Hamiltonian Monte Carlo with strict convergence criteria reduces run-to-run variability in forensic DNA mixture deconvolution

Mateusz Susik[1,2,*], Holger Schönborn[3], Ivo F. Sbalzarini[2,4,5]

1 Biotype GmbH, Dresden, 01109, Germany
2 Technische Universität Dresden, Faculty of Computer Science, Dresden, 01187, Germany
3 qualitype GmbH, Dresden, 01109, Germany
4 Max Planck Institute of Molecular Cell Biology and Genetics, Dresden, 01307, Germany
5 Center for Systems Biology Dresden, Dresden, 01307, Germany

*** m.susik [at] biotype.de**

Abstract

Motivation: Analysis of mixed DNA profiles is one of the common tasks for the forensic practitioners. Due to the complexity of the data the analysis is often performed with Bayesian probabilistic genotyping algorithms. These trade off the precision of the results against the execution time. When the default settings are used, as large as a 10-fold change in likelihood ratios (LR) might be observed when the software is run twice on the same case. In the past this variance has been attributed to the stochasticity of Markov chain Monte Carlo (MCMC) algorithm. As likelihood ratios translate directly to the strength of evidence in a trial, the users of the algorithms expect that the resulting likelihood ratios will be near to identical between runs.
Results: We present an implementation of Hamiltonian Monte Carlo that achieves this goal. Our method reduces the standard deviation of the $\log_{10}$ likelihood ratio around 10 times in case of a previously analysed mixture. We show that choice of the used convergence metric influences the precision. The method classifies correctly contributors and non-contributors in case of selected MIX05, MIX13, and ProvedIt mixtures closely reproducing previously published results. Our software implementation, however, is the first one to benefit from GPU acceleration. The inference process is quick and on a single-replicate mixtures we considered it takes less than 7 minutes for 3 contributor mixtures, less than 35 minutes for 4 contributor mixtures and less than an hour for 5 contributor mixtures with one known contributor.

1 Introduction

Investigators present at a crime scene identify and collect the available physical evidence. As a part of this evidence, DNA samples containing material from multiple contributors (i.e. *mixed DNA samples*) are often collected. The corresponding short tandem repeat data is affected by stochastic events such as severe peak height imbalance, drop-outs, and drop-ins [3], especially in case of the low-template samples. Manual analysis of the data (available in the form of electropherograms (EPG)) is unreliable and biased [37]. Therefore, laboratories should rely on validated statistical software developed to solve the task of DNA mixture deconvolution [22].

The recommended approach [1] for reporting the results of mixture analysis is the likelihood ratio (LR):

$$LR = \frac{P(V|H_p)}{P(V|H_d)} = \frac{\sum_j P(V|S_j)P(S_j|H_p)}{\sum_j P(V|S_j)P(S_j|H_d)}\,, \tag{1}$$

where V is the observed EPG, S_j represents a genotype set - a list of couples denoting the allele designations of contributors. The summation is over all possible genotype sets. H_p and H_d are the hypotheses of the prosecutor and the defendant respectively. A hypothesis assumes inclusion of the given contributors (suspect, victim, etc.) in the mixture and the allele frequencies of the populations the contributors assumedly originate from. Usually, the difference between the hypothesis of the prosecutor and the hypothesis of the defendant is the inclusion of the suspect in the former. In the setting we consider, the number of considered contributors is set by the user. $P(S_j|H_n)$ can be calculated based on the frequencies of the alleles in the population of interest for any hypothesis H [8]. The term *probabilistic genotyping* (PG) is often used to describe the procedures used to calculate LR.

In order to estimate $P(V|S_j)$, assumptions about the underlying data-generating process are made. These assumptions lead to a probabilistic model $P(V|M, S_j)$ with latent variables M. Models where probability is estimated based on the heights of the peaks in the mixture are termed "continuous". Two main algorithmic approaches are used: finding the most likely set of parameters with maximum likelihood estimation [7] or estimating the posterior distribution [31, 34]:

$$P(V|S_j) = \int_M P(V|m, S_j) f(m)\,\mathrm{d}m\,. \tag{2}$$

In the ideal scenario, LR is independent of any choices made by the analyst and any random factors. In practice, however, LR depends on the variation coming from the samples on the crime scene, stochastic events occurring during the amplification process, the allele frequency sampling, settings of data processing software[1], hyperparameterisation of PG software, etc. [30]. Whenever possible, the factors causing the variability should be quantified or limited.

Here, we focus on reducing the run-to-run variability of MCMC-based PG software. We define precision of a solution as the variance between the results of runs with identical hyperparameterization on the same replicates and hypotheses. Practitioners [30] have noted that precision is a factor that has to be considered in addition to the accuracy of the PG systems. Authors of the algorithms state that:

> "The argument is that the existence of variability *[of algorithm results - our note]* raises doubts about whether any of the results should be accepted." [25]

Courts are often unaware of the variability, as expert witnesses usually report a single number output by PG software [23]. The issue might be more severe if the verbal scale of reporting LRs is used. The standard scales of LR [2, 39] coming from evidence interpretation define LR between 100 and 1000 as "moderately strong support" and LR between 1000 and 10000 as "strong support". Let us assume that we use a PG system, which, for the given case, outputs results from a normal distribution: $\log_{10}\mathrm{LR} \sim \mathcal{N}(2.3, 0.5)$. A single run of the software would give "strong support" in $\approx 73\%$ of cases. A technique that reports confidence intervals [8, 10], if correctly applied, would provide the conservative answer of "moderately strong support". Now, let us assume we have access to a more precise solution that always reports

[1] e.g. GeneMapper™

$\log_{10} LR = 2.25$. It would always provide "strong support" on the verbal scale. A scenario where the less precise solution overestimates the evidence is also possible, for example in case when no confidence interval is estimated. This highlights the importance of the low variability of the results.

1.1 Precision of MCMC solutions

There exist studies that quantify the precision of the available commercial solutions [31, 34]. Bauer et al. [5] show a standard deviation of LR between runs larger than 10^4 on a three-contributor example (sample '3-2') when TrueAllele® was used. STRmix™ showed examples with a 10-fold difference between two runs [9, 12]. Some papers argue that a remedy to decreased precision is an increase of MCMC iterations [12, 32], but this involves additional computational cost.

We show here that precision depends on the diagnostic metrics used in the algorithm. The users of MCMC should use convergence metrics to check if the quality of the results is sufficient for further analysis [24]. Although true convergence can not be guaranteed as long as we do not know the true posterior distribution, practitioners usually check the stability of samples. The most popular solution is the univariate *Gelman Rubin diagnostic* (GR) [13, 24], which compares pooled and within-chain variances of samples. For given model parameters, this statistic outputs a value close to 1 when the samples from different chains result in similar estimates for the marginal distribution. We hereinafter use the term "convergence" to denote approximate convergence with regards to the chosen metric.

Some solutions available on the market provide users with convergence diagnostics. STRmix™ [34] calculates a ratio of pooled and within-chain variances of the likelihoods of the model (source: personal communication). GenoProof Mixture [26] uses the maximum out of univariate GRs over the continuous parameters. By default both solutions use a predefined constant number of post burn-in iterations and then report the value of the diagnostic to the user. If the desired threshold (by default 1.2 in STRmix™, 1.05 in GenoProof Mixture) of the convergence statistic has not been achieved, the software offers an option to run additional iterations. Both of these models operate on a mix of continuous and discrete (genotype sets) domains.

However, GR is not a metric that can be used to monitor convergence in discrete dimensions. It is therefore possible that convergence is deduced purely from the continuous parameters, while the genotype sets' distribution have not converged, resulting in low precision of the solution.

The default GR value should be close to one. The authors of the metric state:

> "The condition of GR near 1 depends on the problem at hand; for most examples, values below 1.1 are acceptable, but for a final analysis in a critical problem, a higher level of precision may be required." [24]

Providing evidence in the court should be considered a critical problem and the consequences of wrong or doubtful answers are significant [28]. Other scientists researching convergence metrics note:

> "We argue that a cutoff of GR ≤ 1.1 is much too high to yield reasonable estimates of target quantities." [38]

Addressing this debate in the field of Bayesian inference, we developed a technique that achieves better convergence and, as a consequence, better precision.

1.2 Importance of execution time

We avoid the issue of assessing convergence of genotype sets by marginalising them out. Then, however, a naive implementation of MCMC is not feasible anymore, as the computations could span over many days and the laboratories would not be able to conduct computations with hypotheses that include 4 or 5 unknown contributors. A computationally efficient solution is needed since:

- Users might want to run software with different hyperparameterizations to check robustness of the results (different number of contributors, different analytical thresholds, different priors, etc.).
- Users might want to check the precision of the results.
- The laboratories are often working under time pressure (e.g. cases that attract great media attention, laws of maximum 24/48 hour detention time without charges) .

Aside from the efficiency of the implementation there are multiple factors that influence the execution time, including the number of contributors, the number of alleles per locus, the techniques used to limit the number of considered genotype sets, the convergence criteria, choice of the optimisation problem (maximum likelihood vs Bayesian inference), the specification of the model, the priors used, etc.

In general, there is a trade-off between the precision of the solution and runtime. Quicker runtimes can be easily achieved by running a smaller number of iterations, at the expense of precision. Achieving perfect precision is theoretically possible, if runtime is unimportant, by integrating over the latent variables (see Equation 2).

We improve upon the precision of existing solutions by introducing a novel marginalised model and a more efficient HMC sampler. Thanks to an efficient solution space search, we are able to set a stricter GR threshold of 1.05 with reasonable runtime. We show the effect of the chosen convergence statistic on the variance of the resulting LR. We show how to efficiently deal with the combinatorial growth in the number of genotype sets with increasing number of contributors, and we provide the first GPU-enabled implementation of a DNA mixture algorithm.

2 Model

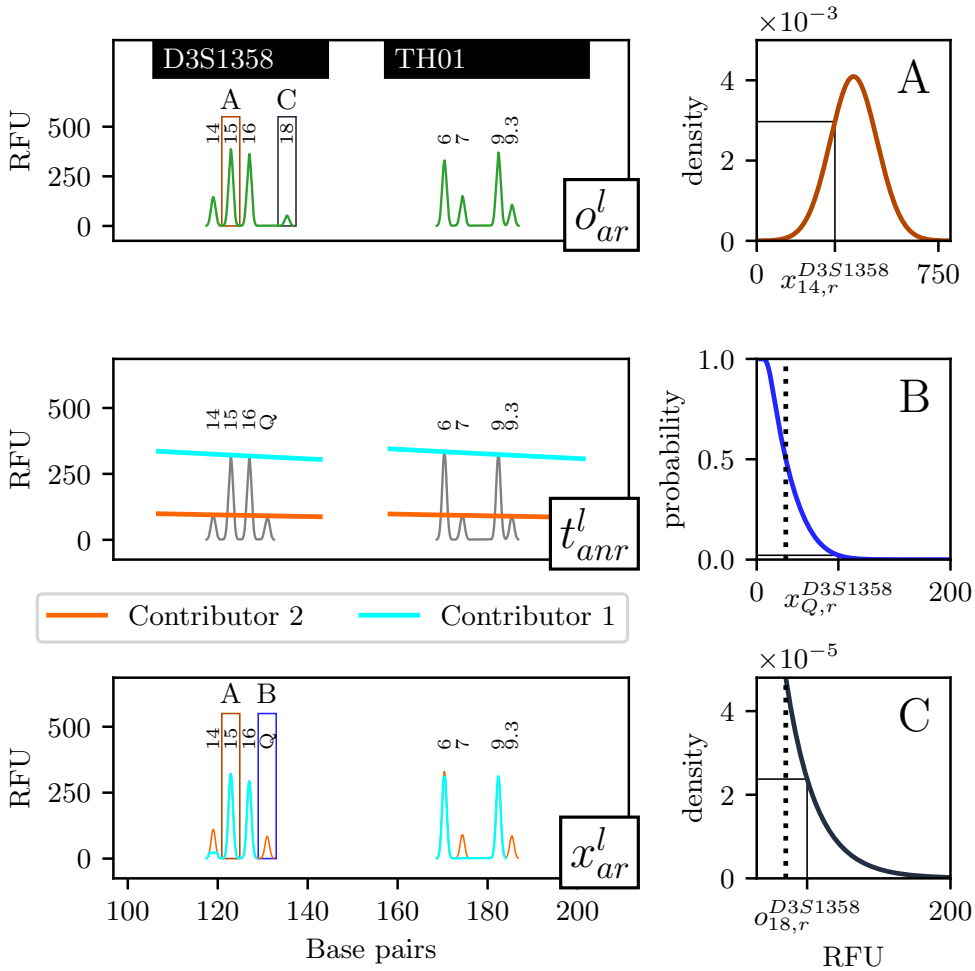
We base our probabilistic model on the work by Taylor et al. [34] due to the large number of studies that describe and evaluate it (e.g. [7, 16, 19, 34, 36]).

The main assumption in this model is the log-normal distribution of the ratio of the observed peak heights from EPG to the peak heights calculated by the generative model (called hereinafter "expected" peak heights). The generative model consists of few steps: First, the expected contributions are calculated for each genotype set from the considered set of parameters. Then the stutter models are applied to create expected peaks. The next step is to calculate the standard deviation $\sigma^l_{ar}(\mathbf{x})$ of the log-normal distribution. Finally, the likelihood of the observed data given the parameters is calculated as the product of likelihoods of all the peaks. The model handles stochastic dropout and drop-in events. These steps are visualised in Figure 1.

To provide a mathematical formulation of the model we denote:

- O^l_{ar} - random variable of an observed peak height at locus l, allele a, replicate r;
- x^l_{ar} - expected peak height;
- f_X - the probability density function of a random variable X;

Figure 1. Visualisation of the probabilistic model. Top left figure: the observed peaks O^l_{ar} at the selected two loci from the green channel of the analysed mixture. Middle left: We analyse a single genotype set: [(15,16),(14,Q)] for D3S1358 locus and [(6,9),(7,9,3)] for TH01. The curves depict values of the expected contributions for all alleles t^l_{anr}. The function is decreasing within a locus due to the effect of decay modelling. TH01 locus was modelled with a larger amplification efficiency parameter than D3S1358. Bottom left: Expected peaks x^l_{ar} are created by applying stutter ratios to t^l_{anr}. In this example we employ only backward stutter for illustration purposes. Composed peaks (i.e. those that consist of both allelic and stutter contribution) are 14 and 15 at D3S1358 and 6 at TH01. Top right: density of the log-normal model for peak 15 at D3S1358 as a function of the expected peak height. Middle right: dropout formula as a function of expected peak height. Catch-all allele "Q" is used in proposed genotype sets to denote dropped out peaks. Bottom right: drop-in formula as a function of observed peak height. Peak 18 at D3S1358 is a drop-in as it is observed and not expected under the considered genotype set. The dotted lines denote the analytical threshold.



- Q - a "catch-all" dropout allele [34];
- h_r^l - the analytical threshold;
- $\mathcal{N}(0, s)$ - a normal distribution with mean 0 and standard deviation s;
- (M, N) - alleles of the contributors at a single locus, e.g $(12, 13)$;

We assess the posterior probability:

$$P(M|V) = \frac{P(V|M)P(M)}{P(V)} = \frac{P(M)\sum_j P(V|M, S_j)P(S_j|M)}{P(V)}\,. \tag{3}$$

As this analysis is independent of the hypothesis, we do not favour any genotypes and treat them as equally likely. In Bayesian inference evidence is usually neglected as it is too expensive to compute and constant w.r.t. model parameters. We thus obtain the unnormalised posterior:

$$P(M|V) \propto P(M) \sum_j P(V|M, S_j)\,, \tag{4}$$

which we use for estimating $P(V|S_j)$ (see Equation 2). We assume peak heights to be conditionally independent given S_j and M, and alleles of a contributor in different loci to be independent from each other. The resulting model is a function E_{ar}^l of M and S_j. The likelihood of the observed EPG given parameters M and a genotype set S_j is then expressed as:

$$\sum_j P(V|M, S_j)P(M) = \int_M P(m) \sum_j \prod_l \prod_r \Bigg(\Bigg(\prod_{b \in B} P\left(O_{br}^l \mid x_{br}^l(m, S_j), O_{br}^l > 0\right) \Bigg) \prod_{d \in D} P\left(\text{dropout} \mid x_{dr}^l(m, S_j)\right) \Bigg) \mathrm{d}m\,. \tag{5}$$

The inner multiplications are performed over the set B of observed peaks and the set D of hypothetical dropped-out peaks. In the following, we abbreviate the notation for $x_{ar}^l(m, S_j)$ to x_{ar}^l. This model formulates separately the relative likelihood of observed peaks and the probabilities of dropout events.

2.1 Observed peaks

In case a peak is observed (i.e. $O_{ar}^l > 0$) our model considers a mixture distribution. The mixture combines sub-models for peaks that are expected (f_Z) and drop-in events $\left(f_{(O_{ar}^l|x_{ar}^l=0, O_{ar}^l>0)}\right)$:

$$P\left(O_{ar}^l \mid x_{ar}^l, O_{ar}^l > 0\right) \quad \propto \quad \mathbb{1}_{(x_{ar}^l>0)} f_Z \;+\; \mathbb{1}_{(x_{ar}^l=0)} d_{\text{rate}} f_{(O_{ar}^l|x_{ar}^l=0, O_{ar}^l>0)}\,. \tag{6}$$

For drop-in events we use the same model as the one introduced by Euroformix [7]:

$$x_{ar}^l = 0, O_{ar}^l > 0 \implies O_{ar}^l - h_r^l \sim Exp(\lambda)\,. \tag{7}$$

Two hyperparameters based on the level of baseline noise in negative controls are required: the drop-in rate d_{rate} and a hyperparameter λ governing the shape of the drop-in probability distribution.

In the case an observed peak is expected, we assume a log-normal distribution following the model presented in [34]:

$$Z = \ln \frac{O^l_{ar}}{x^l_{ar}} \sim N(0, \sigma^l_{ar}(\mathbf{x}))\,. \tag{8}$$

The standard deviation of this distribution depends on the components of the expected peaks and their heights. We consider separate standard deviation parameters for stutter and allelic peak heights:

$$\sigma^l_{ar}(\mathbf{x}) = \frac{1}{x^l_{ar}} \sum_{\psi \in \Psi(a)} \frac{c_\psi x^l_{\psi ar}}{\sqrt{\chi^l_{\psi r}}}\,. \tag{9}$$

We define $\psi \in \Psi(a) = \{a+2, a+1, a, a-1\}$. This means that for a single allele a we consider contributions from the allelic peak a, the backward stutter from the $a+1$ peak, the forward stutter from the $a-1$ peak, and the double backward stutter from the $a+2$ peak.

We use one parameter for allelic peak standard deviation ($c_\psi = c_p$ when $\psi = a$) and one for stutter peak standard deviation ($c_\psi = c_s$ when $\psi \neq a$), but additional parameters for different types of stutter could be introduced without significant changes to the model. The expected peaks heights then are:

$$x^l_{\psi ar} = \sum_n \rho^l_{(2a-\psi)ar} t^l_{\psi nr} \tag{10}$$

$$x^l_{ar} = \sum_{\psi \in \Psi(a)} x^l_{\psi ar}\,, \tag{11}$$

The sum in Formula 10 is performed over the assumed contributors. $\chi^l_{\psi ar}$ models the fact that peak variance is inversely proportional to peak height [14]:

$$\chi^l_{tr} = \frac{1000}{x^l_{tr}+1} + x^l_{tr}\,. \tag{12}$$

Equation 10 includes the normalised stutter ratios (ρ) and product contribution from a contributor ($t^l_{\psi nr}$). To obtain ρ, the unnormalised stutter ratios π are deduced from unambiguous profiles. They are modelled with linear regressions based on allele designation or longest uninterrupted sequences [34]. Normalisation is required, since multiple types of stutter are considered at the same time. The normalisation is calculated as:

$$\rho^l_{\psi ar} = \begin{cases} \left(1 + \pi^l_{(a-2)ar} + \pi^l_{(a-1)ar} + \pi^l_{(a+1)ar}\right)^{-1} & \text{, if } \psi = a \\ \pi^l_{\psi ar}\rho^l_{aar} & \text{, otherwise}\,. \end{cases} \tag{13}$$

The contribution at a selected allele is defined as:

$$t^l_{anr} = \zeta^l_{an}\alpha^l_r w_n e^{0.001 d_{rn}(m^l_a - \overline{m})} t'_r\,, \tag{14}$$

where m^l_a is the molecular weight of an allele, $\overline{m}$ is the average of the largest and smallest observed molecular weights on the EPG within the called peaks. ζ^l_{an} indicates if the nth contributor's genotype includes ath allele in the lth locus and is equal to 2 in case of a homozygote, 1 in case of heterozygote and 0 if the allele is not present. The scalar α^l_r are the locus-specific amplification efficiencies (LSAE), w_n are the weights of the contributors that sum up to 1, d_{rn} are the decay parameters, and t'_r is the total allelic product expressed in relative fluorescent units (RFU).

2.2 Dropout events

In case of a dropout, the peak is unobserved because it is below the analytical threshold. This corresponds directly to:

$$P\left(\text{dropout} \mid E_{dr}^{l}\right) = P\left(Z \leq \ln \frac{h_r^l}{x_{dr}^l}\right) . \tag{15}$$

2.3 Parameters of the model and priors

We define the prior probability on the parameters as:

$$P(M) = P(\alpha|i)P(i)P(c_s)P(c_p) . \tag{16}$$

Here, $P(\alpha|i)$ is the prior $\ln(\alpha_r^l) \sim \mathcal{N}(0, i)$ that prevents the amplification efficiencies from drifting away from 1 too far. $i^2 \sim Exp(\sigma_\alpha)$, where σ_α is a hyperparameter to be optimised by the laboratory [11]. $P(c_s)$ and $P(c_p)$ are optional priors on the peak height standard deviation parameters, which are also present in STRmix™ [11].

The parameters to be explored by the sampling algorithm are then: c_s, c_p, t'_r, w_n, d_{rn}, α_r^l, i. We reduce the number of parameters in case multiple replicates are performed using the same kit. In such a case, the analysis shares the amplification efficiencies between the replicates. Let us denote l_{all} as the number of amplification efficiency parameters. Assuming different kits are used in the replicates, there are $2 + |r| + |n| + |r \times n| + l_{\text{all}}$ parameters whose values have to be estimated (one of the weights is trivial when all the others are set).

2.4 Other considerations

To provide conservative estimates of LR, we use Balding-Nichols subpopulation correction [4, 8], and we report sub-source LRs [35] unless specified otherwise. We consider dropped-out allelic contributions to be separate peaks (i.e., (Q,Q) is considered heterozygous). The total number of genotype sets is reduced by considering at most one drop-in per locus, using a drop-in cap and removing genotypes where the peaks that would be considered stutter would be abnormally tall w.r.t. the origin (see the maximum stutter ratios in Table 2).

3 Sampling algorithm

To improve the precision of the solution we must be able to guarantee the approximate convergence. It is simpler to achieve it if a modern, adaptive sampler is used. We focus on a method that is successfully used in other fields: Hamiltonian Monte Carlo (HMC) [21]. The difference between HMC and MCMC lies in the way how the proposal is chosen. HMC simulates physical system dynamics instead of choosing a random point from the neighbourhood of the current point. HMC is an algorithm that is very efficient compared to MCMC for posteriors with a complex shape and/or large dimensionality [6]. The proposal is dynamically adapted across iterations. To choose it, HMC calculates the gradient of the log-probability of the model. Thanks to the fact that we marginalise out the genotype sets, our model is differentiable. Instead of changing the value of a single parameter (like currently available MCMC software does) our implementation considers multivariate moves each single iteration.

We rely on the proven automatic differentiation framework and the existing implementation of the HMC algorithm from Tensorflow Probability [20]. Thanks to the capabilities of this library, we can provide both CPU and GPU versions of the

Figure 2. Example of a locus from a mixture.

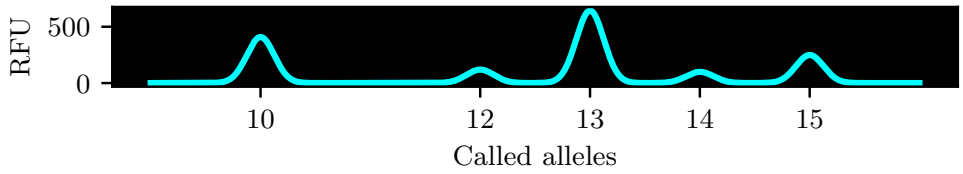


Table 1. Three likely genotype sets and the resulting contributions to peaks for the locus from Figure 2. $C_{n,p}$ denotes the contribution of the n-th contributor from a single copy of p-th allele. $C'_{n,p}$ denotes the contribution from stutter originating from p-th allele. For demonstration purposes only single backward stutter is considered. Duplicated entries are marked with the same colour.

S_j	10	12	13	14	15	Q
(15,Q), (10,13)	$C_{2,10}$	$C'_{2,13}$	$C_{2,13}$	$C'_{1,15}$	$C_{1,15}$	$C_{1,Q}$
(10,Q), (13,15)	$C_{1,10}$	$C'_{2,13}$	$C_{2,13}$	$C'_{2,15}$	$C_{2,15}$	$C_{1,Q}$
(10,13), (12,15)	$C_{1,10}$	$C'_{1,13} + C_{2,12}$	$C_{1,13}$	$C'_{2,15}$	$C_{2,15}$	0

algorithm. However, a naive implementation using this framework would perform below expectations because of the combinatorial growth of the number of possible genotype sets with increasing number of contributors. We therefore introduce an important performance improvement: a deduplication system that can be used with any model that considers all genotypes in a single iteration. Additional improvements are described in Chapter 3 of the Supplementary Material.

For the deduplication system, we consider a single locus l. The expected peak heights x^l_{ar} and the standard deviation $\sigma^l_{ar}(\mathbf{x})$ of the distribution depend on the genotype sets. If we consider the output for a single allelic position, these values depend only on the continuous parameters and the counts of alleles of individual contributors from the selected position and the stutter positions. When multiple genotype sets are considered, the same values are computed multiple times. As an example, consider the locus shown in Figure 2 and the likely genotype sets from Table 1. Computing every expected peak (and the likelihoods of observing the ratios between the observed and expected peaks) for genotype set $\{(10, 13), (12, 15)\}$ is also performed in case of the other genotype sets. In our deduplicated scenario, each such computation is performed only once and the result is reused. In one locus from a 4-contributor mixture we analysed 524 peaks had to be calculated after deduplication. Without deduplication 44473 peaks would have to be calculated. We use the deduplication procedure during both the calculation of the log-probability and the calculation of the HMC gradient.

Algorithms that consider all possible genotype sets per iteration (e.g. our work, Euroformix) are implemented using array programming. Before a run is performed, we precompute the indices where the duplicated entries in the functions will occur and create two data structures: one containing all the information required to calculate the deduplicated expected peak heights and one containing indices for a gather operation that is performed after the log-probabilities of the deduplicated peaks have been calculated. The gather operation unfolds the deduplicated results into a matrix that stores the log-probabilities per replicate, allelic position, chain, and genotype set.

Table 2. The hyperparameters used. SR stands for stutter ratio. The values of some of variance rate and locus specific amplification efficiency (LSAE) standard deviation hyperparameters have been adjusted to our specification of the model (natural logarithm instead of $\log_{10}$, 1/mean for LSAE standard deviation hyperparameter).

	ProvedIt	MIX05, MIX13
Drop-in probability	0.0015	0
Drop-in λ	0.032	N/A
Allele variance shape	5.653	3.57
Allele variance rate	15.7	5.196
Stutter variance shape	1.501	6.97
Stutter variance rate	148.462	9.279
LSAE standard deviation	32.258	33.333
Drop-in cap	$3\ h_r^l$	
Max. observed backward SR	0.3	
Max. observed forward SR	0.15	
Max. observed double backward SR	0.05	
Rare allele frequency	$\frac{2.5}{\text{size of the sampled population}}$	
$F_{ST}(\theta)$	0.01	
Number of chains	4	
Number of burnin steps	1200	
Leapfrog steps per sample	10	
GR stopping threshold	1.05	

4 Experiments

A reliable forensic genotyping solution should not provide false inclusion or false exclusion. Moreover, it should be precise and have low inference runtime. We benchmark these aspects of our solution on publicly available test mixtures from previously published benchmarks: the ProvedIt dataset [32], MIX05, and MIX13 studies [17]. For the GlobalFiler™ mixtures from the ProvedIt dataset, we use the hyperparametrization suggested by Riman et al. [32]. In case of the MIX05 and MIX13 studies, we use the hyperparameterization suggested by Buckleton et al. [15]. For MIX05, MIX13, and precision studies we use the FBI extended caucasian population [29] as a genetic background model. For the challenging mixtures of Subsection 4.3, we use NIST 1036-Caucasian allele frequencies [33].

We denote the contributors in the hypotheses by plus-delimited strings. U denotes an unknown contributor, W is a witness, V is a victim, other entries denote suspects. For example $V + W + S + U$ would denote 4 contributors: the victim, the witness, the suspect and an unknown person. In every case, the defendant's hypothesis is the prosecutor's hypothesis with the suspect replaced by an unknown contributor. All benchmark mixtures were created in laboratories with known ground-truth genotypes of the contributors.

The linear stutter models are fit on single-source profiles (forward, backward, and double-backward stutter for ProvedIt and MIX13 cases) or on data provided by the kit manufacturers (only backward stutter for MIX05 cases). The models are available in Chapter 4 of the Supplementary Material.

All experiments are performed on affordable equipment in the cloud. We used NC8as_T4_v3 instances from Azure cloud (8 vcpus, Nvidia Tesla T4 with 16 GB RAM). An exception has been made for Sample 3, which did not fit within 16 GB of RAM. For this sample we rented a machine with a Tesla A100 graphics card on the Google Cloud.

Table 3. The results of our model and other solutions on MIX05 and MIX13 studies. In most cases the tested solution provided correctly a likelihood ratio larger than 1 (in case of inclusions) or smaller than 1 (in case of exclusions). The only exception is the case of suspect S05C in MIX13 Case 5. The genotype of this suspect has been constructed deliberately to share alleles with the true contributors in every locus. 74 out of 108 laboratories included in the original study have included this suspect. "Incl." stands for inclusion, "Excl." stands for exclusion. In case of Euroformix we used results of version 1.11.4, the results show the average LR out of all the performed runs (10 runs in our case). TPOX locus was ignored in MIX05 Case 4 due to a tri-allelic pattern of the perpetrator.

Case	H_p	Truth	$\log_{10} LR$		
			HMC	EFM	STRmix™
MIX05 Case 1	Perpetrator+U	Incl.	19.34	-	-
MIX05 Case 2	Perpetrator+U	Incl.	25.67	-	-
MIX05 Case 3	Perpetrator+U	Incl.	22.32	-	-
MIX05 Case 4	Perpetrator+U	Incl.	10.44	-	-
MIX13 Case 1	V+S01A	Incl.	20.15	20.18	20.15
MIX13 Case 2	S02A+U+U	Incl.	17.03	17.28	16.98
MIX13 Case 2	S02B+U+U	Incl.	7.50	7.88	7.26
MIX13 Case 2	S02C+U+U	Incl.	5.41	6.11	5.83
MIX13 Case 2	S02D+U+U	Excl.	-16.18	-2.36	-14.03
MIX13 Case 3	V+W+S03A	Incl.	7.87	6.82	7.69
MIX13 Case 3	V+W+S03B	Excl.	$-\infty$	$-\infty$	$-\infty$
MIX13 Case 4	V+S	Incl.	20.23	19.91	20.15
MIX13 Case 5	S05A+U+U	Incl.	3.38	9.26	3.45
MIX13 Case 5	S05B+U+U	Incl.	1.61	9.38	3.32
MIX13 Case 5	S05C+U+U	Excl.	-8.66	6.45	-9.22
MIX13 Case 5	S05A+U+U+U	Incl.	6.2	-	-
MIX13 Case 5	S05B+U+U+U	Incl.	5.96	-	-
MIX13 Case 5	S05C+U+U+U	Excl.	2.63	-	-

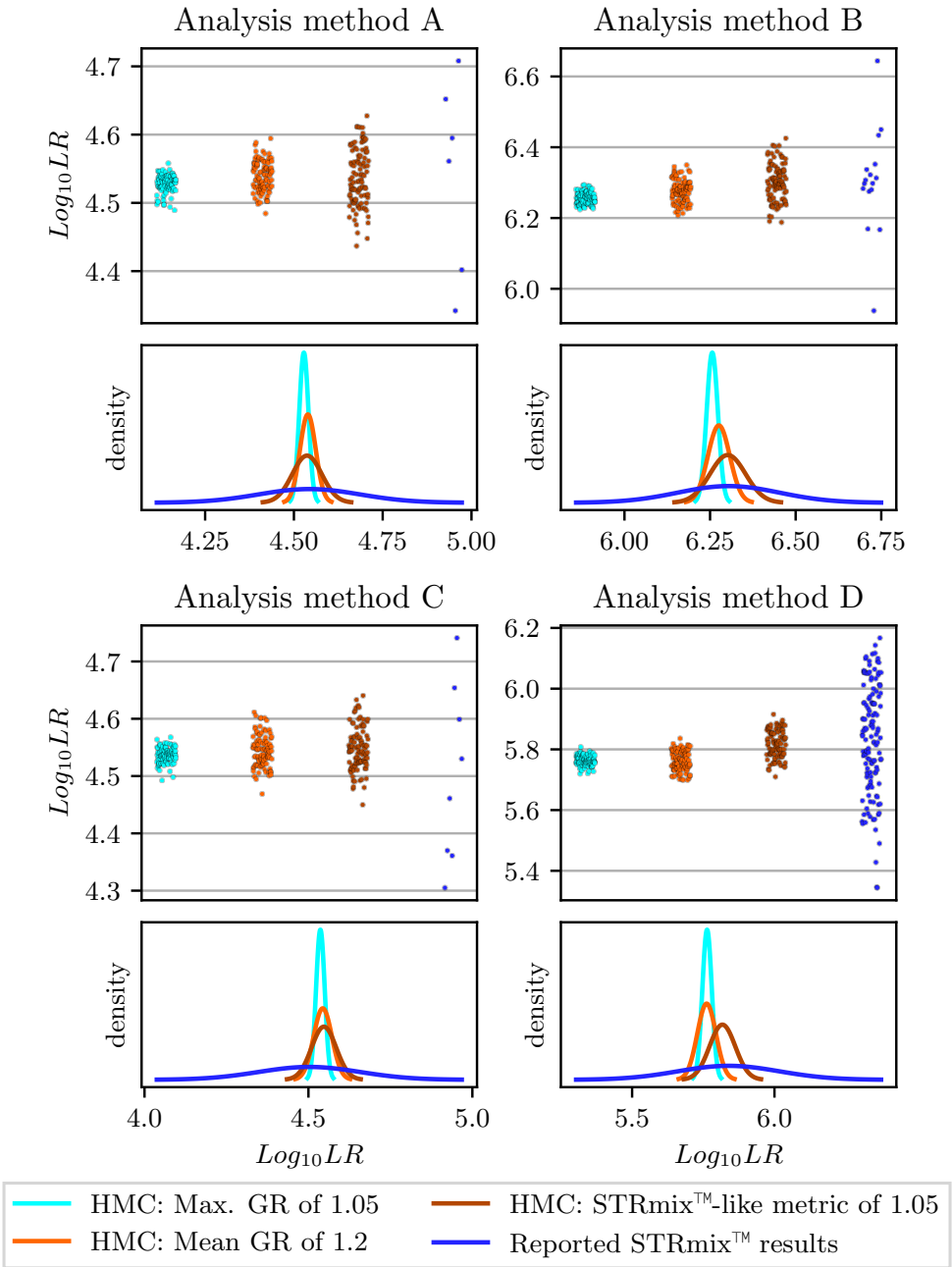
4.1 MIX05 and MIX13 benchmarks

We first benchmark the performance of our method on inter-laboratory studies organised by NIST: MIX05, and MIX13 [17]. In case of MIX05 we analyse simultaneously replicates from different kits: ABI's COFiler, ABI's SGM Plus, Promega's Powerplex 16, and ABI's Profiler Plus. In case of MIX13, we follow the published studies and use only ABI's AmpFLSTR IdentiFiler Plus replicate. All cases are analysed with a global analytical threshold of 50 and the ground truth number of contributors with two exceptions: Case 5 from MIX13 is also analysed with 3 contributors (as most laboratories taking part in the original study estimated this number), and Case 2 from MIX13 uses an analytical threshold of 30 (following the recommendation from NIST). For the capillary electrophoresis fragment analysis files, we used default GeneMapper™ ID-X 1.4 analysis settings. The results are presented in Table 3. Our algorithm reproduces most of the results of other solutions, suggesting its validity. Similar to other solutions, our algorithm provides more conservative values when a smaller number of contributors is chosen [15]. In case of MIX13 Case 4 our algorithm provides a higher LR than the reciprocal of the random match probability. An explanation for this behaviour is given in Section 2.1 of the Supplementary Materials.

4.2 Precision analysis

Next we analyse the precision of our solution in comparison with the state of the art on ProvedIt cases from an interlaboratory study [9]. We used the same analytical

Figure 3. Comparison of precision of our solution under various stopping criteria with previously published results. For every analysis method we run our solution 100 times under three different stopping criteria. For each analysis method we present the raw results in the top figure grouped by the method and the maximum likelihood estimation of a normal distribution on top of the results (bottom plot). The standard deviation of $\log_{10} LR$ is reduced around 10 times when our method is used. In case of published STRmix™ results we ignored the result provided by participant L4A1 (no known contributor included) and second run of the participant L1A1 (not presented on the plot in the original work).



thresholds as Kelly et al. [27]. We report sub-sub-source LRs following [9].

We focus on Sample 1 cases, which were previously used to determine the precision of STRmix™. Four analysis methods of GeneMapper™ were used (called A,B,C, and D). A 10-fold difference in the LR is observed when the software is run multiple times. The authors of the study [9] attribute this lack of precision to the random sampling of MCMC. In Figure 3 we present the results of our HMC implementation when the default stopping criterion is used. Every 300 iterations we check GR and stop the inference if it falls below 1.05 for all parameters. Our solution reduces the standard deviation of the LR on the considered case around 10 fold. For analysis method A our standard deviation is 10.08 times smaller than previously published [9]. For analysis method B it is 8.76 times smaller, for analysis method C 10.99 times smaller, and for analysis method D 10.76 times smaller. In case of Sample 2, Analysis Method C we obtain a $\log_{10} LR$ of 29.0144 ± 0.00254.

.

4.2.1 Precision vs stopping criterion

We quantify the influence of the convergence metric on the precision of the algorithm. We use two alternative stopping criteria: in the first we calculate the mean of GR values and stop the inference if the mean is smaller than 1.2. In the second we use the metric of STRmix™ with a cutoff value of 1.05. In our case, this criterion was sometimes satisfied when the values of GR were still high (mean GR > 1.2). Figure 3 shows the effect of the stopping criterion on the precision.

The difference in precision between our approach and STRmix™ is due also to different convergence criteria. The criterion we use is only possible to apply if the method marginalises out the discrete parameters.

4.3 Computationally challenging mixtures

The final benchmark measures the performance of HMC on the challenging case of mixtures with up to 5 contributors. The goal is to test our solution on cases that are not simple to resolve and are challenging from the precision point of view. For these cases we use the set of low analytical thresholds from Riman et al. [32].

We analyse 4 mixtures: Sample 1 and Sample 2 from Bright et al. [9] analysed here without a known contributor (called here Sample 1b and Sample 2b, respectively), and two 5-person mixtures with one known contributor:

- A05_RD14-0003-30_31_32_33_34-1;1;1;1;1-M3I22-0.315GF-Q1.3_01.15sec
  (named here Sample 3, known contributor is Contributor 31)
- E04_RD14-0003-48_49_50_29_30-1;1;2;4;1-M2d-0.279GF-Q2.1_05.15sec
  (named here Sample 4, known contributor is Contributor 29)

We construct all possible correct prosecutor hypotheses with 1 suspect. We also construct the same number of false hypotheses, choosing the suspects randomly from the NIST 1036 U.S. Population Dataset. We run our method 10 times in each scenario and show the results in Figure 4. In all of the considered cases our algorithm correctly classified contributors and non-contributors. Our solution proves to be precise: in most of the scenarios when the true contributors are considered, the difference between extreme values of $\log_{10} LR$ is under 0.2. The remaining variation comes from the fact that, despite the deduced approximate convergence, the chains consist of stochastic samples.

We present in Table 4 the runtimes in all cases with 3 or more contributors. While they are better than the published runtimes from 2016 [12] (>16 min for 3 unknown

Figure 4. Results for computationally challenging mixtures. For each scenario we run our solution 9 times and plot the sub-source $\log_{10} LR$. In most of the scenarios the difference between the highest and the lowest results is lower than 0.2. We analyse one of the exceptions (Sample 2b when the suspect was contributor 30) in Chapter 2 of the Supplementary Material. We do not plot here exclusion scenarios that resulted in $LR = 0$ for all the runs. These are: Sample 2b - non-contributor C18C_Cauc, C99B_AA and non-contributor ZT80925_Hisp; Sample 3 - non-contributors GT38089_Cauc, JT51484_AA and WT51359_Cauc. The scenarios with a correct suspect are marked with teal colour, the false suspects are marked with orange.

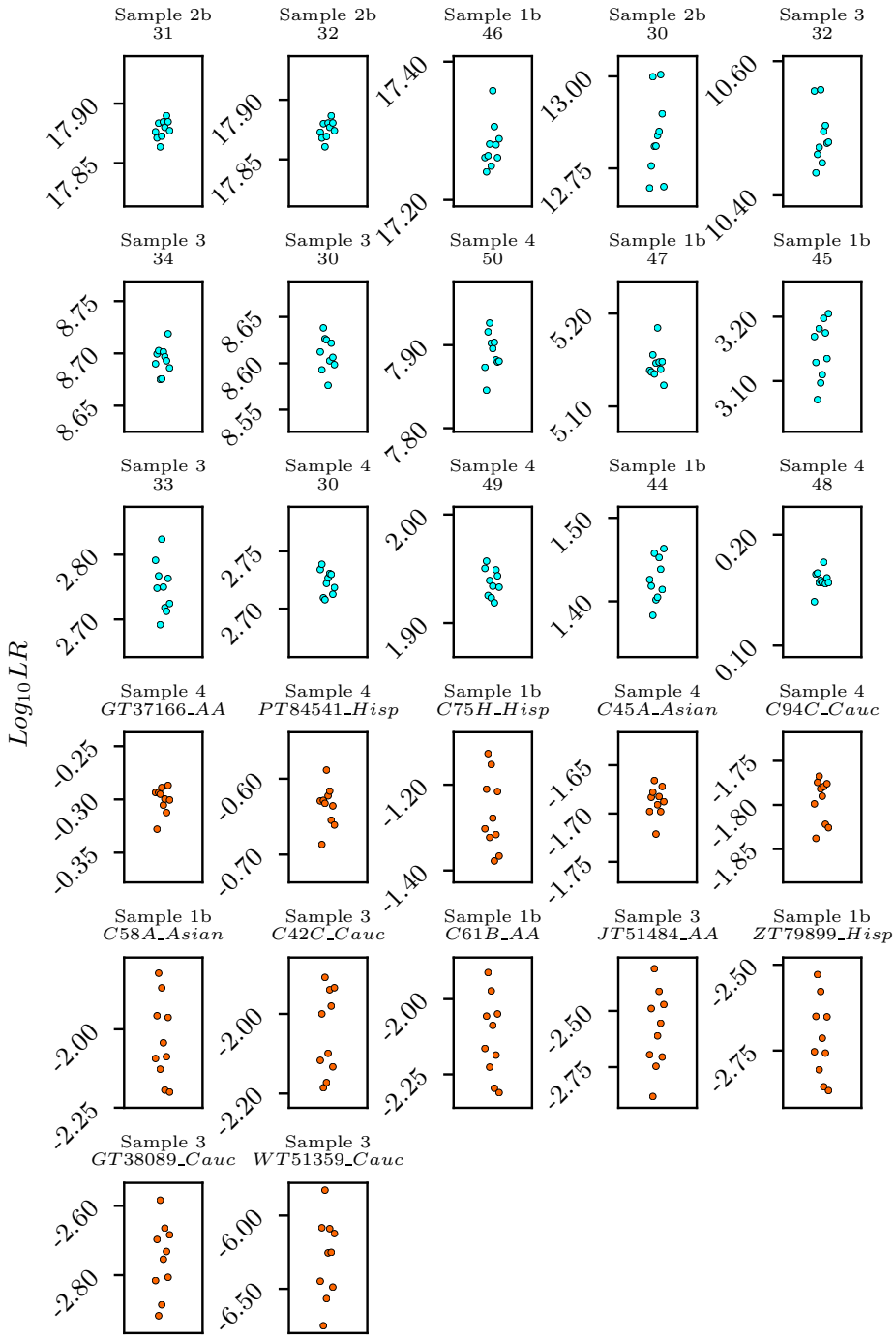


Table 4. The average inference times on GPU for the presented cases with 3 or more contributors. *NoC* stands for "number of contributors".

Case	NoC	NoC known	Inference time
MIX13 Case 2	3	0	3m 59s
MIX13 Case 3	3	0	1m 59s
MIX13 Case 5	3	0	4m 37s
MIX13 Case 5	4	0	31m 15s
Sample 1	4	1	4m 13s
Sample 1	4	0	19m 39s
Sample 2	3	0	6m 55s
5 contr case 1	5	1	59m 55s
5 contr case 2	5	1	27m 15s

contributor mixtures, >1h 40min for 4 unknown contributor mixtures), the runtime of the other solutions might have significantly improved since then as well.

5 Conclusions and future work

Low precision of probabilistic genotyping software can lead to a lowered trust of the users to the results and lowered discriminatory power of the output LRs. Our analysis provides empirical evidence that the stricter convergence metrics are chosen, the better the precision. We presented an implementation of HMC that is able to fulfil strict convergence requirements in a better time frame than previously published results. The solution is able to closely reproduce the previously published LRs on test mixtures, but with an order of magnitude reduced uncertainty when compared with the leading MCMC solution.

In future work we plan to compare our solution against other algorithms on the proposed ProvedIt benchmarks (e.g. [18, 32]), provide comparative analysis on two main approaches algorithmic approaches (Bayesian inference and maximum likelihood estimation) when the same probabilistic model is used and work on further improvements of the precision of the model.

Aside from the variation caused by the algorithm, the precision observed on a single sample in the laboratory depends on multiple factors, including the frequency of the suspect's genotype, the proportion of the suspect's template, the quality of the sample, and the hyperparameterization of the method. These factors must be fixed when comparing different PG solutions. It would be interesting to explore what are the factor that have the largest influence on the variation of the results. The users of the algorithm could be interested in the estimation of the precision of the solution in the worst case.

We did not analyse the precision of techniques based on maximum likelihood estimation. Low precision will not be an issue for an implementation that finds the global maximum.

Acknowledgements

We thank Dr. Sachin Krishnan (Center for Advanced Systems Understanding, Görlitz) and Zofia Dziedzic (University of Wrocław) for discussions on possible improvements of the mathematical notation and Kevin Cheng (Institute of Environmental Science and Research Limited, Wellington) for valuable insights on the published STRmix™ results and the topics of diagnostics of PG software results

References

1. C. Aitken, A. Nordgaard, F. Taroni, and A. Biedermann. Commentary: Likelihood ratio as weight of forensic evidence: A closer look. *Frontiers in Genetics*, 9, June 2018.

2. Association of Forensic Science Providers. Standards for the formulation of evaluative forensic science expert opinion. *Science & Justice*, 49(3):161–164, Sept. 2009.

3. D. J. Balding and J. Buckleton. Interpreting low template dna profiles. *Forensic Science International: Genetics*, 4(1):1–10, Dec 2009.

4. D. J. Balding and R. A. Nichols. DNA profile match probability calculation: how to allow for population stratification, relatedness, database selection and single bands. *Forensic Science International*, 64(2-3):125–140, Feb. 1994.

5. D. W. Bauer, N. Butt, J. M. Hornyak, and M. W. Perlin. Validating TrueAllele ® interpretation of DNA mixtures containing up to ten unknown contributors. *Journal of Forensic Sciences*, 65(2):380–398, Oct. 2019.

6. M. Betancourt. A conceptual introduction to hamiltonian monte carlo, 2018.

7. Ø. Bleka, G. Storvik, and P. Gill. EuroForMix: An open source software based on a continuous model to evaluate STR DNA profiles from a mixture of contributors with artefacts. *Forensic Science International: Genetics*, 21:35–44, Mar. 2016.

8. J.-A. Bright and M. Coble. *Forensic DNA profiling : a practical guide to assigning likelihood ratios*. CRC Press, Boca Raton, 2021.

9. J.-A. Bright et al. STRmix™ collaborative exercise on DNA mixture interpretation. *Forensic Science International: Genetics*, 40:1–8, May 2019.

10. J.-A. Bright, S.-I. Lee, J. Buckleton, and D. Taylor. Revisiting the STRmix™ likelihood ratio probability interval coverage considering multiple factors. *Biorxiv preprint server*, June 2021.

11. J.-A. Bright, D. Taylor, and J. Buckleton. Statement on minor miscoding in bn formulae strmix v1 08 with additional comments, 2016.

12. J.-A. Bright, D. Taylor, C. McGovern, S. Cooper, L. Russell, D. Abarno, and J. Buckleton. Developmental validation of STRmix™, expert software for the interpretation of forensic DNA profiles. *Forensic Science International: Genetics*, 23:226–239, July 2016.

13. S. P. Brooks and A. Gelman. General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, 7(4):434–455, Dec. 1998.

14. J. S. Buckleton. The probability of dropout and drop-in.

15. J. S. Buckleton, J.-A. Bright, K. Cheng, B. Budowle, and M. D. Coble. NIST interlaboratory studies involving DNA mixtures (MIX13): A modern analysis. *Forensic Science International: Genetics*, 37:172–179, Nov. 2018.

16. J. S. Buckleton, K. E. Lohmueller, K. Inman, K. Cheng, J. M. Curran, S. N. Pugh, J.-A. Bright, and D. A. Taylor. Testing whether stutter and low-level DNA peaks are additive. *Forensic Science International: Genetics*, 43:102166, Nov. 2019.

17. J. M. Butler, M. C. Kline, and M. D. Coble. NIST interlaboratory studies involving DNA mixtures (MIX05 and MIX13): Variation observed and lessons learned. *Forensic Science International: Genetics*, 37:81–94, Nov. 2018.

18. K. Cheng, Ø. Bleka, P. Gill, J. Curran, J.-A. Bright, D. Taylor, and J. Buckleton. A comparison of likelihood ratios obtained from EuroForMix and STRmix™. *Journal of Forensic Sciences*, Sept. 2021.

19. K. Cheng, J.-A. Bright, Z. Kerr, D. Taylor, A. Ciecko, J. Curran, and J. Buckleton. Examining the additivity of peak heights in forensic DNA profiles. *Australian Journal of Forensic Sciences*, pages 1–15, Jan. 2020.

20. J. V. Dillon, I. Langmore, D. Tran, E. Brevdo, S. Vasudevan, D. Moore, B. Patton, A. Alemi, M. Hoffman, and R. A. Saurous. Tensorflow distributions, 2017.

21. S. Duane, A. Kennedy, B. J. Pendleton, and D. Roweth. Hybrid monte carlo. *Physics Letters B*, 195(2):216–222, Sept. 1987.

22. Executive Office Executive Office of the President. *Report to the President - Forensic Science in Criminal Courts: Ensuring Scientific Validity of Feature-comparison Methods*. CreateSpace Independent Publishing Platform, 2016.

23. B. L. Garrett, W. E. Crozier, and R. Grady. Error rates, likelihood ratios, and jury evaluation of forensic evidence. *Journal of Forensic Sciences*, 65(4):1199–1209, Apr. 2020.

24. A. Gelman. *Bayesian data analysis*. Chapman & Hall/CRC, Boca Raton, Fla, 2nd edition, 2004.

25. P. Gill, C. Benschop, J. Buckleton, Ø. Bleka, and D. Taylor. A review of probabilistic genotyping systems: EuroForMix, DNAStatistX and STRmix™. *Genes*, 12(10):1559, Sept. 2021.

26. F. M. Götz, H. Schönborn, V. Borsdorf, A.-M. Pflugbeil, and D. Labudde. GenoProof mixture 3—new software and process to resolve complex DNA mixtures. *Forensic Science International: Genetics*, 6:e549–e551, Dec. 2017.

27. H. Kelly, J.-A. Bright, M. Kruijver, S. Cooper, D. Taylor, K. Duke, M. Strong, V. Beamer, C. Buettner, and J. Buckleton. A sensitivity analysis to determine the robustness of STRmix™ with respect to laboratory calibration. *Forensic Science International: Genetics*, 35:113–122, July 2018.

28. K. Kwong. The algorithm says you did it: The use of black box algorithms to analyze complex dna evidence. *Harvard Journal of Law & Technology*, 31:275, 2017.

29. T. R. Moretti, L. I. Moreno, J. B. Smerick, M. L. Pignone, R. Hizon, J. S. Buckleton, J.-A. Bright, and A. J. Onorato. Population data on the expanded CODIS core STR loci for eleven populations of significance for forensic DNA analyses in the united states. *Forensic Science International: Genetics*, 25:175–181, Nov. 2016.

30. G. S. Morrison. Special issue on measuring and reporting the precision of forensic likelihood ratios: Introduction to the debate. *Science & Justice*, 56(5):371–373, Sep 2016.

31. M. W. Perlin, M. M. Legler, C. E. Spencer, J. L. Smith, W. P. Allan, J. L. Belrose, and B. W. Duceman. Validating TrueAllele ® DNA mixture interpretation. *Journal of Forensic Sciences*, 56(6):1430–1447, Nov. 2011.

32. S. Riman, H. Iyer, and P. M. Vallone. Examining performance and likelihood ratios for two likelihood ratio systems using the PROVEDIt dataset. *PLOS ONE*, 16(9):e0256714, Sept. 2021.

33. C. R. Steffen, M. D. Coble, K. B. Gettings, and P. M. Vallone. Corrigendum to 'us population data for 29 autosomal str loci'[forensic sci. int. genet. 7 (2013) e82–e83]. *Forensic Science International: Genetics*, 31:e36–e40, 2017.

34. D. Taylor, J.-A. Bright, and J. Buckleton. The interpretation of single source and mixed DNA profiles. *Forensic Science International: Genetics*, 7(5):516–528, Sept. 2013.

35. D. Taylor, J.-A. Bright, and J. Buckleton. The 'factor of two' issue in mixed DNA profiles. *Journal of Theoretical Biology*, 363:300–306, Dec. 2014.

36. D. Taylor, J.-A. Bright, and J. Buckleton. Using probabilistic theory to develop interpretation guidelines for y-STR profiles. *Forensic Science International: Genetics*, 21:22–34, Mar. 2016.

37. W. C. Thompson. Painting the target around the matching profile: the texas sharpshooter fallacy in forensic dna interpretation. *Law, Probability and Risk*, 8(3):257–276, Jul 2009.

38. D. Vats and C. Knudson. Revisiting the Gelman–Rubin diagnostic. *Statistical Science*, 36(4), Nov. 2021.

39. S. Willis, A. Ligertwood, J. Molina, C. Berger, G. Zadora, A. Nordgaard, B. Rasmusson, L. Lunt, C. Champod, A. Biedermann, T. Hicks, F. Taroni, and X. Zhu. Enfsi guideline for evaluative reporting in forensic science, 03 2015.