STROOPWAFEL

simulating rare outcomes from astrophysical populations, with application to gravitational-wave sources★


Published in:
Monthly Notices of the Royal Astronomical Society

DOI:
10.1093/mnras/stz2558

Link to publication

Citation for published version (APA):

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (http://dare.uva.nl)
STROOPWAFEL: simulating rare outcomes from astrophysical populations, with application to gravitational-wave sources


Affiliations are listed at the end of the paper

Accepted 2019 August 26. Received 2019 August 16; in original form 2019 May 2

ABSTRACT
Gravitational-wave observations of double compact object (DCO) mergers are providing new insights into the physics of massive stars and the evolution of binary systems. Making the most of expected near-future observations for understanding stellar physics will rely on comparisons with binary population synthesis models. However, the vast majority of simulated binaries never produce DCOs, which makes calculating such populations computationally inefficient. We present an importance sampling algorithm, STROOPWAFEL, that improves the computational efficiency of population studies of rare events, by focusing the simulation around regions of the initial parameter space found to produce outputs of interest. We implement the algorithm in the binary population synthesis code COMPAS, and compare the efficiency of our implementation to the standard method of Monte Carlo sampling from the birth probability distributions. STROOPWAFEL finds \( \sim 25-200 \) times more DCO mergers than the standard sampling method with the same simulation size, and so speeds up simulations by up to two orders of magnitude. Finding more DCO mergers automatically maps the parameter space with far higher resolution than when using the traditional sampling. This increase in efficiency also leads to a decrease of a factor of \( \sim 3-10 \) in statistical sampling uncertainty for the predictions from the simulations. This is particularly notable for the distribution functions of observable quantities such as the black hole and neutron star chirp mass distribution, including in the tails of the distribution functions where predictions using standard sampling can be dominated by sampling noise.

Key words: gravitational waves – methods: statistical – software: development – binaries: general – stars: evolution.

1 INTRODUCTION
The direct detection of gravitational waves originating from merging binary black holes (BHs) and neutron stars (NSs) has opened up a new window on the Universe, and marked the birth of gravitational-wave astrophysics as a new field of research (Abbott et al. 2016, 2017). At the time of writing, the first two observing runs of Advanced LIGO and Virgo have been completed (The LIGO Scientific Collaboration 2018). A few dozen detections are expected during the third observing run, and we can anticipate hundreds of detections per year when the next generation of detectors with higher sensitivities comes online (Abbott et al. 2018).

The detections are starting to reveal the properties of the population of merging binary BHs and NSs. The distributions of the inferred masses and spins contain valuable information about their origin. Distinguishing different theories for their formation and learning about the complex physical processes that govern the lives of their possible massive star progenitors require comparing observed populations with theoretical predictions.

Theoretical simulations of the population of merging double compact objects (DCOs) are challenging because gravitational-wave events represent an extremely rare outcome of binary evolution. From a thousand massive binary systems, typically only of the order of one, or less, yields a DCO. A meaningful comparison with population observations requires simulating a statistically significant sample of events. When sampling from the birth distributions, which is a form of sampling commonly used in binary population synthesis, this often means we need to sample at least many millions of binary systems. For example, Kruckow et al.
In this paper, we present a new algorithm, STROOPWAFEL.\textsuperscript{1} We have designed STROOPWAFEL to improve the efficiency of simulations of rare astrophysical events, and so to enable accurate simulations of populations of extremely rare events at reasonable computational cost. The algorithm first explores the initial parameter space until it finds a preliminary population of systems of interest. This exploration is done by stochastically sampling from the birth distributions. STROOPWAFEL then concentrates the later sampling towards regions in the initial parameter space that are in the vicinity of the initial parameters of the interesting binaries found during the exploration phase. This is an example of ‘Adaptive Importance Sampling’ (AIS), described further in the next section.

We focus here on the application of the study of DCO mergers as gravitational-wave sources, but our algorithm is much more broadly applicable. The user can specify any target population of interest. An advantage of the algorithm is that it can handle the bifurcations and stochastic behaviour that naturally occur in the physical processes in binary population synthesis simulations, and which lead to discontinuous output surfaces. Finally, with our algorithm we can easily derive the uncertainties on the estimated parameters, which can be a challenge for sampling methods such as MCMC and Gaussian process regression emulators.

This paper is organized as follows. In Section 2, we describe the algorithm and provide expressions for how to calculate statistical estimates and the uncertainties from the simulations. We further derive the optimal relative duration of the exploratory phase and the total number of simulations, given the rarity of the target population. In Section 3, we provide a demonstration of our algorithm. We apply it to population synthesis simulations of DCO mergers. In Section 4, we outline caveats and future directions for further improvement and refinement of the algorithm. We conclude and summarize in Section 5.

2 METHOD

Our algorithm is conceptually simple. It uses a strategy that may be familiar from playing the classic game Battleship. The aim of this game is to guess the coordinates of the ships of the other player, which are placed on a regular discrete grid. Most players will probably start with an exploration phase randomly trying different coordinates. After one or more successful ‘hits’, most players will change their search strategy and instead try to refine their search by trying coordinates that are close to the successful hits until they uncover the full location and orientation of the ship. It has been shown that this is a more successful strategy compared to searching completely randomly throughout the entire game (e.g. Jones 1977).

Our algorithm, STROOPWAFEL, follows a conceptually similar strategy, but instead of aiming to win a game of Battleship the algorithm is designed to improve the efficiency for simulating populations of rare events (that is, rare outcomes from the space of initial conditions). Successful hits in this analogy are finding systems of interest that are part of a certain target population. These

\textsuperscript{1}All data that have been used for this study, accompanied with a Jupyter notebook that reproduces the main results, can be found here (https://doi.org/10.5281/zenodo.3387651). The code for the STROOPWAFEL algorithm will be made publicly available after acceptance. Early inquiries can be addressed to the lead author.
may be systems that result in DCO mergers or anything that the user specifies. We improve the efficiency by focusing on areas of the initial parameter space near to those that produced outcomes of interest during a prior, exploratory, sampling phase. Instead of Monte Carlo sampling from the birth probability distributions, STROOPWAFEL uses information from that exploratory sampling phase to create an alternative distribution function, from which it then samples.

This class of Monte Carlo methods is generically called ‘Importance Sampling’ (Kahn & Harris 1951; Kahn & Marshall 1953). Since we do not know in advance which areas of the initial parameter space should receive extra attention, we use AIS, for which see e.g. Torrie & Valleau (1977), Hesterberg (1995), Cappé et al. (2004), Pennanen & Koivu (2006), Cornuet et al. (2012), and Ortiz & Pack Kaelbling (2013). The nature of the AIS algorithm makes it straightforward to tune the focus of the simulation on a specific target population or function of interest (Cappé et al. 2008). Such AIS algorithms also allow for straightforward calculations of the sampling uncertainties. The STROOPWAFEL implementation of AIS is similar to that in Cornuet et al. (2012), but includes a new method to guide the fraction of the computational effort that should be spent on the exploratory phase (see Section 2.2.4).

While the concept of our algorithm is not complicated, there are some mathematical details involved in making the implementation efficient and robust. For example, some of the subtlety involved is in making sure not to concentrate too closely on locations that previously led to success. If we only look exactly where we have looked before, then we do not learn anything new.

Section 2.1 introduces binary population synthesis as a mapping between input and output parameter spaces, along with some notation that is useful for the description of our algorithm. Section 2.2 presents the key details of STROOPWAFEL. We explain how we shape the adaptive sampling distribution from the information found in an initial exploratory phase, and how to optimally combine the samples from both the exploratory and adapted phases to estimate the population quantities of interest. We also describe how STROOPWAFEL self-consistently determines how long the exploratory phase should last as a fraction of the simulation time, based on continually updated estimates of the rareness of the target population. Section 2.3 illustrates the practical characteristics of our AIS algorithm in an idealized way, providing an explanatory summary of the behaviour of STROOPWAFEL for users who do not wish to learn all the mathematical details.

2.1 Definition of concepts and symbols

Binary population synthesis models the population observables for a particular class of event, under a set of assumptions about the physics. Predicting such an output population typically involves simulating many individual systems from their initial conditions. Only a small fraction of those simulated systems may produce outcomes that are of interest for that study.

Selecting which specific points in the input space (i.e. the initial conditions) to simulate into the output space (i.e. the observables) is a key part of population synthesis. This process is called sampling, and must appropriately take into account the relative frequency of different initial conditions. Ideally, it should also efficiently explore the initial parameter space. Examples of these initial parameters are the initial masses of the two stars, \( m_{1,i} \) and \( m_{2,i} \), and the initial separation \( a_i \) between the two stars. For a given initial composition, these three dimensions are often regarded as adequate initial conditions. However, more generally, these input conditions may be distributed over many dimensions.

Each initial binary system \( x_i \in \{x_1, ..., x_N\} \) in a binary population synthesis can thus be written as \( x_i = (m_{1,i}, m_{2,i}, a_i, ...) \), which has a combined birth distribution

\[
\pi(x_i) = \pi(m_{1,i}, m_{2,i}, a_i, ...) \tag{1}
\]

This distribution of initial conditions is often taken as the Monte Carlo sampling distribution. In practice, simulations of binary-star populations that aim to study outcomes such as mergers between BHs and NSs do not sample from the full range of initial conditions of stellar binaries. Such simulations ignore stars whose mass is too low to produce a BH or NS, which is a simple form of importance sampling. The normalization of \( \pi \) actually used for the sampling is then corrected to take this into account when predicting event rates.

For each initial binary, \( x_i \), the final state of the binary \( y_f \) is determined using the binary population synthesis model \( u \),

\[
y_f = u(x_i) \tag{2}
\]

In many cases, a simulation is run to study binaries that evolve to a certain target subtype \( T \), e.g. maybe \( T \) is the population of binary BH mergers. The following indicator function describes whether a binary \( y_f \) is of interest:

\[
\mathbb{1}_T(y_f) := \begin{cases} 1 & \text{if } y_f \in T \text{ (a hit)} \\ 0 & \text{if } y_f \notin T \text{ (a miss).} \end{cases} \tag{3}
\]

which equals 1 if \( x_i \) simulates to the target binary system \( T \) (a hit) and zero if not (a miss). Combining equations (2) and (3) gives the function

\[
\phi(x_i) := \mathbb{1}_T(u(x_i)) \tag{4}
\]

which is a shorthand notation to describe whether an initially drawn binary evolved into a binary of the target population.

The samples from the initial parameter space that produced a binary of the target population [i.e. \( \phi(x_i) = 1 \)] can then be given by the set

\[
x_T := (m_{1,T}, m_{2,T}, a_T, ...) \tag{5}
\]

At the end of a simulation, the properties of the model population and the statistical uncertainties on those predicted properties can both be determined using the standard Monte Carlo estimator (Fermi & Richtmyer 1948; Metropolis & Ulam 1949). For example, the relative formation rate of the target population, \( R_T \), is estimated with

\[
\mathbb{E}_x[R_T] \approx \frac{1}{N} \sum_{i=1}^{N} \phi(x_i), \tag{6}
\]

where \( N \) is the total number of samples used, \( E \) is the notation for the estimated mean, and the subscript \( x \) denotes that the samples \( x_i \) are distributed following the birth distribution \( \pi \) (cf. equation 1). We shall refer to this relative formation rate, \( R_T \), throughout this section. Mathematically, it is a fractional volume from the initial

\[\text{Binary population synthesis simulations often sample in the initial mass ratio, } m_{2,i}/m_{1,i}, \text{ rather than the mass of the initially least massive star. This includes the simulations we present later; see also Appendix C. Whether sampling in } m_{2,i} \text{ or the mass ratio, it has also been common to assume that the initial parameters are independent of each other, e.g.: } \pi(x_i) = \pi(m_{1,i}) \cdot \pi(m_{2,i}) \cdot \pi(a_i) \cdot \ldots \ldots . \text{ This assumption of separability may not be valid, as found by Abt, Gomez & Levy (1990), Moe & Di Stefano (2017), and Klencki et al. (2018).} \]
Table 1. Summary of the parameters that are used throughout this paper. Hits refer to binaries of the chosen target population.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(u)</td>
<td>Binary population synthesis model</td>
</tr>
<tr>
<td>(x_i)</td>
<td>Initial state of a binary system</td>
</tr>
<tr>
<td>(y_f)</td>
<td>Final state of a binary system</td>
</tr>
<tr>
<td>(T)</td>
<td>Target subpopulation of binaries of interest</td>
</tr>
<tr>
<td>(\pi(x))</td>
<td>Set of hits from the exploratory phase</td>
</tr>
<tr>
<td>(\Omega)</td>
<td>Physical parameter range of the simulation</td>
</tr>
<tr>
<td>(q(x))</td>
<td>Instrumental distribution</td>
</tr>
<tr>
<td>(q(x_\text{ref}))</td>
<td>Normalized instrumental distribution</td>
</tr>
<tr>
<td>(N)</td>
<td>The total number of samples in the simulation</td>
</tr>
<tr>
<td>(N_{\text{expl}})</td>
<td>The number of binaries in the exploratory phase</td>
</tr>
<tr>
<td>(N_{\text{ref}})</td>
<td>The number of binaries in the refinement phase</td>
</tr>
<tr>
<td>(N_T)</td>
<td>The number of systems of the target population</td>
</tr>
<tr>
<td>(T_{\text{expl}})</td>
<td>The number of hits in the exploratory phase</td>
</tr>
<tr>
<td>(f_{\text{expl}})</td>
<td>The fraction of samples in the exploration phase</td>
</tr>
<tr>
<td>(\kappa)</td>
<td>Scale factor for the widths of the Gaussians</td>
</tr>
<tr>
<td>(w_i)</td>
<td>Statistical weight of sample (x_i)</td>
</tr>
<tr>
<td>(\tilde{w}_i)</td>
<td>Recomputed statistical weight</td>
</tr>
</tbody>
</table>

binary parameter space, weighted by the probability of forming a binary system at each part of that initial parameter space. This quantity is not a physical rate, but gives a formation rate for the population of interest as a fraction of the total number of initial binary systems formed. So it only differs from a true formation rate by a physical normalization. We consider it appropriately intuitive to keep referring to this as a fractional, or relative, rate. A summary of the parameters and symbols used throughout this paper is shown in Table 1.

2.2 Adaptive sampling algorithm to increase efficiency of simulation

Our algorithm consists of three main steps, as illustrated in Fig. 1:

(I) Exploration: We first explore the parameter space by sampling directly from the birth distribution \(\pi\) until eventually a sufficient population of events of interest is found.

(II) Adaptation: We construct multivariate Gaussian distributions in the initial parameter space around each of the events of interest found during the exploration phase. We scale the widths of each of the Gaussians with the local sampling density. We create the adapted sampling distribution \(q\), from here on referred to as the instrumental distribution, by combining the Gaussians into a mixture distribution.

(III) Refinement: We draw the samples for the remaining simulations from this instrumental distribution. Each sample is assigned a weight so that the predicted population appropriately reflects the birth distribution \(\pi\).

The rest of this subsection explains these steps in more detail.

2.2.1 The instrumental distribution

When the exploratory phase has ended, the set of binaries of the target population \(x_T\) (i.e. hits) contains \(N_{T, \text{expl}}\) binaries that were found using a total of \(N_{\text{expl}}\) samples. In the STROOPWAFEL algorithm, these samples are then used to create an adapted instrumental distribution \(q(x)\), which is focused around the areas in the initial parameter space that produced the binaries of interest during the exploratory phase. The remaining binaries are thereafter sampled from this instrumental distribution. To obtain unbiased estimates of the target population, weights \(w_i\) are incorporated for each sample as is standard in importance sampling

\[
\tilde{w}_i = \frac{\pi(x_i)}{q(x_i)},
\]

where \(\pi\) is the distribution of initial conditions, as given in equation (1).

The instrumental distribution \(q(x)\) can be chosen to be any probability distribution function, but a robust instrumental distribution is characterized by the following criteria:

(i) The weights \(w_i\) are always finite and well defined. That is, \(q(x_i) = 0\) implies \(\pi(x_i) \neq 0\) for all \(i\).

(ii) The instrumental distribution is efficient if \(q(x)\) is close to the (unknown) target distribution of the binary population synthesis study, i.e. when the instrumental distribution \(q(x)\) is proportional to \(\pi(x)\), as shown by Kahn & Marshall (1953).

(iii) It should be computationally inexpensive to generate random samples from \(q(x)\) as well as to calculate the probability \(q(x_i)\) for each sample \(x_i\).

In order to achieve these properties, the instrumental distribution, \(q(x)\), in STROOPWAFEL is chosen to be a mixture\(^3\) of \(N_{T, \text{expl}}\) Gaussian distributions \(q_k\) given by

\[
q(x) = \frac{1}{N_{T, \text{expl}}} \sum_{k=1}^{N_{T, \text{expl}}} q_k(x; \mu_k, \Sigma_k),
\]

where each \(q_k\) contributes \(1/N_{T, \text{expl}}\) to the mixture distribution.

However, when drawing from \(q(x)\), some samples will fall outside the physical range of the parameter space \(\Omega\) (e.g. when drawing a binary with a negative stellar mass). Such samples can immediately be rejected and redrawn. By doing so, we in practice sample from the normalized physical mixture distribution

\[
\tilde{q}(x) = \frac{1}{1 - F_{\text{rej}}} q(x) 1_{\Omega}(x),
\]

where \(1_{\Omega}(x)\) is the indicator function that equals 1 when the sample \(x\) lies in the physical range of the parameter space and 0 if not. The factor \(1 - F_{\text{rej}}\) is the fraction of samples from \(q(x)\) that are drawn outside of the physical parameter space. The factor \(1/(1 - F_{\text{rej}})\) thus corrects for the normalization of \(\tilde{q}(x)\). It is computationally inexpensive to draw samples from \(\tilde{q}(x)\) since \(F_{\text{rej}}\) can be estimated once with a Monte Carlo simulation, and one can draw randomly from each Gaussian \(q_k(x)\) separately.

The Gaussian distributions \(q_k(x; \mu_k, \Sigma_k)\) are parametrized by their means \(\mu_k\) and covariance matrices \(\Sigma_k\). The covariance matrix, \(\Sigma_k\), determines the width of the Gaussian distributions. We adopt a diagonal covariance matrix

\[
\Sigma_k = \begin{bmatrix}
\sigma_{1,k}^2 & 0 & \cdots \\
0 & \ddots & \\
\vdots & \ddots & \sigma_{d,k}^2
\end{bmatrix},
\]

where \(d\) is the number of dimensions of the initial parameter space.

\(^3\)In this context, ‘mixture’ has a standard mathematical meaning. A sample drawn from \(q\) is drawn from each Gaussian \(q_k\) with a probability \(N_{T, \text{expl}}\) instead of taking the sum of normally distributed samples. The sum of two jointly normally distributed random variables will still have a normal distribution (even if the means are not the same) whereas a mixture of two normally distributed variables will have two peaks (assuming the means are far enough apart).
We scale the width of each Gaussian, given by the covariance matrix $\Sigma_k$, with the average distance to the next sampled binary $x_i$ in the prior distribution. This allows the algorithm to construct broader Gaussian distributions around previously found hits that lie in the regions of the parameter space that are less densely explored. If the one-dimensional marginalized prior of the $j$-th parameter is $\pi_j$, then the standard deviation $\sigma_{j,k}$ is given by

$$\sigma_{j,k} = \kappa \frac{1}{\pi_j(x_k) N_{\text{expl}}}$$

(11)

where $N_{\text{expl}}$ represents the number of samples used for the exploration phase, the power $1/d$ scales this number to the effective number of samples per dimension and the factor $\pi_j(x_k)$ scales the width to the density of samples in the exploration phase. We also introduce a free parameter, $\kappa$, that scales the widths of the Gaussian distributions. This enables us to regulate how tightly the mixture of Gaussian distributions covers the parameter space near the successful binaries $x_T$. In this paper, we adopt $\kappa = 2$, which we chose the following tests with a toy model (for which, see Section 2.2.5 and Appendix B).

### 2.2.2 Combining samples from the exploratory phase and refinement phase

It is desirable to make use of the samples from both the exploration and refinement phases. The optimal way to achieve this is somewhat subtle. In principle, this could be done by merging the samples and weights into a combined estimate (see chapter 14 in Robert & Casella 2013). However, Hesterberg (1995), Veach & Guibas (1995), and later Owen & Zhou (2000) showed that using deterministic multiple mixture weights is an efficient and robust way of combining the samples. This approach uses the fact that the combined samples from the exploratory phase and refined samples can be represented by a mixture sampling distribution $Q(x)$ from both phases

$$Q(x) = f_{\text{expl}} \pi(x) + (1 - f_{\text{expl}}) q(x),$$

(12)

where $f_{\text{expl}} = N_{\text{expl}}/N$ is the fraction of samples spent on the exploratory phase. By analogy with equation (7), the weights of all the $N$ samples can be recalculated with

$$\tilde{w}_i = \pi(x_i) / Q(x_i).$$

(13)

This use of deterministic multiple mixture weights is not fundamental to STROOPWAFEL. Our motivation for using deterministic multiple mixture weights is conservative, to increase the stability against potential sampling artefacts. One of the samples drawn from the important function $q(x)$ may occasionally be extremely large. Such extreme weights could remain so large as to be problematic when merging the samples with the original weights $w_i$ — no matter how efficient the sampling is in the exploratory phase (Cornuet et al. 2012). The use of deterministic multiple mixture weights suppresses this potential rare difficulty.

The multiple mixture weight approach ignores the distribution from which a given draw was sampled. This does not affect the estimators for the predicted values, although it does introduce a very small bias to the uncertainty estimators, which we confirmed to be negligible in our toy model tests. Recalculating the weights in this way yields comparable or better estimates than those that are obtained when merging the samples or using inverse-variance weighting for our AIS algorithm. Indeed, He & Owen (2014) derived a bound for the variance of the balance heuristic for such estimators that combine samples from different distributions and found that this is an efficient way of combining samples. See also section 3 in Veach & Guibas (1995) and section 2 in Cornuet et al. (2012) for a more detailed discussion.

### 2.2.3 Calculating statistical estimates using the adaptive distribution

At the end of each run, the properties of the target population, such as the rate of formation $R_T$ of members of the target population, and distribution functions of population observables, can be determined by standard Monte Carlo estimates. Because we have drawn the samples from a different distribution than the birth distribution, we have to incorporate weights to make sure that the estimators for these quantities reflect the correct formation probabilities. For
example, the relative formation rate $R_T$ of the target population within the simulation is estimated by the mean

$$R_T \approx \mathbb{E}[R_T] = \frac{1}{N} \sum_{i=1}^{N} \phi(x_i) w_i.\quad (14)$$

The uncertainty in this rate is represented by the variance about the mean by

$$\mathbb{V}[R_T] = \frac{s^2_0[\phi(x)\mu(x)]}{N} = \frac{N}{N} \sum_{i=1}^{N} \phi(x_i)^2 \overline{w}_i^2 - R_T^2,\quad (15)$$

where $s_0$ is the (sample) standard deviation for samples drawn from the mixture sampling distribution $Q(x)$. This equation is known as the asymptotic variance. The other moments or statistical estimates for the target population can be similarly calculated.

### 2.2.4 The duration of the exploratory phase

An important choice in AIS algorithms is deciding when to switch from the exploratory phase to the refinement phase. This choice can have a substantial impact on the performance of the algorithm. Leaving the exploratory phase too early can result in missing important regions of the initial parameter space that produce systems in the target population. On the other hand, switching to the refinement phase too late will miss out on the advantages of the algorithm, as most time will be spent sampling from the birth distributions instead of the more efficient adapted distribution.

A method often used in AIS algorithms to determine the fraction of samples that should be spent on exploring the parameter space is by using the effective sample size: $(\sum w_i^2)^2/\sum (w_i^2)$, (ESS; Hesterberg 1995; Liu 2008). This is a measure of efficiency and corresponds to the equivalent number of independent samples drawn from the prior distribution. However, it can be difficult to know in advance what a good value for the ESS should be. Instead, since STROOPWAFEL is a two-step adaptive algorithm, we can directly derive a value for $f_{\text{expl}}$ by using the estimated rareness of the population, which we self-consistently calculate during the exploratory phase.

Here we estimate the optimal fraction $f_{\text{expl}}$ of the total number of samples $N$ that we should spend on the exploratory phase. The challenge is that we do not know in advance how good our adaptive sampling distribution will be. Here, as a simplified proxy for the imperfect sampling distribution, we assume that the adaptive sampling distribution is determined sufficiently well that it perfectly matches the target distribution over most of the parameter space, but that a small region of the target parameter space could be missing samples due to a limited number of samples drawn during the exploratory phase, and thus have an adaptive sampling probability of zero (see Appendix A for details). In other words, we divide the volume of the input parameter space that successfully produces systems of interest into two parts, one which we assume we have accurately found and one which remains missing. We then find $f_{\text{expl}}$ such that the event rate uncertainty is minimized; specifically, we require that the contribution to the event rate from potentially undiscovered islands is smaller than, or no worse than similar to, the sampling uncertainty in the rate contributed by the islands that are successfully found.

We assume that we sample from the mixture distribution $Q(x)$, and aim to estimate the rate $R_T$ with $N$ total samples. After simulating $N$ samples, we assume we have identified a target binary-forming region with total weight $z_1$, whereas a region with weight $z_2$ is yet undiscovered such that the estimated rate of the target population is

$$R_T = z_1 + z_2 \approx \mathbb{E}[R_T].\quad (16)$$

The uncertainty on this rate estimate is described by the variance, which we can approximate with (see Appendix A for more details)

$$\mathbb{V}[R_T] \approx \frac{1}{N} \left[ \frac{z_1}{z_1 f_{\text{expl}}} + \frac{z_2}{f_{\text{expl}}} - (z_1 + z_2)^2 \right].\quad (17)$$

The smallest uncertainty on the rate $R_T$ is obtained for the value of $f_{\text{expl}}$ where the variance $\mathbb{V}[R_T]$ is the lowest. By taking the derivative of $\mathbb{V}[R_T]$ with respect to $f_{\text{expl}}$, and then finding the roots of the derivative, we find that this minimum occurs at

$$f_{\text{expl}} = 1 - \frac{z_1(\sqrt{1 - z_1} - \sqrt{z_2})}{\sqrt{1 - z_1(1 - z_1) + z_1}}.\quad (18)$$

To make practical use of this during our simulations, we need ongoing live estimates for $z_1$ and $z_2$. For the region that has been identified, we adopt $z_1 \approx \mathbb{E}_z[R_T]$ during the simulation using equation (14). We approximate the target population region that is yet undiscovered, $z_2$ by $z_2 \approx \frac{1}{(1/f_{\text{expl}}N)}$. This represents the weight of stochastic sampling noise in the exploratory phase when using a total of $f_{\text{expl}}N$ samples. Moreover, this choice of $f_{\text{expl}}$ ensures that the estimated uncertainty on the rate estimate is always comparable or larger than the uncertainty of missing a region, $1/f_{\text{expl}}N$.

While the running estimate of $z_2$ on the right-hand side of equation (18) is a function of $f_{\text{expl}}$, rather than explicitly solving for $f_{\text{expl}}$, we choose to iteratively approach the optimal solution over the course of the exploratory phase. The resulting $f_{\text{expl}}$ is similar to the $f_{\text{expl}}$ that is obtained if we had used the dependence of $z_2$ on $f_{\text{expl}}$ when solving for the minimum in equation (17).

We note that we have implicitly assumed that the adaptive sampling phase is perfectly efficient, i.e. that all the draws in the adaptive sampling phase find a member of the target output population. Here, that is a conservative assumption, since a less-than-perfect efficiency will increase the sampling uncertainty with respect to the known islands (i.e. $z_1$). Therefore, imperfect efficiency in the adaptive sampling phase decreases the chance that the uncertainty from undiscovered islands is significant.

STROOPWAFEL internally uses equation (18) to estimate $f_{\text{expl}}$ for clarity here, we can additionally assume that $z_1$ and $z_2$ are much smaller than 1, i.e. that the target population is a rare outcome of the initial conditions. Then we obtain the simplified equation

$$f_{\text{expl}} \approx 1 - \frac{z_1}{z_1 + \sqrt{z_2}}.\quad (19)$$

From this simplified equation, it becomes clear that we recover the intuitively correct limit for extremely rare events, i.e. if $z_1 = \mathbb{E}_z[R_T] \rightarrow 0$, we find $f_{\text{expl}} \rightarrow 1$, which suggests that we should spend all our simulation time on exploration, as expected. On the other hand, once we find at least one hit in the exploratory phase we find $f_{\text{expl}} \neq 1$, and so the variance of our rate estimate is expected to decrease when drawing some of the samples from the adapted distribution, compared to taking all samples from the birth distribution.

Lastly, from equation (17) it also becomes clear that $f_{\text{expl}} \approx 0.5$ once we have found $N_{T,\text{expl}} \sim \sqrt{f_{\text{expl}}}$ target binaries. In other
words, \( f_{\text{expl}} \sim 1 \) if \( N \leq 1/R_T^2 \), and therefore the total number of samples \( N \) should generally be similar to or larger than \( 1/R_T^2 \).

### 2.2.5 Determining the free parameter \( \kappa \) from tests with a toy model

We present results from the application of our method to astrophysical simulations in Section 3. Here we explore the methodology with a toy model to test the performance of the algorithm and determine the value of the free parameter \( \kappa \). In principle, \( \kappa = 1 \) could be adopted. Smaller values of \( \kappa \) will increase the efficiency of the STROOPWAFEL algorithm, but increase the chance of missing an important region of the output surface because the Gaussian distributions \( q_k \) are too narrow and do not cover the output surface well. Excessively large values of \( \kappa \), meanwhile, will decrease the efficiency of finding samples of interest in the refinement phase and lower the gain of STROOPWAFEL. After performing tests with a toy model, as described in Appendix B, we adopt the value \( \kappa = 2 \). However, when applying STROOPWAFEL in higher dimensions the optimal value for \( \kappa \) may well change.

### 2.2.6 Summary of STROOPWAFEL algorithm

The algorithm for STROOPWAFEL, combining the methods discussed in this section, is summarized in Algorithm 1.

**Algorithm 1: STROOPWAFEL algorithm**

1. \( i = 0; \)
2. 
3. **(I) Exploration:**
4. \( f_{\text{expl}} = 1; \)
5. while \( i/N \leq f_{\text{expl}} \) do
6. \( i += 1; \)
7. draw new sample \( x_i \sim \pi(x); \)
8. evaluate sample \( y_i = u(x_i); \)
9. if \( y_i \in T \) then
10. count hits \( + = 1; \)
11. \( x_T \leftarrow x_i \) (add hit to the found collection of hits);
12. update estimate \( f_{\text{expl}} \) iteratively using Eq. 18
13. \( z_1 = E_x[R_T] \) and \( z_2 = 1/(f_{\text{expl}} T); \)
14. end
15. end
16. 
17. **(II) Adaptation:**
18. set \( \mu = x_T; \)
19. Calculate \( \Sigma \) by determining \( \sigma_{i,k}(x_k) \) for all \( k = 1, ..., N_T; \)
20. This gives \( q(x; \mu, \Sigma); \)
21. 
22. **(III) Refinement:**
23. while \( N_{\text{expl}} \leq i \leq N \) do
24. \( i += 1; \)
25. draw new sample \( x_i \sim q(x; \mu, \Sigma) \Phi_2(x); \)
26. evaluate sample \( y_i = u(x_i); \)
27. end
28. 
29. **Post processing:**
30. calculate \( F_{\text{expl}} \) and mixture weights \( \tilde{w}_i \sim \frac{\pi(x_i)}{q(x_i)}, \)
31. calculate desired population quantities such as the rate \( R_T \).

**Figure 2.** Top panel: the fraction of total samples that should be spent on the exploratory phase versus the fractional rate, \( R_T \), of the target population in a simulation. Different curves show the effect of varying the total number of samples \( N \) in the simulation. Bottom panel: expected sampling uncertainty on the predicted event rate versus the fractional rate, for two different choices of \( N \). The dashed lines show the expected uncertainty from ‘traditional’ Monte Carlo sampling (i.e. \( \sqrt{N/R_T} \)). The solid lines show the minimum possible statistical uncertainty from STROOPWAFEL sampling, i.e. if the efficiency in the refinement phase is 1. In practice, the statistical rate uncertainty when using STROOPWAFEL will lie in the area shaded in grey. All panels: the coloured vertical bars indicate the fractional rate of the target populations, and the star symbols show the corresponding values of these parameters, for the six simulations with \( N = 10^6 \) described in Section 3.

### 2.3 Characteristic behaviour

Here we use the analytic derivations from Section 2.2 to illustrate the characteristics of our algorithm in idealized cases. This is intended to help users of STROOPWAFEL understand the expected behaviour without needing to master the details presented above.

Key variables for this illustration are:

(i) \( R_T \), the formation rate of the population under study. This is expressed as the fraction of binaries, when drawn from initial conditions following the birth probability distribution, that yield target systems.

(ii) \( N \), the total number of binary systems (i.e. samples) used in a given population simulation, which is chosen by the user.

(iii) \( f_{\text{expl}} \), the fraction of the total number of samples that should optimally be spent on the exploration phase. This is chosen automatically by STROOPWAFEL during the exploration phase (see Section 2.4.2), when the algorithm estimates the formation rate \( R_T \).

Fig. 2 presents derived quantities as a function of the fractional rate of the target population \( R_T \). This figure, and Fig. 3, includes
Comparisons to observational data will typically be made using distribution functions of predicted quantities (e.g. component masses), not just event rates. We later demonstrate the improvements provided by our algorithm for predictions of distribution functions. None the less, this overall decrease in statistical rate uncertainty for fixed sample number in a simulation is indicative of the improvements enabled by applying STROOPWAFEL to a target population.

Fig. 3 shows the increase in the number of simulated binaries of interest versus traditional Monte Carlo sampling from a birth distribution for a simulation with fixed \( N = 10^5 \). The efficiency in the refinement phase is not known in advance. We show predictions for a refinement-phase efficiency of 1 and 0.1; as long as the total number of successful samples is dominated by those drawn during the refinement phase, the maximum possible gain is roughly proportional to the refinement-phase efficiency. The value for the efficiency of the refinement phase varies between \( \sim 3.4 \cdot 10^{-2} \) and \( 3.7 \cdot 10^{-1} \) in our example astrophysical simulations (see Section 3 and Table 2).

### 3 RESULTS

In this section, we demonstrate the power and advantages of STROOPWAFEL. Our algorithm could be applied to many sampling routines, but the illustration here uses the binary population synthesis code COMPAS (Stevenson et al. 2017; Barrett et al. 2018; Vigna-Gómez et al. 2018). The physical assumptions and parameter settings we adopt are briefly summarized in Appendix C.

We combine STROOPWAFEL and COMPAS to model six different target populations. Four are simulations of subtypes of DCOs that merge in a Hubble time: (1) all DCO mergers (i.e. BH–NS, NS–NS, and BH–BH), (2) BH–BH mergers, (3) NS–NS mergers, and (4) BH–NS mergers. Additionally, we model two simulations of extremely rare events by focusing on a subset of the above, namely (5) BH–BH mergers with total system masses in excess of \( m_{\text{tot}} \geq 50 M_{\odot} \) and (6) NS–NS mergers that merge within \( t_c \leq 50 \) Myr from the moment of the DCO formation (where \( t_c \) is the coalescence time). A summary of the results for these simulations can be found in Table 2.

In this section, we present detailed results from simulations 1 to 4, as these target populations are those most commonly discussed in the literature. We just present the key findings for simulations 5 and 6. For each target population, we compare a simulation using our sampling algorithm to one that uses birth distribution Monte Carlo sampling, which for conciseness we will typically call traditional sampling. Both the STROOPWAFEL and traditional simulations sample \( N = 10^6 \) initial binaries.

The overall gain that is obtained when using STROOPWAFEL depends on the simulation and the initial efficiency of the ‘traditional’ method that the algorithm is compared with. For example, the choices for the initial parameter space can change how much the sampling can be improved when using STROOPWAFEL. We use settings that are commonly used in population synthesis studies of DCO mergers.

The remainder of the section is structured as follows. Section 3.1 demonstrates the increased efficiency of STROOPWAFEL at finding binaries from the target population. Section 3.2 discusses how that increased efficiency can be used to speed up simulations. Section 3.3 shows how our algorithm produces better resolution of the target population. Section 3.4 describes how our sampling method leads to smaller statistical uncertainties in predicted population distribution functions. Section 3.5 shows how STROOPWAFEL
Table 2. Summary of the results from six target populations that are modelled in this paper to demonstrate our STROOPWAFEL algorithm. We list the fraction of samples spent in the exploratory phase, $f_{\text{expl}}$, and the efficiency of finding ‘hits’ in the exploratory and refinement phases. The gain in refinement is the ratio between the efficiency of finding samples of the target population during the refinement phase of STROOPWAFEL and traditional sampling (where the efficiency of traditional sampling is equal to the efficiency of the STROOPWAFEL exploratory phase). $N_T$traditional and $N_T$STROOPWAFEL represent the total number of systems of interest that are found by the end of the simulation (using a total of $10^6$ samples). The last column is the overall gain that we found when using STROOPWAFEL compared to traditional Monte Carlo sampling from the birth distributions, which is defined by the ratio $N_T$STROOPWAFEL/$N_T$traditional.

<table>
<thead>
<tr>
<th>nr</th>
<th>Target subpopulation</th>
<th>$f_{\text{expl}}$</th>
<th>Efficiency exploratory</th>
<th>Efficiency refinement</th>
<th>Gain refinement</th>
<th>$N_T$traditional</th>
<th>$N_T$STROOPWAFEL</th>
<th>Gain overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>All DCO mergers in a Hubble time</td>
<td>0.23</td>
<td>6.78 · 10^{-3}</td>
<td>3.05 · 10^{-1}</td>
<td>45×</td>
<td>6.71 · 10^{-1}</td>
<td>2.35 · 10^{-5}</td>
<td>35×</td>
</tr>
<tr>
<td>2</td>
<td>BH–BH mergers in a Hubble time</td>
<td>0.27</td>
<td>5.25 · 10^{-3}</td>
<td>3.69 · 10^{-1}</td>
<td>70×</td>
<td>5.16 · 10^{-1}</td>
<td>2.71 · 10^{-5}</td>
<td>53×</td>
</tr>
<tr>
<td>3</td>
<td>BH–NS mergers in a Hubble time</td>
<td>0.66</td>
<td>6.36 · 10^{-4}</td>
<td>7.38 · 10^{-2}</td>
<td>116×</td>
<td>6.55 · 10^{-2}</td>
<td>2.55 · 10^{-4}</td>
<td>39×</td>
</tr>
<tr>
<td>4</td>
<td>NS–NS mergers in a Hubble time</td>
<td>0.59</td>
<td>9.03 · 10^{-4}</td>
<td>9.71 · 10^{-2}</td>
<td>108×</td>
<td>8.93 · 10^{-2}</td>
<td>4.00 · 10^{-4}</td>
<td>45×</td>
</tr>
<tr>
<td>5</td>
<td>BH–BH mergers $m_{\text{tot}} \geq 50 M_\odot$</td>
<td>0.69</td>
<td>5.45 · 10^{-4}</td>
<td>3.55 · 10^{-1}</td>
<td>651×</td>
<td>5.44 · 10^{-2}</td>
<td>1.10 · 10^{-5}</td>
<td>203×</td>
</tr>
<tr>
<td>6</td>
<td>NS–NS mergers with $t_c \leq 50 \text{Myr}$</td>
<td>0.77</td>
<td>3.43 · 10^{-4}</td>
<td>3.38 · 10^{-2}</td>
<td>99×</td>
<td>3.32 · 10^{-2}</td>
<td>7.95 · 10^{-5}</td>
<td>24×</td>
</tr>
</tbody>
</table>

becomes even more important for recovering tails of distribution functions and when considering observational bias. Section 3.6 discusses how STROOPWAFEL handles well the bifurcations and discontinuities in the binary population synthesis parameter space.

### 3.1 On the gain of generating binaries of the target distribution

We find that the number of binaries of these target populations increases by factors of about 25–200 when using our STROOPWAFEL sampling algorithm compared to simulations with traditional sampling. The panels in Fig. 4 showcase this by presenting the number of systems formed from the target population as a function of total number of sampled binaries, for both our sampling method and traditional birth distribution Monte Carlo sampling. For these four target populations, the gains are between $\sim$35 and 55. For the two additional extremely rare target populations, we find gains of 24 and 203. The gains are also shown in the last column of Table 2, and Fig. 3.

At the beginning of each simulation, during the STROOPWAFEL exploratory phase, the two sampling methods produce similar number of binaries of interest (i.e. hits), only different by random chance. The duration of that exploratory phase is determined by $f_{\text{expl}}$, as derived in Section 2.2.4. For our simulated target populations, using $N = 10^6$ samples, $f_{\text{expl}}$ ranges between $\approx 0.2$ and 0.8 (see $f_{\text{expl}}$ in Table 2). The algorithm then switches to the more focused refinement phase, using the information from the hits found during the exploratory phase. During this refinement phase, our sampling algorithm is 45–650 times more efficient at finding hits (see the sixth column in Table 2).

The difference in efficiency gains between the populations originates mostly from two effects. First, the different rareness of the target populations (e.g. for these assumptions, BH–NS mergers are more rarely produced than BH–BH mergers) influences how much the efficiency increases during the refinement phase and also the duration of the exploratory phase. Both are important factors for determining the overall gain in efficiency. Secondly, the structure of the output surfaces influences how well the Gaussian mixture distribution covers the regions of interest in the output space. A more stochastic or discontinuous output space (e.g. many small islands of hits) will lead to smaller efficiencies in the refinement phase of STROOPWAFEL. This effect is most noticeable in the different gains between the BH–BH systems with total masses over 50 $M_\odot$ and the NS–NS systems that merge within 50 Myr.

Supernova remnants receive a random natal kick in our simulations. These kicks induce stochastic and discontinuous behaviour into the output surfaces, leading to lower refinement-phase efficiency and gain. The kicks are typically larger for NSs than BHs, and so affect NS–NS simulations the most. Conversely, the gain is greatest for the BH–BH merger populations that are least affected by these stochastic kicks (as shown in Figs 2 and 3).

The largest overall gain shown in Table 2, and Fig. 3, is for modelling a BH–BH population with a total mass over 50 $M_\odot$. For this population, it would be possible to increase the efficiency of Monte Carlo sampling from the birth distributions by making thoughtful changes to the boundaries of the initial parameter space, so the factor of $\approx 200$ improvement we show for STROOPWAFEL is higher than would arise when comparing to more carefully targeted use of standard Monte Carlo sampling in this case. However, well-informed choices in the initial parameter space would also benefit STROOPWAFEL by increasing the efficiency of the initial exploration phase. Moreover, one-dimensional cuts to the parameter space would become increasingly inefficient when sampling in higher dimensions. STROOPWAFEL automatically finds the regions of interest, and avoids the risk of incorrect choices in restricting the initial parameter space.

The increase in the number of events decreases the sampling uncertainty in the predicted event rates. Although the standard uncertainty from Poisson noise decreases with the square root of the number of target systems found, i.e. as $1/\sqrt{N_T}$; in our weighted sampling case, it also depends on the variance in the weights [see equation (15)]. We find that our sampling algorithm results in 10.5 times smaller sampling uncertainties compared to traditional Monte Carlo sampling in this case. However, well-informed choices in the initial parameter space, so the factor of $\approx 200$ improvement we show for STROOPWAFEL is higher than would arise when comparing to more carefully targeted use of standard Monte Carlo sampling in this case. However, well-informed choices in the initial parameter space would also benefit STROOPWAFEL by increasing the efficiency of the initial exploration phase. Moreover, one-dimensional cuts to the parameter space would become increasingly inefficient when sampling in higher dimensions. STROOPWAFEL automatically finds the regions of interest, and avoids the risk of incorrect choices in restricting the initial parameter space.

### 3.2 Speeding up simulations

Instead of using STROOPWAFEL to obtain more information from a simulation with the same number of samples, one could alternatively aim for a certain precision in the predicted event rates. In that case, STROOPWAFEL can be used to speed up the simulation, since this precision will be reached using a fraction of the number of samples required when using traditional sampling. Traditional sampling would require 25–200 more simulations than STROOPWAFEL to achieve the same number of target binaries, and a factor of around 10–100 times more simulations to achieve the
The number of simulated binaries $N_T$ falling into the target population as a function of the total number of binaries $N_{\text{binaries}}$ sampled for the traditional sampling method (grey dashed line) and the sampling method presented in this study (solid coloured line). The four panels show the simulations for each of the four target subpopulations. In each panel, the duration of the exploratory phase is shown with a hashed grey area. In the background, the standard Poisson fractional uncertainties of 0.3, 1, and 3 per cent are shown with dashed lines.

The speed-up factor further depends on the computational cost of simulating samples from the chosen distribution. It might be that the binaries of interest require more or less computational time than other binaries. Therefore, the speed-up when using the adaptive distribution $Q$ depends on the science case of interest. In the simulations performed for this study, the average computational cost (in CPU time) of simulating typical individual binaries sampled from the adaptive distribution $Q$ was up to a factor of 2 smaller than for individual binaries sampled from the birth distribution. Therefore, the total speed-up was up to another factor of 2 larger in our simulations than from more efficient sampling alone.

More generally, we note that the gain or relative speed-up from using STROOPWAFEL will depend on the target population and the traditional method with which it is compared. First of all, the speed-up from STROOPWAFEL will generally be greater (smaller) if the target population is more (less) rare. This is shown in Fig. 3.

Equivalently, if one chooses a larger initial parameter space (e.g. sampling $m_{1,i}$ from [1, 150]$M_\odot$ instead of $m_{1,i}$ from [5, 150]$M_\odot$ used here), the gain would have been larger as the event of interest becomes rarer (assuming no binaries in the extended range form a binary of the target population). Second, in some binary population synthesis studies the primary mass is sampled uniformly in $\log m_{1,i}$ space. This is a form of importance sampling. The gain of using STROOPWAFEL (with uniform sampling in $\log m_{1,i}$ during the exploratory phase) could be lower than gains without this importance sampling if importance sampling makes the traditional Monte Carlo more efficient. Nevertheless, we would still expect a significant gain from STROOPWAFEL – especially if using that form of importance sampling in the exploratory phase significantly decreases the duration of this phase.

3.3 Mapping the parameter space with higher resolution

The increase in computational efficiency from STROOPWAFEL leads to finding substantially more events of the target population,
which naturally enables a much higher resolution mapping of both
the input and output parameter spaces. Figs 6 and 7 show examples
of how the parameter space is explored in far greater detail using our
sampling method compared to traditional birth distribution Monte
Carlo sampling.

Fig. 6 shows the location of the target population in the initial
parameter space of primary mass $m_{1,f}$ and separation log $a_i$ at
birth. With our sampling method, we obtain more detailed contours and more contour levels that map the initial parameter space with
higher resolution. This leads to better knowledge of the initial
conditions of a binary system that yield a binary of the target
population. Physically, the structures seen in the input parameter
space correspond to the assumed physics of the different formation
channels leading to compact-object mergers. More details are

Meanwhile, Fig. 7 shows the higher resolution mapping from
STROOPWAFEL for the output space of the final masses of the
compact objects $m_{1,c}$ and $m_{2,c}$ in each DCO. We plot on top the
gravitational-wave events found from O1 and O2 data from The
LIGO Scientific Collaboration (2018) and Zackay et al. (2019).4

The simulations with our STROOPWAFEL algorithm again yield
higher resolutions and more systems of the target populations in
the regions overlapping with the observations. This is important
in order to compare observations and theory and test the physical
assumptions in our models.

Fig. 7 shows that even in the STROOPWAFEL simulation, there
are relatively few samples consistent with the 90 per cent credible
regions for some of the highest mass gravitational-wave

\begin{align*}
\text{Chirp mass} & = \frac{m_{1,f} m_{2,f}}{(m_{1,f} + m_{2,f})^{3/5}} \\
\text{where} & = \frac{(m_{1,c} m_{2,c})^{3/5}(m_{1,c} + m_{2,c})^{1/5}}{m_{1,c} + m_{2,c}}
\end{align*}

4Publicly available data can be found at https://www.gw-openscience.org/catalog/GWTC-1-confident/html.
Figure 6. Contour plots of the locations in log $m_{i,1}$ and log $a_i$ space of the hits $x_T$ (i.e. binaries of the target population) found in each simulation when using the traditional birth distribution Monte Carlo sampling (left-hand panels) and the sampling method STROOPWAFEL developed in this study (right-hand panels). Contours represent a constant density of binaries of the target population found per unit area in log $m_{i,1}$ – log $a_i$ space. The colour gradient indicates the number of samples per area $\Delta S$, the size of which is shown with a black rectangle. If the density is below the level of our lowest contour, we plot the individual points. The four different panels from top to bottom represent the first four target populations shown in Table 2. The total number of hits $N_T$ found in each simulation is quoted in parentheses. The metallicity assumed in all simulations is $Z = 0.001$. 
Figure 7. Similar to Fig. 6 but now for the output parameters: the final compact-object masses $m_{1,f}$ and $m_{2,f}$ of the DCO. We overplot the gravitational-wave observations from O1 and O2 from The LIGO Scientific Collaboration (2018) in black and from Zackay et al. (2019) in red. Error bars indicate 90 per cent credible regions around the median. The metallicity assumed in all simulations is $Z = 0.001$; selection effects of gravitational-wave detectors are not accounted for.
amplitudes, overemphasize the rare and frequently undersampled tails of the simulated distributions. An example is shown in Fig. 9, where we plot the predicted distribution of chirp masses for BH–NS systems estimated using traditional birth distribution Monte Carlo or by using the STROOPWAFEL algorithm. The shown distributions are weighted by the sensitivity of gravitational-wave interferometers, approximated as a bias dependent on the primary DCO mass $m_{1}^{2}$ (Fishbach & Holz 2017). We also show $1\sigma$ and $2\sigma$ confidence intervals that are calculated by bootstrapping the samples 1000 times. Our algorithm produces much smoother distribution predictions with much smaller sampling uncertainties compared to traditional sampling methods for the same number of samples simulated. In particular, Fig. 9 demonstrates that simulations using the traditional Monte Carlo sampling from the birth distributions undersample the high-mass end of the population. This will be particularly significant when comparing population models to observations.

Fig. 9 corresponds to a particular model choice; variations of the model have to be considered in order to compare with observations. The displayed distribution is from a simulation at a single metallicity of $Z = 0.001$, while a range of metallicities will contribute to the observed BH–NS merger population. An integration over the metallicity-dependent cosmic star formation history is therefore required. The properties of BH–NS mergers will be explored with COMPAS in future work.

4 DISCUSSION

We have demonstrated that the performance of STROOPWAFEL is substantially superior to traditional Monte Carlo sampling from the birth probability distributions. For the types of rare events simulated in Section 3, the gain is already so large that the current implementation of our algorithm can contribute to drastic speed-ups of binary population synthesis simulations. Hence, we have postponed some natural improvements to STROOPWAFEL until later, but we discuss them here. After those, we discuss additional potential applications for our algorithm.

4.1 The exploratory phase

During the exploratory phase in STROOPWAFEL, the initial parameter space is sampled by drawing random binaries from the priors (as in traditional birth distribution Monte Carlo sampling) until $N_{\text{expl}}$ events of interest are found. There are several features of the exploratory phase that could be optimized and improved.

(i) We now use sampling from the birth distribution $\pi$ for drawing the random binaries in the exploratory phase. Future improvements of STROOPWAFEL could use more efficient sampling algorithms in the exploratory phase. Examples include (1) using importance sampling in the exploratory phase when there is an existing guess at a more efficient sampling distribution, or (2) implementing techniques such as Latin hypercube sampling (LHS; Eglajs & Audze 1977; McKay, Beckman & Conover 1979; Iman, Davenport & Zeigler 1980; Iman, Helton & Campbell 1981). LHS is a Monte Carlo method that generates near-random samples that are more equally distributed throughout the initial parameter space by placing
Figure 9. Predicted distribution of the chirp mass of the merging BH–NS population using STROOPWAFEL (green) and traditional (grey) sampling. In both the cases, the simulation uses $N = 10^5$ samples and the distributions are weighted by the sensitivity of gravitational-wave interferometers using Fishbach & Holz (2017). The shaded regions show the $1\sigma$ and $2\sigma$ confidence intervals that are calculated by bootstrapping the samples 1000 times. This distribution is for a particular set of model assumptions, including a single metallicity $Z = 0.001$, and an integration over a metallicity-dependent cosmic star formation history is required for comparisons with observations. The same *scipy* kernel density estimator smoothing with a dimensionless kernel density estimator factor of about 0.1 is used for traditional and STROOPWAFEL distributions (see also Appendix D).

only one sample in each row and column of the Latin square. By doing so, it could improve the sampling in the exploratory phase as the probability of the randomly drawn samples being clustered decreases slightly.

(ii) The duration of the exploratory phase is now determined with $f_{\text{expl}}$, which is optimized for the uncertainties on the rates of the target distribution. A future improvement would be to determine $f_{\text{expl}}$ based on the uncertainty in the simulated output distribution function. See also Section 4.3.

(iii) The exploratory phase duration is optimized under the simplifying assumption that the instrumental distribution will match the target distribution except for some missing regions in parameter space. The optimization could also consider the level of fluctuation in the instrumental sampling distribution (i.e. the variance in the weights).

(iv) If the structures in the parameter space have a known minimum volume, we could use this to derive a better-informed estimate for the uncertainty contributing from the probability of missing such structures in the exploratory phase. This seems unlikely to apply to binary-star population synthesis, but might be relevant for other applications of STROOPWAFEL.

4.2 The refined sampling phase

The Gaussians that are used to form the instrumental distribution are currently constructed using diagonal covariances ($\Sigma_k$). These then remain unchanged throughout the refinement phase of AIS – even though much more information becomes available about the distribution of hits in the initial parameter space. A potential future improvement is to update the instrumental sampling distribution during the refinement phase. In principle, this might be done locally, with only the samples drawn from each of the individual Gaussians used to update the corresponding element of the instrumental distribution. Doing so would avoid a potentially expensive nearest neighbour search, as the tree is automatically built for free by the sampling already being performed. See also, for example, the AMIS algorithm described in Cornuet et al. (2012). Adaptive distribution choices beyond a mixture of Gaussians could also be explored.

4.3 Adapting to uncertainty in distribution functions

Observational selection effects must be applied to model predictions in order to statistically compare models against observations. These selection effects are generally applied after population synthesis models are generated, and may place significant weight on rarely formed systems in the tails of output distribution functions (e.g. higher mass DCO binaries). Even though STROOPWAFEL can
greatly improve the overall number of systems produced from a simulation, there may still be relatively few systems in these tails.

In principle, we could tune STROOPWAFEL to produce a model population weighted towards any observational population distribution, i.e. optimizing for observational selection effects and spending less time on systems that do not contribute to the observed sample. This can be achieved by incorporating selection effects directly in the instrumental distribution rather than applying them to STROOPWAFEL outputs. This can be practically implemented by changing the instrumental distribution weights. At the moment, all Gaussians contribute equally to the mixture distribution with a weight $1/N_{\text{exp}}$ but instead the contribution of each Gaussian can be weighted with the probability of observing the system to focus the simulation on systems that are more likely to contribute to the observable population. An extreme example of this approach would be redefining the target population to be an even rarer subset of the initial target population, e.g. tails of a distribution function. STROOPWAFEL could also be used to sample from regions of the initial parameter space giving rise to properties consistent with specific observed systems (see also Andrews et al. 2018).

The current implementation of STROOPWAFEL might be thought of as something like adaptive mesh refinement, familiar from hydrodynamics, applied to the phase space of binary population synthesis. This potential future development of STROOPWAFEL would be refining in the space of predicted observables. This development of STROOPWAFEL could naturally be applied to modelling any population for comparison to observations, not only intrinsically rare populations.

4.4 STROOPWAFEL in higher dimensions

The demonstrations in this paper have all used STROOPWAFEL to sample in a three-dimensional birth parameter space of the two component masses and the initial orbital separation. STROOPWAFEL can be readily applied to sample in more dimensions. The scaling parameter $\kappa$ may well have a different optimal value for higher dimensions, and should be investigated before being applied to higher dimensions, although we anticipate moving away from using diagonal covariances (see Section 4.2). Additional potential dimensions to add to the space of initial conditions include, e.g. initial compositions, the initial eccentricity of the system, or the spins of the stars. Moreover, in COMPAS, systems are labelled from the start with vectors representing normalized versions of the supernova kicks that will be applied during compact-object formation (see also e.g. Andrews et al. 2018); each kick adds three dimensions to the parameter space. Potentially, applying STROOPWAFEL to the kick vectors can be promising since the kick magnitudes and directions can significantly affect the fates of the systems. In our simulations, this would be especially important to increase the gain in simulating NS–NS mergers, as the kicks contribute most to the current stochastic output surfaces for this target population.

4.5 Combining STROOPWAFEL with MCMC or Gaussian process regression emulators on continuous spaces

STROOPWAFEL could be applied directly to the combined parameter space of initial parameters of an individual system (e.g. the initial masses and separations) and hyper-parameters describing the model assumptions (e.g. wind-driven mass-loss rates, common-envelope physics).

Alternatively, STROOPWAFEL could be combined with other methods for exploring the parameter space, such as emulators based on Gaussian process regression (see e.g. Barrett et al. 2017; Taylor & Gerosa 2018). Some of the parameters map continuously to the output space, while others exhibit discontinuities. STROOPWAFEL distinguishes itself in handling bifurcations and stochastic output surfaces. On the other hand, emulators can be more efficient in sampling parameters that are smooth. Thus, an intended future development is to combine STROOPWAFEL with such methods to obtain the best overall efficiency.

STROOPWAFEL output samples could also be converted into probability distributions using Gaussian mixture models based on Dirichlet processes (Del Pozzo et al. 2018).

5 SUMMARY AND CONCLUSIONS

We have presented a new sampling algorithm that aims to improve the efficiency of simulating rare events in astrophysical populations, and demonstrated its utility for binary population synthesis of gravitational-wave merger populations. Our algorithm STROOPWAFEL adaptively improves the sampling distribution to focus more computational time on the target population. Some key findings of our investigation are:

(i) Using STROOPWAFEL, we find a factor of about $25–200\times$ more systems of interest in simulations of a certain length, as compared to Monte Carlo sampling from the birth distributions. To simulate the same number of events of interest with such commonly used Monte Carlo sampling would require up to two orders of magnitude more computational time. This gain will improve binary population synthesis simulations by making it computationally feasible both to include more details of the relevant massive-star physics and to explore a greater number of variations of the physical assumptions of the model.

(ii) The increase in efficiency of STROOPWAFEL leads to higher resolution mapping of both the input and output parameter spaces. This reduces the sampling uncertainty by factors of $\approx 3–10$ for our simulations with $10^6$ total samples.

(iii) STROOPWAFEL improvements are particularly significant when simulating extremely rare events or tails of distribution functions, such as the most massive BH–BH mergers or early NS–NS mergers.

(iv) One of the core strengths of STROOPWAFEL is that it can handle well the bifurcations and discontinuities that naturally occur in the parameter spaces of binary population synthesis simulations. Such stochasticity often poses a challenge for applying sampling and emulation methods such as MCMC and Gaussian process regression emulators that rely on smoothness to converge and produce independent samples.

Future improvements to the STROOPWAFEL algorithm (discussed in Section 4) should be able to further improve its performance. This could make it more realistic for next-generation binary population synthesis simulations to include detailed stellar evolution models while also exploring more variations in the model physics and assumptions. Such improvements will help in comparing population models to population data, and so help to constrain the physics of evolutionary processes occurring on timescales too long to directly observe.
ACKNOWLEDGEMENTS
We thank especially T. Fragos, D. Szécsi, and M. Renzo for constructive discussions and comments. We also thank C. Berry, R. Willcox, and M. Wassink for their helpful suggestions on this paper. We thank D. Stops for technical support. We thank the anonymous referees for their helpful comments and careful feedback on this manuscript. FSB, AVG, IM, SJ, and JG thank the Kavli Foundation, Niels Bohr Institute, and DARK Cosmology Centre in Copenhagen for their hospitality and for organizing the Kavli summer school in gravitational-wave astrophysics 2017 where part of this work has been performed. The work has been performed under the Project HPC-EUROPA3 (INFRAIA-2016-1-730897), with the support of the EC Research Innovation Action under the H2020 Programme; in particular, FSB gratefully acknowledges the support of the School of Mathematics, University of Edinburgh and Birmingham Institute for Gravitational Wave Astronomy, University of Birmingham and the computer resources and technical support provided by EPCC. FSB also acknowledges support through the McKinsey excellence grant and Kapteyn grant. SJ and IM thank the Aspen Center for Physics for hospitality while part of this work was performed, supported by National Science Foundation grant PHY-1607611. SJ is further grateful for partial support of this work via a grant from the Simons Foundation. SDM and FSB acknowledge funding by the European Union’s Horizon 2020 research and innovation programme from the European Research Council (ERC) (grant agreement no. 715063), and by the Netherlands Organisation for Scientific Research (NWO) as part of the Vidi research program BinWaves with project number 639.042.728. SS is supported by the Australian Research Council Centre of Excellence for Gravitational Wave Discovery and by the Netherlands Organisation for Scientific Research (NWO) as part of the Vidi research program BinWaves with project number 639.042.728. SS is supported by the Australian Research Council Centre of Excellence for Gravitational Wave Discovery and by the Netherlands Organisation for Scientific Research (NWO) as part of the Vidi research program BinWaves with project number 639.042.728. SS is supported by the Australian Research Council Centre of Excellence for Gravitational Wave Discovery and by the Netherlands Organisation for Scientific Research (NWO) as part of the Vidi research program BinWaves with project number 639.042.728.

This research has made use of data, software, and/or web tools obtained from the Gravitational Wave Open Science Center (https://www.gw-openscience.org), a service of the LIGO Laboratory, the LIGO Scientific Collaboration, and the Virgo Collaboration. LIGO is funded by the U.S. National Science Foundation, Virgo is funded by the French Centre National de Recherche Scientifique (CNRS), the Italian Istituto Nazionale della Fisica Nucleare (INFN), and the Dutch Nikhef, with contributions by Polish and Hungarian institutes.

REFERENCES
Abbott B. P. et al., 2018, Living Rev. Relativ., 21, 3
Euler L., 1782, Euler Archive - All Works 530. Zeeuwsch Genootschap der Wetenschappen te Vlissingen, 9, 85
Hesterberg T., 1995, Technometrics, 37, 185
Jones N. D., 1977, DAIMI Report Series Vol. 6, Blindfold Games Are Harder Than Games with Perfect Information, p. 80
McKay M. D., Beckman R. J., Conover W. J., 1979, Technometrics, 21, 239
Opik E., 1924, Publ. Tartu Astrofizica Obs., 25, 1
APPENDIX A: DERIVATION OF THE VARIANCE USED FOR OPTIMIZING THE LENGTH OF THE STROOPWAFEL EXPLORATION PHASE

In this section, we derive the expression for the variance [equation (17)] on the rate estimate used to optimize \( f_{\text{expl}} \). We can estimate the optimal fraction \( f_{\text{expl}} \) of samples that we should spend in the exploratory phase by taking into account the probability of not identifying a target population forming region in the exploratory phase. We assume that we sample from the mixture distribution \( Q(x) \), which is given by equation (12):

\[
Q(x) = f_{\text{expl}} \pi(x) + (1 - f_{\text{expl}}) \tilde{q}(x),
\]

where \( \pi(x) \) is the prior (used for the exploratory phase sampling) and \( \tilde{q}(x) \) is the mixture of Gaussians. We also assume we aim to estimate the rate \( R_T \) with \( N \) total samples. After simulating these \( N \) samples, we may have identified a target binary-forming region with total weight \( z_1 \) whereas a region with weight \( z_2 \) is yet unidentified such that the estimated rate of the target population is [equation (16)]

\[
R_T = \begin{cases} 
\text{identified} & z_1 + z_2 \approx \mathbb{E}_Q[R_T], \\
\text{unidentified} & \end{cases}
\]

The variance on \( R_T \), \( \mathbb{V}_Q[R_T] \), is a measure of the uncertainty of the estimated rate \( \mathbb{E}_Q[R_T] \). Therefore, the optimal value of \( f_{\text{expl}} \) is one that minimizes the variance on \( R_T \). To determine this, we first derive an estimate for the variance \( \mathbb{V}_Q[R_T] \).

Using the continuous definition for the variance, we have

\[
\mathbb{V}_Q[R_T] = \frac{1}{N} \left[ \int \phi(x)^2 \tilde{w}(x)^2 Q(x) \, dx - \mathbb{E}_Q[R_T]^2 \right],
\]

(A3)

where the \( 1/N \) factor comes from taking the variance of the mean (average).

Since

\[
\tilde{w}(x) = \pi(x)/Q(x),
\]

and \( \mathbb{E}_Q[R_T] = z_1 + z_2 \), we find that

\[
\mathbb{V}_Q[R_T] = \frac{1}{N} \left[ \int \phi(x)^2 \tilde{w}(x)^2 \pi(x) \, dx - (z_1 + z_2)^2 \right].
\]

(A5)

Since we assumed that the target binary-forming region is equal to \( z_1 + z_2 \), by definition no binaries of the target population are found outside this region and thus \( \phi(x) = 0 \) outside \( z_1 \) and \( z_2 \). Using this, we can rewrite the integral in equation (A5) as

\[
\int \phi(x)^2 \tilde{w}(x)^2 \pi(x) \, dx = \int_{z_1} \tilde{w}(x)^2 \pi(x) \, dx + \int_{z_2} \tilde{w}(x)^2 \pi(x) \, dx.
\]

(A6)

We now assume that we have found enough binaries of our target population in our exploratory phase and that we do not have a bias for an output function such that the Gaussian mixture \( \tilde{q}(x) \) is effectively flat over \( z_1 \). In other words, we assume that on the target binary-forming region \( z_1 \)

\[
\tilde{q}(x) \approx \frac{\pi(x)}{z_1},
\]

(A7)

where \( \pi(x) \) is the birth distribution.

Using this in equation (A1), we approximate that \( Q(x) \approx (1 - f_{\text{expl}}) \pi(x)/z_1 + f_{\text{expl}} \pi(x) \) in \( z_1 \) such that

\[
\tilde{w}(x) \approx \frac{1}{(1 - f_{\text{expl}})/z_1 + f_{\text{expl}}} \quad \text{for } x \text{ in } z_1.
\]

(A8)

In addition, we also assume that the our Gaussian mixture \( \tilde{q} \) is negligible outside of the target binary-forming regions, i.e. \( q(x) = 0 \) outside of \( z_1 \). In other words, we assume that \( z_2 \) is far enough from \( z_1 \) that the probability is zero to sample it with \( \tilde{q} \), and that we have completely missed it during the exploratory sampling. By doing so, we obtain that on \( z_2 \) we have

\[
\tilde{w}(x) \approx \frac{1}{f_{\text{expl}}} \quad \text{for } x \text{ on } z_2.
\]

(A9)

Substituting equations (A8) and (A9) into the integral expression of equation (A6) then yields

\[
\int \phi(x)^2 \tilde{w}(x)^2 \pi(x) \, dx \approx \frac{z_1}{(1 - f_{\text{expl}})/z_1 + f_{\text{expl}}} + \frac{z_2}{f_{\text{expl}}}.
\]

(A10)

where we used \( \int_{z_1} \pi(x) \, dx = z_1 \) and \( \int_{z_2} \pi(x) \, dx = z_2 \).

We can now write the variance as

\[
\mathbb{V}_Q(R_T) \approx \frac{1}{N} \left[ \frac{z_1}{(1 - f_{\text{expl}})/z_1 + f_{\text{expl}}} + \frac{z_2}{f_{\text{expl}}} - (z_1 + z_2)^2 \right],
\]

(A11)

i.e. equation (17).

APPENDIX B: TOY MODEL

We construct a toy model that can be run without too much computational burden and is inspired by binary population synthesis.
simulations to study the performance of STROOPWAFEL. The advantage of a toy model is that the moments are analytically known and the toy model can be repeatedly evaluated at minimal computational cost. We use this toy model to investigate a suitable choice for the scale parameter \( \kappa \) in the width of Gaussian sampling distributions [see equation (11)].

We build the toy model in the three-dimensional parameter space defined by the initial parameters \( x_1, x_2, \) and \( x_3 \). The distribution functions (and ranges) of \( x_1, x_2, \) and \( x_3 \) are chosen to be similar to the initial parameters \( m_{1i}, a_i, \) and \( q_i \) used for our binary population synthesis model (see Appendix C). Similarly to the birth distribution of \( a_i \) in the binary population synthesis code, we sample in \( \log x_2 \). The output of the toy model is constructed from a union of disconnected volumes \( D \) and an output function \( \phi(x_i) \) given by

\[
\Pi_{D, \text{toymodel}}(x_i) = \begin{cases} 
1 & \text{if } x_i \in D \\
0 & \text{otherwise,} 
\end{cases}
\]

where \( D = D_0 \cup D_1 \cup D_2 \) is the union of three cuboids with the following vectors for the location of the centre and the length of each cuboid in the \( x_1, x_2, \) and \( x_3 \) direction

\[
\begin{align*}
D_0 &= [20, 34, 0.3] \pm [1.9, 8.0, 0.1], \\
D_1 &= [40, 1.0, 0.3] \pm [1.7, 0.6, 0.2], \\
D_2 &= [34, 7.0, 0.8] \pm [1.8, 0.6, 0.1].
\end{align*}
\]

The two-dimensional projected distribution of the union \( D \) in \( x_3 \) and \( \log x_2 \) is shown in bright green in Fig. B1. One might notice that the disconnected target regions look similar to the ‘boats’ in the game Battleship – which is often played with a strategy that is conceptually similar to STROOPWAFEL. The fractional rates produced by these regions are the integrals over their volumes of the local density of the birth probability distributions (the prior distribution). The fractional rate of \( D \) in the initial parameter space equals \( V_D = 0.0013127 \), where a prior (birth distribution) of \( x_1 \propto x_1^{-2.3} \) is assumed on \( x_1 \) and flat priors are assumed on \( \log x_2 \) and \( x_3 \). The value of \( V_D \) is chosen to be similar to the average yield of DCO mergers in the publicly available simulations\(^6\) of Vigna-Gómez et al. (2018). In addition, the fractional rate produced from \( D_0 \) is similar to that from \( D_1 \), whereas the fractional rate from \( D_2 \) is relatively 10 times smaller. (For these parameter scalings, the absolute volume of \( D_0 \) is larger than the volume of \( D_1 \), but the different prior distributions weight the volume of \( D_1 \) more highly.)

We run repeated simulations varying the parameter \( \kappa \) in STROOPWAFEL. We fix the total number of samples to \( N = 10^6 \). We know the true value for the volume integral \( V_D \) and calculate for each simulation the deviation between the fractional rate estimate and this true weighted volume. The closer to zero this deviation is, the better the estimate. For each variation of \( \kappa \), we run 100 simulations.

The result of one such simulation for \( \kappa = 2 \) is shown in Fig. B1. In the STROOPWAFEL simulation, the three islands are recovered with much better resolution than in traditional Monte Carlo sampling from the prior. The dark regions around the islands \( D_0, D_1, \) and \( D_2 \) in the STROOPWAFEL simulation demonstrate that our method focuses more of the computational time around the regions of interest. In both simulations, the islands \( D_0 \) and \( D_1 \) contain more samples than \( D_2 \), as expected.

Fig. B2 shows, in blue, the 1σ deviation from the true value for STROOPWAFEL simulations as a function of \( \kappa \). The result of repeated simulations with traditional birth distribution Monte Carlo sampling is shown as a reference on the right. The shades of green in the background show regions of 0.3, 1, and 3 per cent fractional sampling uncertainty on the rate estimate.

Excessively small values of \( \kappa \) lead to biases in the estimated rate of more than 10 per cent. This is because overly narrow Gaussian distributions create ‘holes’ in the adapted sampling distribution.

\(^6\)Populations are available at http://www.sr.bham.ac.uk/compas/data.
The approach we use to simulate a synthetic population of DCOs is similar to other binary population synthesis studies (including, e.g. Belczynski, Kalogera & Bulik 2002; Hurley, Tout & Pols 2002; Dominik et al. 2012). We evolve a population of binary systems from their birth until they form a DCO system or otherwise either merge or disrupt. We then make a subselection of the DCOs that consist of two compact objects that merge within the age of the Universe through gravitational-wave emission and study the properties of this population.

In general, we follow the fiducial model described in Vigna-Gómez et al. (2018). We mention the most important assumptions here. The birth distribution for the primary mass $m_{1,i}$ is chosen to be a power-law distribution known as the initial mass function (IMF) where $p(m_{1,i}) \propto m_{1,i}^{-2}$ with $\alpha = 2.3$ for massive stars (Kroupa 2001). For the simulations, we draw $m_{1,i}$ in $[5, 150] M_\odot$. The initial mass ratio $q_i = m_2/m_{1,i}$ of binary systems is suggested from observations to follow a flat distribution (e.g. Tout 1991; Mazeh et al. 1992; Goldberg & Mazeh 1994; Sana et al. 2012) given by $p(q_i) \propto 1$. We adopt $q_i \in [0, 1]$. The initial separation $a_i$ is assumed to be flat in the log, also known as Opik’s law $p(a_i) \propto \frac{1}{a}$ (Opik 1924; Aab 1983). We choose $a_i \in [0.01, 1000] a_u$. We assume that all our binaries have circular orbits at birth. These distributions and parameter ranges resemble commonly used settings for binary population synthesis simulations.

Our changes to the fiducial model from Vigna-Gómez et al. (2018) are the following:

(i) We use a metallicity of $Z = 0.001$ for all our simulations.
(ii) We use the DELAYED prescription for the core-collapse supernova treatment from Fryer et al. (2012).
(iii) We use a prescription for pair-instability supernovae and pulsational pair-instability supernovae based on Woosley (2017). The implementation in COMPAS is described in Stevenson et al. (2019).

We fix the total number of binaries in each simulation to $N = 10^6$ both for when using birth distribution Monte Carlo and STROOPWAFEL-sampled simulations. The total computational time for each of the birth distribution Monte Carlo simulations is approximately 180 CPU h. The total computational time of the STROOPWAFEL simulations is up to a factor of 2 lower as a result of a decrease in average simulation time per sample in our sampling method compared to traditional sampling (see Section 3.2). This result rises from binaries that become a gravitational-wave source costing on average less computational time to simulate with COMPAS than other binaries.

## APPENDIX D: BANDWIDTH VARIATIONS OF KERNEL DENSITY ESTIMATOR

Figs 8 and 9 use the same resolution (i.e. bandwidth or bin width) to estimate the chirp mass distribution function for the output of both the STROOPWAFEL and birth distribution Monte Carlo simulations. The bin width of a histogram or the bandwidth of a kernel density estimator can strongly influence the estimated distribution. Hence, in practice, the bandwidth should be adapted to the resolution available in the data.

We show in Fig. D1 the predicted chirp mass distribution of BH–BH mergers using adapted resolutions for birth distribution Monte Carlo sampling and STROOPWAFEL sampling.

We estimate the adapted bandwidth using *Scott’s Rule*, which, in one dimension, scales as $\propto N_{T}^{-1/4}$. This is the default bandwidth choice in the `scipy` kernel density estimator function. For the
Figure D1. Predicted chirp mass distribution of the BH–BH merger population using STROOPWAFEL (orange) and traditional (grey) sampling. In all cases, the simulation uses \( N = 10^6 \) samples and the distributions are weighted to the sensitivity of gravitational-wave interferometers using Fishbach & Holz (2017). The shaded regions show the 1\( \sigma \) and 2\( \sigma \) confidence intervals, which are calculated by bootstrapping the samples 100 times. This distribution is for a particular set of model assumptions, including a single metallicity \( Z = 0.001 \), and an integration over a metallicity-dependent cosmic star formation history is required for comparisons with observations.

In the bottom panel, a kernel bandwidth of about 0.044 is used for the STROOPWAFEL method, whereas for the traditional method we use a bandwidth of about 0.12.

STROOPWAFEL sampling, we replace \( N_T \) with the ESS given by
\[
\left( \sum \tilde{w}_i \right)^2 / \sum (\tilde{w}_i^2)
\]
(which, in practice, is approximately equal to \( N_T^{-1/4} \text{STROOPWAFEL} \)). The top panel of Fig. 9 shows the estimated chirp mass distribution from both sampling methods for a dimensionless kernel density estimator factor for the bandwidth of \( ESS^{-1/4} \text{STROOPWAFEL} \approx 0.044 \). This bandwidth is too small for the \( 53 \times \) smaller birth distribution Monte Carlo BH–BH population, which therefore shows significant statistical noise fluctuations. The middle panel of Fig. 9 shows the estimated chirp mass distribution from both the sampling methods for a bandwidth of \( N_T^{-1/4} \text{Traditional} \approx 0.12 \). This bandwidth causes smaller statistical fluctuations for the traditional sampling, but removes some of the more detailed features for the STROOPWAFEL sampled distribution. In the bottom panel, the two plots are combined, showing the distributions with the relative bandwidths. The STROOPWAFEL obtains smaller uncertainties on the distribution as well as a higher resolution, which is a result from the higher number of BH–BH mergers found in this simulation.

This paper has been typeset from a TeX/LaTeX file prepared by the author.