flux space exceeds the cost of generating a sample by far. In all but the smallest test problems, the error bars of four runs are too small ($< 1\%$) to be visible. For the simplices the ratio increases quadratically with dimension ($0.00024 \cdot d^2 + 0.044 \cdot d + 1.4$). For GEMs, there is a strong trend towards an increase in ratio with model dimension. Details on the models are found in Table A in S1 Appendix.

<https://doi.org/10.1371/journal.pcbi.1011378.g002>

For simplices and GEMs, Fig 3 shows the obtained ESS/t , relative to the unthinned CHRR baseline. The plots fan out in a series of curves for the test problems with different model dimensions. As a rule, for higher dimensions, higher relative ESS/t values are achieved. All curves share the characteristic that with increasing τ , the relative ESS/t first increases before dropping again. The peak indicates the optimal thinning constant choice, specific to the investigated test problem. While for simplices pronounced peaks emerge, the curves for GEMs show plateaus, indicating that a range of thinning constants perform equally well. This apparent insensitivity in ESS/t performance for GEMs is convenient, since it implies that hyperparameter tuning of τ does not need to be precise for achieving near optimal ESS/t values, if its value is set to a suitable order of magnitude. Clearly, thinning constants at the lower end of the plateau are to be preferred due to a better resource usage (see below).

To study how the optimal (in terms of sampling efficiency) thinning constant relates to the problem dimension, we then quantified the speed-up of CHRR for the apparently best thinning constant, hitherto denoted $\hat{\tau}$, among the set of tested constants. Results are shown in Fig 4A. For simplices, relative speed-ups grow roughly to the square of the polytope dimension. The relative speed-ups achieved for GEMs also grow with effective model dimension, with values ranging from 6 for the *E. coli* core model up to 770 for *Recon1*. For the latter, CHRR with $\hat{\tau}$ required only 0.32 s to converge and reach an ESS of 4,954, the unthinned variant required almost 80 times longer (25 s) to converge, while it reached an ESS of only 507. Notably, the speed-ups observed for GEMs surpass those observed for simplices of similar dimensions. Hence, our benchmarks show that the beneficial effect of thinning, when appropriately tuned, is not only plausible in theory, but it is also of high practical relevance for solving high-dimensional UCPS problems.

Since thinning drastically affects performances, our findings imply that the consequences of selecting τ values need to be taken into consideration when benchmarking novel aspiring

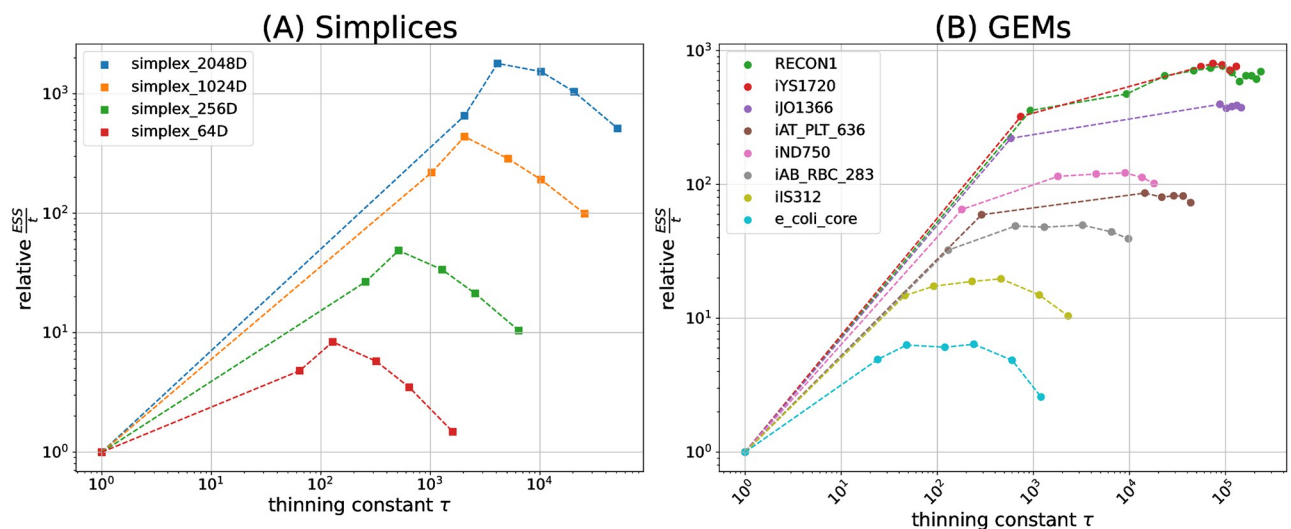


Fig 3. Double logarithmic plot of sampling efficiencies, relative to the unthinned sampling efficiencies, for selected thinning constants τ . The corresponding absolute ESS/t values are provided in Fig A in S1 Appendix. (A) Benchmark results for simplices. The peak efficiency is reached for a thinning constant $\hat{\tau} = 2 \cdot d$, with d being the number of effective dimensions. Over all, the maximum speed-up is achieved for the 2,064-dimensional simplex, being 1,798 times faster in terms of ESS/t compared to unthinned CHRR. (B) Benchmark results for GEMs. Setting $\tau > 1$ generally improves the ESS/t . For increasing d , larger thinning constants further boost the sampling efficiency. In all cases, the emerging plateaus indicate that the optimal ESS/t is not overly sensitive to the precise choice of the thinning constant.

<https://doi.org/10.1371/journal.pcbi.1011378.g003>

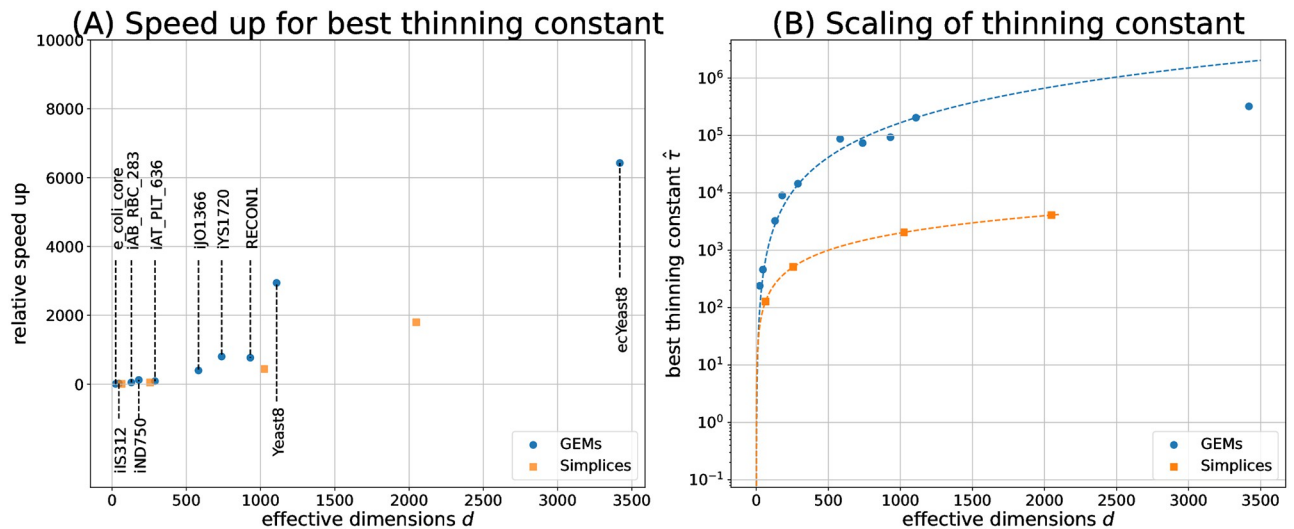


Fig 4. Performances of best thinning constants. (A) Speed-ups achieved for the best thinning constants $\hat{\tau}$, compared to the unthinned baseline, for simplices (orange squares) and GEMs (blue circles). (B) Scaling of the best thinning constant given the effective model dimensions d (vertical axis scaled logarithmically). Guidance for the selection of useful thinning constants $\hat{\tau}$ for GEMs (blue line) and simplices (orange line) is obtained by regression. For problem dimensions above 500, $\hat{\tau}$ is about two orders of magnitude larger for GEMs than for simplices.

<https://doi.org/10.1371/journal.pcbi.1011378.g004>

UCPS algorithms against CHRR, as for example the truncated log-concave sampling with reflective Hamiltonian Monte Carlo [53] or the Riemannian Hamiltonian Monte Carlo in a Constrained Space [58].

3.3 Simple thinning guidelines for simplices and GEMs

We use the previously benchmarked GEMs and simplices as training data for deriving practical guidelines on how to configure thinning for CHRR for optimized performance. For d -dimensional simplices, the linear relation

$$\hat{\tau}_{\text{simplex}} = 2 \cdot d \quad (13)$$

explains the measured data perfectly. For GEMs, on the other hand, a quadratic relation of the form $0.164 \cdot d^2$ matches the data well. Considering Fig 3B, indicating that optimal sampling performance is not overly sensitive to thinning constant choice, as a memorable rule of thumb for efficient GEM sampling, we propose:

$$\hat{\tau}_{\text{GEM}} = \frac{d^2}{6} \quad (14)$$

which only marginally deviates (less than 1.5%) from the fitted line.

Compared to the advice previously given by Haraldsdóttir et al. [42], i.e., $8 \cdot d^2$, our GEM guideline advocates thinning constants that are a factor of 48 smaller. To put the difference between the former and our updated guideline into perspective, a 48-fold increase in τ means a 48-fold increase in CHRR update steps, of which the vast majority is discarded. As we have argued before, discarding samples can lead to an improvement of sampling efficiency, but only if the loss of information due to dropping samples is over-compensated by information from additional CHRR update steps. In addition, in Fig 3 we show that selecting the thinning constant too large risks missing the performance peak, which then wastes computational

resources. In particular, we observe that a 48-fold increase in τ would lead to lower sampling efficiencies for eight out of the twelve test problems, because the performance peak would be clearly missed. For the remaining four test-problems, i.e., GEMs of higher dimensionality, we did not attempt to characterize the plateau around the best thinning constants, since this would be of little practical relevance. It is generally preferable to set the thinning constant to the minimal value on the plateau, because then less work is discarded, and more samples are stored than for larger thinning constants, while achieving the same ESS/t . Nonetheless, it is imaginable that the performance plateaus for the four largest GEMs stretch across an even wider range of thinning constants, meaning that the difference in ESS/t resulting from the previous rule of Haraldsdóttir et al. [42] and our guideline may turn out to be smaller.

While more work could be invested into fine-tuning $\hat{\tau}$ for the problem at hand, this risks investing more computational work than is eventually gained, because the estimation of ESS is noisy for short CHRR runs. In summary, we showed that a previous guideline for GEM sampling is suboptimal from the perspective of sampling efficiency. Instead, we promote a new GEM guideline, given by Eq (14), which empirically maximizes sampling performance without unnecessarily discarding computational work (green computing).

3.4 Result verification

To verify our proposed guideline for GEM sampling using CHRR, we applied it to three large, out-of-sample GEMs: *Yeast8* (1,108 dimensions), *ecYeast8* (3,4109 dimensions), as well as *Recon3D* (4,861 dimensions), the largest GEM currently available in the BiGG database [13]. All validation models were larger than the largest training model (*Recon1*), cf. Table A in S1 Appendix).

Given the large validation models, it is computationally infeasible to benchmark inefficient thinning constants (especially unthinned CHRR). Therefore, we need alternative ways to test, whether our guideline leads to optimized sampling. For the yeast models, *Yeast8* and *ecYeast8*, we benchmarked our guideline $\tau = \frac{d^2}{6}$, a smaller thinning constant $\tau = \frac{d^2}{36}$, and a larger thinning constant d^2 . This experimental setup allowed us to test, whether our guideline actually finds the (potentially wide) peak ESS/t . After checking for convergence, we measured the ESS/t , see Fig 5.

For *Yeast8*, setting τ according to our guideline clearly produced the best performance. For *ecYeast8* a six times smaller thinning constant was slightly more efficient (around 4%). However, choosing a six times larger thinning constant than our guideline decreased performance substantially. Because the ESS/t as a function of τ is not multimodal, we can conclude, that the 48 times larger τ , suggested previously [42], performs worse. The estimated time ratios for *Yeast8* (957) and *ecYeast8* (2090) let us conclude that very small thinning constants, such as $\tau = 1$, are inefficient, because too much time would be spent on the back-transform. From these two test-cases, it is clear, that thinning should not be too small nor too large and that our guideline leads to practical performance.

For testing our guideline with the largest validation model, *Recon3D*, we needed to minimize the computational work even further, to achieve results in a reasonable time. We selected two thinning constants to test, namely our guideline, see Eq (14), and previous advice ($\tau = 8 \cdot d^2$) [42]. By benchmarking these two thinning constants, we test, whether our guideline leads to an improvement over existing advice.

For the previous advice ($\tau = 8 \cdot d^2 = 8 \cdot 4,861^2 = 189,034,568$), we observed that CHRR requires 1.1 h to produce and store one sample per chain. Because it is computationally infeasible to run $\tau = 8 \cdot 4,861^2$ to convergence, i.e., to obtain a representative number of samples, we estimate an upper bound on the efficiency of this thinning constant. Assuming optimistically that

each of these samples is uncorrelated, i.e., $ESS = \#stored\ samples$, we project that it takes approximately 110 h until convergence, because both parts of the split chain are required to reach an ESS of 50 [55]. For four parallel and independent chains, as in our benchmarks above and as advised [55], this corresponds, in the best of all cases, to an ESS/t of at most $\frac{4}{1.1h} = 0.001\ s^{-1}$. Using the newly proposed guideline in Eq (14), CHRR with a thinning factor of $\approx 3,938,220$, it takes 27 h to generate 1,000 samples per chain. Using four parallel chains, this setup results in an ESS of 726 with an $\hat{R}$ of ≈ 1.01 . Therewith, our guideline yielded an $ESS/t = 0.0075\ s^{-1}$, which is 7.5 times more efficient than the overly optimistic upper bound we found for the previously advised rule, when both methods use four chains. In summary, the new guideline is not only

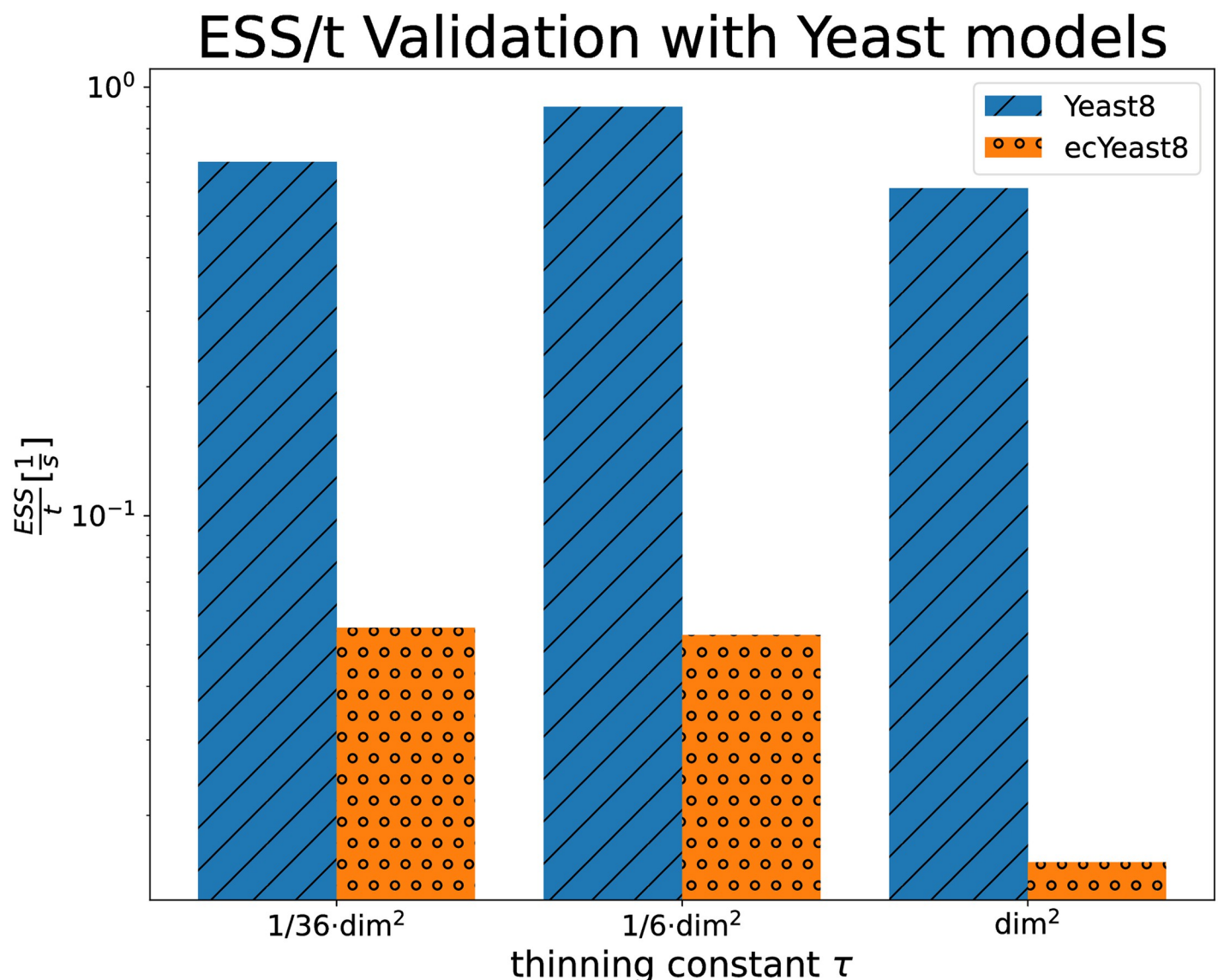


Fig 5. Measured ESS/t for a range of thinning constants τ for the 1,108 dimensional *Yeast8* (blue) and the 3,419 dimensional *ecYeast8* models. For *Yeast8*, the guideline $\hat{\tau} = d^2/6$ produced the highest performance, while for *ecYeast8* $\hat{\tau} = d^2/6$ was measured to be around 4% more efficient $ESS/t = 0.0547\ s^{-1}$ vs. $ESS/t = 0.0527\ s^{-1}$.

<https://doi.org/10.1371/journal.pcbi.1011378.g005>

more efficient, but it also produces a converged set of samples for storage in a fraction of the time. By using three validation models, we have empirically demonstrated that our guideline leads to efficient sampling, also for larger GEMs.

4 Conclusion and outlook

CHRR is the leading algorithm for sampling convex polytopes uniformly, having a single adjustable parameter, the thinning constant. In this work, we showed that the choice of the thinning constant has dramatic consequences on sampling efficiencies, which are instrumental for solving high-dimensional UCPS problems. Here, selecting an appropriate thinning constant can make the difference between sampling success and failure, thus, being the key to sample challenging network models to convergence. Especially for contemporary GEM sampling, it is therefore crucial to have a guideline at hand for the selection of the thinning constant that achieves near-optimal performance.

With a range of UCPS problems at hand, we studied the effect of thinning on CHRR sampling performance quantitatively, after explaining the statistical underpinnings of the efficiency metric (ESS/t). From our quantitative benchmarks, we derived simple guidelines for optimal CHRR thinning constant choice for simplices and GEMs, which we validated using three out-of-sample, larger GEMs. Applying these guidelines is not only beneficial for obtaining uniform samples of polytopes faster, but also helps to migrate UCPS towards green computing, as both the CPU time and the storage cost of samples are reduced. Concerning the question of how the computational requirements of CHRR with optimal thinning scale with effective model dimension, our numerical results reveal a quadratic and linear correlation for simplices and GEMs, respectively. Using the methods we have presented in our study, our guideline can be updated as new types of models and data become available.

Benchmarking new UCPS algorithms and comparing their performances with those of leading algorithms, such as CHRR, is important to advance the field and a topic of active research [53, 58]. Recognizing the substantial impact of thinning on the performance of CHRR, we advise performing comparisons with tuned thinning, and to report the used thinning constant, which will help their reproduction.

Replacing the sequential per-sample by a one-shot or batched back-transform (Algorithm 1, L 12) could be more efficient, as long as memory issues are not limiting (storing unthinned samples of large models, such as *Recon3D*, consumes prohibitively much memory). To combat memory bottlenecks, Stein thinning [59], a technique to optimally compress the MCMC output, may be applied before sample back-transformation. However, while this combination of techniques is promising, the implementation is not straightforward. In contrast, fixed-frequency thinning, as discussed in this work, is already implemented in many existing CHRR packages [4, 42, 46] and immediately boosts performance without requiring additional work.

Despite thinning being a common MCMC practice, it is still controversially discussed. Statisticians have pointed out that thinning is often not necessary and that it typically wastes computational resources, unless the cost of using the samples is high [48, 49]. Even then, Geyer argues that a thinning constant of two or three times the problem dimension should be suited in nearly all cases. Consequently, the conventional advice given to the MCMC practitioner is to not thin MCMC outputs, unless memory or post-processing capabilities are practically limiting. By showing that CHRR stands out among MCMC algorithms in that thinning is critical to performance for sampling high-dimensional convex polytopes, such as GEMs, our study encourages researchers to systematically examine conventional MCMC advice in their specific application domain.

Supporting information

S1 Appendix. Supporting information. Overview of benchmark problems, measured sampling efficiencies, convergence diagnostics, and exemplary flux distributions for selected GEMs.
(PDF)

Acknowledgments

The authors thank Axel Theorell, Richard D. Paul and Johannes Seiffarth for fruitful discussions and feedback.

Author Contributions

Conceptualization: Johann F. Jadebeck, Katharina Nöh.

Formal analysis: Johann F. Jadebeck.

Funding acquisition: Wolfgang Wiechert, Katharina Nöh.

Investigation: Johann F. Jadebeck.

Methodology: Johann F. Jadebeck.

Project administration: Katharina Nöh.

Resources: Wolfgang Wiechert.

Software: Johann F. Jadebeck.

Supervision: Katharina Nöh.

Validation: Johann F. Jadebeck.

Writing – original draft: Johann F. Jadebeck, Katharina Nöh.

Writing – review & editing: Johann F. Jadebeck, Wolfgang Wiechert, Katharina Nöh.

References

1. Bordbar A, Monk JM, King ZA, Palsson BO. Constraint-based models predict metabolic and associated cellular functions. *Nature Reviews Genetics*. 2014; 15(2):107–120. <https://doi.org/10.1038/nrg3643> PMID: 24430943
2. Fang X, Lloyd CJ, Palsson BO. Reconstructing organisms in silico: genome-scale models and their emerging applications. *Nature Reviews Microbiology*. 2020; 18(12):731–743. <https://doi.org/10.1038/s41579-020-00440-4> PMID: 32958892
3. Chung CH, Lin DW, Eames A, Chandrasekaran S. Next-generation genome-scale metabolic modeling through integration of regulatory mechanisms. *Metabolites*. 2021; 11(9):606. <https://doi.org/10.3390/metabo11090606> PMID: 34564422
4. Herrmann HA, Dyson BC, Vass L, Johnson GN, Schwartz JM. Flux sampling is a powerful tool to study metabolism under changing environmental conditions. *npj Systems Biology and Applications*. 2019; 5(1):32. <https://doi.org/10.1038/s41540-019-0109-0> PMID: 31482008
5. Long MR, Ong WK, Reed JL. Computational methods in metabolic engineering for strain design. *Current Opinion in Biotechnology*. 2015; 34:135–141. <https://doi.org/10.1016/j.copbio.2014.12.019> PMID: 25576846
6. Chen Y, Nielsen J. Mathematical modeling of proteome constraints within metabolism. *Current Opinion in Systems Biology*. 2021; 25:50–56. <https://doi.org/10.1016/j.coisb.2021.03.003>
7. Øyås O, Stelling J. Genome-scale metabolic networks in time and space. *Current Opinion in Systems Biology*. 2018; 8:51–58. <https://doi.org/10.1016/j.coisb.2017.12.003>

8. Robinson JL, Kocabaş P, Wang H, Cholley PE, Cook D, Nilsson A, et al. An atlas of human metabolism. *Science Signaling*. 2020; 13(624):eaaz1482. <https://doi.org/10.1126/scisignal.aaz1482> PMID: 32209698
9. Machado D, Andrejev S, Tramontano M, Patil KR. Fast automated reconstruction of genome-scale metabolic models for microbial species and communities. *Nucleic Acids Research*. 2018; 46(15):7542–7553. <https://doi.org/10.1093/nar/gky537> PMID: 30192979
10. Heinken A, Hertel J, Acharya G, Ravcheev DA, Nyga M, Okpala OE, et al. Genome-scale metabolic reconstruction of 7,302 human microorganisms for personalized medicine. *Nature Biotechnology*. 2023; <https://doi.org/10.1038/s41587-022-01628-0> PMID: 36658342
11. Zorrilla F, Buric F, Patil KR, Zeleznik A. metaGEM: reconstruction of genome scale metabolic models directly from metagenomes. *Nucleic Acids Research*. 2021; 49(21):e126–e126. <https://doi.org/10.1093/nar/gkab815> PMID: 34614189
12. Ye C, Wei X, Shi T, Sun X, Xu N, Gao C, et al. Genome-scale metabolic network models: from first-generation to next-generation. *Applied Microbiology and Biotechnology*. 2022; 106(13–16):4907–4920. <https://doi.org/10.1007/s00253-022-12066-y> PMID: 35829788
13. King ZA, Lu J, Dräger A, Miller P, Federowicz S, Lerman JA, et al. BiGG Models: A platform for integrating, standardizing and sharing genome-scale models. *Nucleic Acids Research*. 2015; 44(D1):D515–D522. <https://doi.org/10.1093/nar/gkv1049> PMID: 26476456
14. Norsigian CJ, Pusalta N, McConn JL, Yurkovich JT, Dräger A, Palsson BO, et al. BiGG Models 2020: multi-strain genome-scale models and expansion across the phylogenetic tree. *Nucleic Acids Research*. 2019; 48(D1):D402–D406. <https://doi.org/10.1093/nar/gkz1054> PMID: 31696234
15. Brunk E, Sahoo S, Zielinski DC, Altunkaya A, Dräger A, Mih N, et al. *Recon3D* enables a three-dimensional view of gene variation in human metabolism. *Nature Biotechnology*. 2018; 36(3):272–281. <https://doi.org/10.1038/nbt.4072> PMID: 29457794
16. Colarusso AV, Goodchild-Michelman I, Rayle M, Zomorodi AR. Computational modeling of metabolism in microbial communities on a genome-scale. *Current Opinion in Systems Biology*. 2021; 26:46–57. <https://doi.org/10.1016/j.coisb.2021.04.001>
17. Thiele I, Sahoo S, Heinken A, Hertel J, Heirendt L, Aurich MK, et al. Personalized whole-body models integrate metabolism, physiology, and the gut microbiome. *Molecular Systems Biology*. 2020; 16(5):e8982. <https://doi.org/10.15252/msb.20198982> PMID: 32463598
18. Heinken A, Basile A, Thiele I. Advances in constraint-based modelling of microbial communities. *Current Opinion in Systems Biology*. 2021; 27:100346. <https://doi.org/10.1016/j.coisb.2021.05.007>
19. Kim J, Reed JL, Maravelias CT. Large-scale bi-level strain design approaches and mixed-integer programming solution techniques. *PLOS ONE*. 2011; 6(9):e24162. <https://doi.org/10.1371/journal.pone.0024162> PMID: 21949695
20. Schellenberger J, Zielinski DC, Choi W, Madireddi S, Portnoy V, Scott DA, et al. Predicting outcomes of steady-state ^{13}C isotope tracing experiments using Monte Carlo sampling. *BMC Systems Biology*. 2012; 6(1):9. <https://doi.org/10.1186/1752-0509-6-9> PMID: 22289253
21. Domenzain I, Sánchez B, Anton M, Kerkhoven EJ, Millán-Oropeza A, Henry C, et al. Reconstruction of a catalogue of genome-scale metabolic models with enzymatic constraints using GECKO 2.0. *Nature Communications*. 2022; 13(1):3766. <https://doi.org/10.1038/s41467-022-31421-1> PMID: 35773252
22. De Martino D, Mori M, Parisi V. Uniform sampling of steady states in metabolic networks: heterogeneous scales and rounding. *PLOS ONE*. 2015; 10(4):1–14. <https://doi.org/10.1371/journal.pone.0122670> PMID: 25849140
23. Theorell A, Stelling J. Metabolic networks, microbial consortia, and analogies to smart grids. *Proceedings of the IEEE*. 2022; 110(5):541–556. <https://doi.org/10.1109/JPROC.2022.3158396>
24. Ciomek K, Kadziński M. Polyrun: A Java library for sampling from the bounded convex polytopes. *SoftwareX*. 2021; 13:100659. <https://doi.org/10.1016/j.softx.2021.100659>
25. Drouineau H, Planque B, Mullon C. RCaN: a software for chance and necessity modelling. *bioRxiv*. 2021; 2021.06.09.447734.
26. Chalkis A, Christoforou E, Emiris IZ, Dalamagas T. Modeling asset allocations and a new portfolio performance score. *Digital Finance*. 2021; 3(3):373–373. <https://doi.org/10.1007/s42521-021-00042-6>
27. Lubini M, Coles J. A sampling strategy for high-dimensional spaces applied to free-form gravitational lensing. *Monthly Notices of the Royal Astronomical Society*. 2012; 425(4):3077–3084. <https://doi.org/10.1111/j.1365-2966.2012.21673.x>
28. Leake J, McSwiggen CS, Vishnoi NK. Sampling matrices from Harish-Chandra-Itzykson-Zuber densities with applications to quantum inference and differential privacy. *arXiv*. 2020. Available from: <https://arxiv.org/abs/2011.05417>.

29. Schellenberger J, Palsson BØ. Use of randomized sampling for analysis of metabolic networks. *Journal of Biological Chemistry*. 2009; 284(9):5457–61. <https://doi.org/10.1074/jbc.R800048200> PMID: 18940807
30. Loghmani SB, Veith N, Sahle S, Bergmann FT, Olivier BG, Kummer U. Inspecting the solution space of genome-scale metabolic models. *Metabolites*. 2022; 12(1):43. <https://doi.org/10.3390/metabo12010043> PMID: 35050165
31. Machado D, Herrgård MJ, Rocha I. Stoichiometric representation of gene–protein–reaction associations leverages constraint-based analysis from reaction to gene-level phenotype prediction. *PLOS Computational Biology*. 2016; 12(10):1–24. <https://doi.org/10.1371/journal.pcbi.1005140> PMID: 27711110
32. Heinonen M, Osmala M, Mannerström H, Wallenius J, Kaski S, Rousu J, et al. Bayesian metabolic flux analysis reveals intracellular flux couplings. *Bioinformatics*. 2019; 35(14):i548–i557. <https://doi.org/10.1093/bioinformatics/btz315> PMID: 31510676
33. Beyß M, Parra-Peña VD, Ramirez-Malule H, Nöh K. Robustifying experimental tracer design for ¹³C-metabolic flux analysis. *Frontiers in Bioengineering and Biotechnology*. 2021; 9. <https://doi.org/10.3389/fbioe.2021.685323> PMID: 34239861
34. Bernstein DB, Sulheim S, Almaas E, Segrè D. Addressing uncertainty in genome-scale metabolic model reconstruction and analysis. *Genome Biology*. 2021; 22(1):64. <https://doi.org/10.1186/s13059-021-02289-z> PMID: 33602294
35. Dinh HV, Sarkar D, Maranas CD. Quantifying the propagation of parametric uncertainty on flux balance analysis. *Metabolic Engineering*. 2022; 69:26–39. <https://doi.org/10.1016/j.ymben.2021.10.012> PMID: 34718140
36. Gelman A, Carlin JB, Stern S H, Dunson B D, et al. *Bayesian Data Analysis*. 3rd ed. Chapman and Hall/CRC; 2013.
37. Wiback SJ, Famili I, Greenberg HJ, Palsson BØ. Monte Carlo sampling can be used to determine the size and shape of the steady-state flux space. *Journal of Theoretical Biology*. 2004; 228(4):437–447. <https://doi.org/10.1016/j.jtbi.2004.02.006> PMID: 15178193
38. Smith RL. Efficient Monte Carlo procedures for generating points uniformly distributed over bounded regions. *Operations Research*. 1984; 32(6):1296–1308. <https://doi.org/10.1287/opre.32.6.1296>
39. Kaufman DE, Smith RL. Direction choice for accelerated convergence in Hit-And-Run sampling. *Operations Research*. 1998; 46(1):84–95. <https://doi.org/10.1287/opre.46.1.84>
40. Lovász L. Hit-and-Run mixes fast. *Mathematical Programming*. 1999; 86(3):443–461. <https://doi.org/10.1007/s101070050099>
41. Robert CP, Casella G. *Monte Carlo statistical methods*. Springer Texts in Statistics. Springer New York; 2004. Available from: <http://link.springer.com/10.1007/978-1-4757-4145-2>.
42. Haraldsdóttir HS, Cousins B, Thiele I, Fleming RMT, Vempala S. CHRR: coordinate Hit-and-Run with rounding for uniform sampling of constraint-based models. *Bioinformatics*. 2017; 33(11):1741–1743. <https://doi.org/10.1093/bioinformatics/btx052> PMID: 28158334
43. Fallahi S, Skaug HJ, Alendal G. A comparison of Monte Carlo sampling methods for metabolic network models. *PLOS ONE*. 2020; 15(7):1–24. <https://doi.org/10.1371/journal.pone.0235393> PMID: 32609776
44. Theorell A, Jadebeck JF, Nöh K, Stelling J. PolyRound: polytope rounding for random sampling in metabolic networks. *Bioinformatics*. 2021; 38(2):566–567. <https://doi.org/10.1093/bioinformatics/btab552>
45. Heirendt L, Arreckx S, Pfau T, Mendoza SN, Richelle A, Heinken A, et al. Creation and analysis of biochemical constraint-based models using the COBRA Toolbox v.3.0. *Nature Protocols*. 2019; 14(3):639–702. <https://doi.org/10.1038/s41596-018-0098-2> PMID: 30787451
46. Jadebeck JF, Theorell A, Leweke S, Nöh K. HOPS: high-performance library for (non-)uniform sampling of convex-constrained models. *Bioinformatics*. 2020; 37(12):1776–1777. <https://doi.org/10.1093/bioinformatics/btaa872>
47. Gollub MG, Kaltenbach HM, Stelling J. Probabilistic thermodynamic analysis of metabolic networks. *Bioinformatics*. 2021; 37(18):2938–2945. <https://doi.org/10.1093/bioinformatics/btab194> PMID: 33755125
48. Geyer CJ. Practical Markov Chain Monte Carlo. *Statistical Science*. 1992; 7(4):473–483. <https://doi.org/10.1214/ss/1177011147>
49. Link WA, Eaton MJ. On thinning of chains in MCMC. *Methods in Ecology and Evolution*. 2012; 3(1):112–115. <https://doi.org/10.1111/j.2041-210X.2011.00131.x>
50. Kim WJ, Kim HU, Lee SY. Current state and applications of microbial genome-scale metabolic models. *Current Opinion in Systems Biology*. 2017; 2:10–18. <https://doi.org/10.1016/j.coisb.2017.03.001>

51. Wiechert W, Möllney M, Petersen S, de Graaf AA. A universal framework for ^{13}C metabolic flux analysis. *Metabolic Engineering*. 2001; 3(3):265–283. <https://doi.org/10.1006/mben.2001.0188> PMID: 11461148
52. Cousins B., Vempala S. A practical volume algorithm. *Mathematical Programming Computation*. 2016; 8(2):133–160. <https://doi.org/10.1007/s12532-015-0097-z>
53. Chalkis A, Fisikopoulos V, Papachristou M, Tsigaridas EP. Truncated log-concave sampling for convex bodies with Reflective Hamiltonian Monte Carlo. *ACM Transactions on Mathematical Software*. 2023; <https://doi.org/10.1145/3589505>
54. Zhang Y, Gao L. On numerical solution of the Maximum Volume Ellipsoid problem. *SIAM Journal on Optimization*. 2003; 14(1):53–76. <https://doi.org/10.1137/S1052623401397230>
55. Vehtari A, Gelman A, Simpson D, Carpenter B, Bürkner PC. Rank-Normalization, folding, and localization: an improved $\hat{R}$ for assessing convergence of MCMC (with discussion). *Bayesian Anal*. 2021; 16(2):667–718. <https://doi.org/10.1214/20-BA1221>
56. Hucka M, Finney A, Sauro HM, Bolouri H, Doyle JC, Kitano H, et al. The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics*. 2003; 19(4):524–531. <https://doi.org/10.1093/bioinformatics/btg015> PMID: 12611808
57. Kumar R, Carroll C, Hartikainen A, Martin O. ArviZ a unified library for exploratory analysis of Bayesian models in Python. *Journal of Open Source Software*. 2019; 4(33):1143. <https://doi.org/10.21105/joss.01143>
58. Kook Y, Lee YT, Shen R, Vempala SS. Sampling with Riemannian Hamiltonian Monte Carlo in a constrained space. *arXiv*. 2022. Available from: <https://arxiv.org/abs/2202.01908>.
59. Riabiz M, Chen WY, Cockayne J, Swietach P, Niederer SA, Mackey L, et al. Optimal thinning of MCMC output. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*. 2022. <https://doi.org/10.1111/rssb.12503>