Fast Generation of Exchangeable Sequences of Clusters Data

Keith Levin[1*] and Brenda Betancourt[2]

[1*]Department of Statistics, University of Wisconsin–Madison, 1300 University Ave, Madison, Wisconsin, USA.
[2]Statistics & Data Science, NORC at the University of Chicago, 4350 East-West Highway, Bethesda, Maryland, USA.

*Corresponding author(s). E-mail(s): kdlevin@wisc.edu;
Contributing authors: betancourt-brenda@norc.org;

Abstract

Recent advances in Bayesian models for random partitions have led to the formulation and exploration of Exchangeable Sequences of Clusters (ESC) models. Under ESC models, it is the cluster sizes that are exchangeable, rather than the observations themselves. This property is particularly useful for obtaining microclustering behavior, whereby cluster sizes grow sublinearly in the number of observations, as is common in applications such as record linkage, sparse networks and genomics. Unfortunately, the exchangeable clusters property comes at the cost of projectivity. As a consequence, in contrast to more traditional Dirichlet Process or Pitman-Yor process mixture models, samples a priori from ESC models cannot be easily obtained in a sequential fashion and instead require the use of rejection or importance sampling. In this work, drawing on connections between ESC models and discrete renewal theory, we obtain closed-form expressions for certain ESC models and develop faster methods for generating samples a priori from these models compared with the existing state of the art. In the process, we establish analytical expressions for the distribution of the number of clusters under ESC models, which was unknown prior to this work.

Keywords: random partition, microclustering, Bell polynomials, renewal theory

1 Introduction

Random partitions are integral to a variety of Bayesian clustering methods, with applications to text analysis (Blei, 2012; Blei, Ng, & Jordan, 2003) genetics (Falush, Stephens, & Pritchard, 2003; Pritchard, Stephens, & Donnelly, 2000), entity resolution (Binette & Steorts, 2022) and community detection (Legramanti, Rigon, Durante, & Dunson, 2022), to name but a few. The most widely used random partition models are those based on Dirichlet processes and Pitman-Yor processes (Antoniak, 1974; Ishwaran & James, 2003; Sethuraman, 1994), most notably the famed Chinese Restaurant Process (CRP; Aldous, 1985). A drawback of these models is that they generate partitions in which one or more cells of the partition grows linearly in the number of observations n. This property is undesirable in applications to, for example, record linkage and social network modeling, where data commonly exhibit a large number of small clusters. For these applications, a different mechanism is needed that better captures the growth of cluster sizes with n.

The solution to this issue is to deploy models with the *microclustering* property, whereby the

size of the largest cluster grows sublinearly in the number of observations n. An early attempt in this direction appeared in Zanella et al. (2016). The authors were motivated by record linkage applications (Binette & Steorts, 2022) where clusters are expected to remain small even as the number of observations increases. This initial class of models, constructed under the Kolchin representation of Gibbs partitions (Kolchin, 1971), places a prior κ on the number of clusters K, and then draws from a distribution μ over cluster sizes conditional on K. This approach is comparatively simple, admitting an algorithm that facilitates sampling a priori and a posteriori similar to the CRP. Unfortunately, the distributions of the number of clusters and the size of a randomly chosen cluster are not straightforwardly related to the priors κ and μ. More to the point, it is not yet theoretically proven that this family of models indeed exhibits the microclustering property asymptotically.

More recently, Betancourt, Zanella, and Steorts (2022) considered a different approach to microclustering, called *Exchangeable Sequences of Clusters* (ESC) models. These models belong to the class of finitely exchangeable Gibbs partitions (Gnedin & Pitman, 2006; Pitman, 2006), named for the fact that the cluster sizes $S_1, S_2, \ldots$ are finitely exchangeable. An ESC model is specified by a distribution μ over cluster sizes (or a prior over such distributions), and a partition is generated by drawing cluster sizes independently from μ conditional on the event that these cluster sizes sum to n. That is, having specified a distribution μ on the positive integers, we draw cluster sizes $S_1, S_2, \ldots$ i.i.d. μ, conditional on the event

$$E_n = \left\{ \exists K : \sum_{j=1}^{K} S_j = n \right\}. \tag{1}$$

The advantage of this model is that the prior μ straightforwardly encodes a distribution over cluster sizes, in the sense that the size of a randomly chosen cluster is (in the large-n limit) distributed according to μ (Betancourt et al., 2022, Theorem 2). Furthermore, unlike the model proposed in Zanella et al. (2016), the microclustering property has been theoretically established for ESC models (Betancourt et al., 2022, Theorem 3). Indeed, to the best of our knowledge, ESC models remain the only random partition model for which the microclustering property has been rigorously established.

While ESC models are more interpretable and have better-developed theory than previously-proposed microclustering models, there is no known relationship between the cluster size distribution μ and the number of clusters K under these models. Recently, Natarajan, De Iorio, Heinecke, Mayer, and Glenn (to appear) (Proposition 2) established the distribution of the number of clusters K for the case where μ is a shifted negative binomial, one of the specific models first proposed by Betancourt et al. (2022). Bystrova, Arbel, King, and Deslandes (2020) established the behavior of K under a related class of Gibbs-type processes. Nonetheless, a general description of the behavior of K under ESC models remains open. Additionally, since ESC models require conditioning on E_n, previous approaches to sampling a priori amount to drawing repeatedly from μ and checking whether or not the cluster sizes $S_1, S_2, \ldots$ satisfy the condition in event E_n. We note that (approximate) sampling a posteriori from ESC models (rather than a priori) has relied on the "chaperones" algorithm (see Betancourt et al., 2022; Zanella et al., 2016), a modification of the well-known reseating algorithm TODO:cite, meant to ensure that clusters do not become empty during the reseating step. As suggested by an anonymous reviewer, this approach might be modified to produce (approximate) a priori samples. This would come at the expense of the bookkeeping overhead required by the chaperones algorithm, and would produce approximate rather than exact samples from the prior. As such, we do not pursue it here, but highlight it as an avenue for future work.

In this paper, we resolve both of the issues highlighted above by

1. Establishing analytic expressions for the distribution of the number of clusters under ESC models by relating the ESC generative process to known results in renewal theory and enumerative combinatorics.
2. Leveraging these connections with enumerative combinatorics to more efficiently sample from ESC models.

Apart from the prior works outlined above, the literature on the microclustering property is

scarce but diverse. Previous work includes models that sacrifice finite exchangeability to handle data with a temporal component (e.g., arrival times Di Benedetto, Caron, & Teh, 2021), general finite mixture models with constraints on cluster sizes (Jitta & Klami, 2018; Klami & Jitta, 2016; Silverman & Silverman, 2017), and models for sparse networks based on random partitions with power-law distributed cluster sizes (Bloem-Reddy, Foster, Mathieu, & Teh, 2018). Recently, Lee and Sang (2022) considered the question of *balance* in cluster sizes, as encoded by majorization of cluster size vectors. Clearly, this is an emergent area of research with a variety of applications for which efficient sampling methods are crucial.

2 Main Results

We begin by defining the ESC model more rigorously. Our goal is to generate a partition of $[n] = \{1, 2, \ldots, n\}$. Under the ESC model, this is done by first selecting a distribution μ on the positive integers (e.g., from a prior P_μ). As mentioned above, we may think of μ as encoding a distribution over cluster sizes, though this intuition is really only true asymptotically as $n \to \infty$. Having picked a distribution μ, the ESC model generates partition sizes by drawing $S_1, S_2, \ldots$ i.i.d. from μ, conditional on E_n defined in Equation (1), according to the following procedure:

1. Draw $\mu \sim P_\mu$
2. Draw $S_1, S_2, \ldots \overset{\text{i.i.d.}}{\sim} \mu$ conditional on E_n.
3. Define K_n to be the unique integer such that $\sum_{j=1}^{K_n} S_j = n$.
4. Assign the n observations to K_n clusters by randomly permuting the vector

$$(1, 1, \ldots, 1, 2, 2, \ldots, 2, \ldots, K_n, K_n, \ldots, K_n)$$

in which 1 appears S_1 times, 2 appears S_2 times, k appears S_k times, etc.

As discussed in the introduction, this model raises two key challenges. First, while μ naturally encodes the (asymptotic) cluster size distribution, it is not immediately clear how to relate the behavior of the number of clusters K_n to μ or to our prior P_μ. This raises a challenge for the purposes of interpretability and user-friendliness of the model. Second, generating samples a priori from this distribution is non-trivial, since one must condition on the event E_n that $\sum_{j=1}^{K_n} S_j = n$. We address both of these concerns by drawing on the connections between the ESC model, renewal theory and enumerative combinatorics.

2.1 Generating ESC Partitions

Let us consider the matter of generating partitions from ESC models. Betancourt et al. (2022) suggest drawing $S_1, S_2, \ldots$ i.i.d. according to μ until $\sum_{j=1}^{k} S_j \geq n$ for some $k \leq n$. If equality holds, then $(S_1, S_2, \ldots, S_k)$ is a valid sequence of cluster sizes (i.e., the event E_n holds), otherwise a new sequence is generated. Unfortunately, on average, this procedure must be repeated $1/\Pr[E_n \mid \mu]$ times before a valid sequence is generated. Thus, crucial to this approach is that $\Pr[E_n \mid \mu]$ be bounded away from zero for large n. This fact is established in Betancourt et al. (2022) for the case where μ has finite mean by identifying the cluster sizes $S_1, S_2, \ldots$ with the waiting times of a discrete renewal process and appealing to the following result (see, for example, Theorem 2.6 in Barbu & Limnios, 2009).

Lemma 1. *Let μ be a distribution on the positive integers with finite mean and generate $S_1, S_2, \ldots \overset{\text{i.i.d.}}{\sim} \mu$. With E_n as defined in Equation (1),*

$$\lim_{n \to \infty} \Pr[E_n \mid \mu] = \frac{1}{\mathbb{E}[S_1 \mid \mu]}.$$

Trouble arises in the event that $\mathbb{E}[S_1 \mid \mu]$ is large (or infinite), since then we may need to generate many samples $S_1, S_2, \ldots$ from μ before obtaining a usable sequence. To alleviate this issue and allow for the possibility that μ has infinite expectation, we propose an alternative approach to generating cluster sizes conditional on E_n. We begin by writing, for positive integers $s_1, s_2, \ldots$,

$$\begin{aligned} &\Pr\left[S_1 = s_1, S_2 = s_2, \cdots \mid E_n, \mu\right] \\ &= \Pr\left[S_1 = s_1 \mid E_n, \mu\right] \\ &\cdot \Pr\left[S_2 = s_2, S_3 = s_3, \cdots \mid S_1 = s_1, E_n, \mu\right]. \end{aligned} \tag{2}$$

Since E_n is precisely the event that there exists a k such that $\sum_{j=1}^{k} S_j = n$, we have

$$\begin{aligned}
&\Pr\left[E_n \mid S_1 = s_1, \mu\right] \\
&\quad = \Pr\left[\exists k : \sum_{j=2}^{k} S_j = n - s_1 \;\middle|\; \mu\right] \\
&\quad = \Pr\left[E_{n-s_1} \mid \mu\right],
\end{aligned}$$

from which we have

$$\begin{aligned}
&\Pr\left[S_1 = s_1 \mid E_n, \mu\right] \\
&\quad = \frac{\Pr\left[E_n \mid S_1 = s_1, \mu\right] \Pr[S_1 = s_1 \mid \mu]}{\Pr\left[E_n \mid \mu\right]} \\
&\quad = \frac{\Pr\left[E_{n-s_1} \mid \mu\right] \mu_{s_1}}{\Pr\left[E_n \mid \mu\right]}.
\end{aligned} \tag{3}$$

Since the variables $S_1, S_2, \ldots$ are drawn i.i.d.,

$$\begin{aligned}
&\Pr\left[S_2 = s_2, S_3 = s_3, \cdots \mid S_1 = s_1, E_n, \mu\right] \\
&\qquad = \Pr\left[S_1 = s_2, S_2 = s_3, \cdots \mid E_{n-s_1}, \mu\right].
\end{aligned}$$

Plugging this and Equation (3) into Equation (2), we have, for $1 \le s_1 \le n$,

$$\begin{aligned}
&\Pr\left[S_1 = s_1, S_2 = s_2, \cdots \mid E_n, \mu\right] \\
&\quad = \Pr\left[S_1 = s_2, S_2 = s_3, \cdots \mid E_{n-s_1}, \mu\right] \\
&\qquad \cdot \mu_{s_1} \Pr\left[E_{n-s_1} \mid \mu\right] / \Pr[E_n \mid \mu].
\end{aligned} \tag{4}$$

This equation suggests a recursive approach to generating cluster size sequences, which we formalize in Algorithm 1. Crucially, this algorithm avoids the runtime dependence on $\Pr[E_n \mid \mu]$ exhibited by the naïve sampling approach.

Theorem 2. *For any $s_1, s_2, \cdots \in [n]$ satisfying $\sum_{j=1}^{k} s_j = n$, the sequence $(X_1, X_2, \ldots, X_k)$ generated by Algorithm 1 satisfies*

$$\begin{aligned}
&\Pr\left[X_1 = s_1, X_2 = s_2, \ldots, X_k = s_k \mid \mu\right] \\
&\quad = \Pr\left[S_1 = s_1, S_2 = s_2, \ldots, S_k = s_k \mid E_n, \mu\right]
\end{aligned}$$

Proof. By construction of Algorithm 1, the variable m is initialized to $m \leftarrow n$, and thus

$$\Pr[X_1 = s_1 \mid \mu] = \frac{\mu_{s_1} \Pr[E_{n-s_1} \mid \mu]}{\Pr[E_n \mid \mu]}.$$

Algorithm 1 Given distribution $\mu = (\mu_n)_{n=1}^{\infty}$, generate $S_1, S_2, \cdots \mid E_n$.

1: Compute $\Pr[E_t \mid \mu]$ for $t = 1, 2, \ldots, n$.
2: $m \leftarrow n$; $\quad k \leftarrow 1$
3: **while** $m > 0$ **do**
4: Draw X_k according to

$$\Pr[X_k = s; m] = \frac{\mu_s \Pr[E_{m-s} \mid \mu]}{\Pr[E_m \mid \mu]}$$
$$s \in \{1, 2, \ldots, m\}$$

5: $m \leftarrow m - X_k$; $\quad k \leftarrow k + 1$
6: **end while**
7: Return $(X_1, X_2, \ldots, X_{k-1})$

It follows that

$$\begin{aligned}
&\Pr\left[X_1 = s_1, \ldots, X_k = s_k \mid \mu\right] \\
&\quad = \Pr[X_1 = s_1 \mid \mu] \\
&\qquad \cdot \Pr\left[X_2 = s_2, \ldots, X_k = s_k \mid X_1 = s_1, \mu\right] \\
&\quad = \Pr\left[X_2 = s_2, \ldots, X_k = s_k \mid X_1 = s_1, \mu\right] \\
&\qquad \cdot \mu_{s_1} \Pr[E_{n-s_1} \mid \mu] / \Pr[E_n \mid \mu].
\end{aligned}$$

Given $X_1 = s_1$, Algorithm 1 sets $m \leftarrow n - s_1$ and $k \leftarrow 2$, and draws X_2 according to

$$\Pr\left[X_2 = s_2 \mid X_1 = s_1, \mu\right] = \frac{\mu_{s_2} \Pr[E_{n-s_1-s_2} \mid \mu]}{\Pr[E_{n-s_1} \mid \mu]},$$

whence

$$\begin{aligned}
&\Pr\left[X_1 = s_1, \ldots, X_k = s_k \mid \mu\right] \\
&= \frac{\mu_{s_1} \Pr[E_{n-s_1} \mid \mu]}{\Pr[E_n \mid \mu]} \cdot \frac{\mu_{s_2} \Pr[E_{n-s_1-s_2} \mid \mu]}{\Pr[E_{n-s_1} \mid \mu]} \\
&\quad \cdot \Pr\left[X_3 = s_3, \ldots, X_k = s_k \mid X_1 = s_1, X_2 = s_2, \mu\right].
\end{aligned}$$

Repeating this argument, we have

$$\begin{aligned}
&\Pr\left[X_1 = s_1, \ldots, X_k = s_k \mid \mu\right] \\
&= \left(\prod_{j=1}^{k-1} \frac{\mu_{s_j} \Pr[E_{n-\sum_{t=1}^{j} s_t} \mid \mu]}{\Pr[E_{n-\sum_{t=1}^{j-1} s_t} \mid \mu]}\right) \\
&\quad \cdot \Pr[X_k = s_k \mid X_1 = s_1, \ldots, X_{k-1} = s_{k-1}, \mu] \\
&= \left(\prod_{j=1}^{k-1} \frac{\mu_{s_j} \Pr[E_{n-\sum_{t=1}^{j} s_t} \mid \mu]}{\Pr[E_{n-\sum_{t=1}^{j-1} s_t} \mid \mu]}\right) \\
&\quad \cdot \frac{\mu_{s_k} \Pr[E_0 \mid \mu]}{\Pr[E_{n-\sum_{t=1}^{k-1} s_t} \mid \mu]}.
\end{aligned}$$

Since $\Pr[E_0 \mid \mu] = 1$, we conclude that

$$\Pr\left[X_1 = s_1, \ldots, X_k = s_k \mid \mu\right] = \frac{\left(\prod_{j=1}^{k} \mu_{s_j}\right)}{\Pr[E_n \mid \mu]}.$$

On the other hand, repeated application of Equation (4) yields

$$\Pr[S_1 = s_1, S_2 = s_2, \ldots, S_k = s_k \mid E_n, \mu] = \frac{\left(\prod_{j=1}^{k} \mu_{s_j}\right)}{\Pr[E_n \mid \mu]},$$

which completes the proof. □

Remark 1 (Sampling a posteriori)**.** *Algorithm 1 generates a priori samples from an ESC, and it is natural to extend these ideas to sampling a posteriori. Unfortunately, sampling a posteriori from the ESC requires more complicated machinery (see, e.g., the "chaperones" algorithm; Betancourt et al., 2022; Zanella et al., 2016), and it is not obvious how to adapt the speedups exhibited by Algorithm 1 to these a posteriori sampling schemes. We anticipate that ideas like those introduced in this paper can be extended and applied to the problem of sampling a posteriori from ESC models, especially for choices of μ with "nice" structure that supports fast updates, but our focus in the present paper is on prior elicitation and calibration. We leave to future work the matter of extending these techniques to more general sampling problems.*

Algorithm 1 generates samples from an ESC model without the conditioning required by the method proposed in Betancourt et al. (2022), provided that we can compute $\Pr[E_n \mid \mu]$ for arbitrary choices of $n \geq 0$. Computation of this quantity requires the use of the k-th partial exponential Bell polynomial (Charalambides, 2002),

$$B_{n,k}(x_1, x_2, \ldots, x_{n-k+1}) = \sum_{j_1, j_2, \ldots, j_{n-k+1}} \frac{n! \prod_{i=1}^{n-k+1} \left(\frac{x_i}{i!}\right)^{j_i}}{j_1! j_2! \cdots j_{n-k+1}!}, \tag{5}$$

where the sum is over all non-negative integers $j_1, j_2, \ldots, j_{n-k+1}$ satisfying $\sum_{i=1}^{n-k+1} j_i = k$ and $\sum_{i=1}^{n-k+1} i j_i = n$.

Theorem 3. *Let μ be a probability distribution on the positive integers. Then*

$$\Pr[E_n \mid \mu] = \sum_{k=1}^{n} \frac{k!}{n!} B_{n,k}\Big(1!\mu_1, 2!\mu_2, \ldots, (n-k+1)!\mu_{n-k+1}\Big).$$

Proof. Viewing the cluster sizes $S_1, S_2, \ldots$ as the waiting times of a discrete-time renewal process (Barbu & Limnios, 2009), E_n corresponds to the event that a renewal occurs at time n. A key result from renewal theory relates $\Pr[E_n \mid \mu]$ and the cluster size distribution μ via their probability generating functions. Let $M(s)$ denote the ordinary generating function of the sequence $\mu = (\mu_n)_{n=1}^{\infty}$. That is, for $s \geq 0$,

$$M(s) = \sum_{k=0}^{\infty} \mu_k s^k,$$

where $\mu_0 = 0$ by assumption (i.e., in the language of renewal theory, waiting times are positive; in the language of the ESC model, there are no empty clusters). For each $n = 0, 1, 2, \ldots$, let $u_n = \Pr[E_n \mid \mu]$, with $u_0 = 1$ by convention (i.e., a renewal always occurs at time 0). Letting $U(s)$ be the ordinary generating function of the sequence $(u_n)_{n=0}^{\infty}$, one can show (see, e.g., Barbu & Limnios, 2009, Proposition 2.1) that for all $s \geq 0$,

$$U(s) = \sum_{k=0}^{\infty} u_k s^k = \frac{1}{1 - M(s)}. \tag{6}$$

This suggests a natural approach to computing $\Pr[E_n \mid \mu] = u_n$ using the fact that u_n can be determined from the n-th derivative of $U(s)$ evaluated at $s = 0$. Defining the functions $f(z) = 1/z$ and $g(s) = 1 - M(s)$, observe that for all $n = 0, 1, 2, \ldots$,

$$U^{(n)}(s) = \frac{d^n}{ds^n} f(g(s))$$

and for $n = 1, 2, \ldots$, we have

$$f^{(n)}(z) = \frac{(-1)^n n!}{z^{n+1}}, \quad \text{and} \quad g^{(n)}(s) = -M^{(n)}(s).$$

Applying Faá di Bruno's formula (Charalambides, 2002, Theorem 11.4),

$$\begin{aligned}U^{(n)}(s) &= \frac{d^n}{ds^n} f(g(s)) \\ &= \sum_{k=1}^{n} f^{(k)}(g(s)) \\ &\quad \cdot B_{n,k}\left(g'(s), g''(s), \ldots, g^{(n-k+1)}(s)\right) \\ &= \sum_{k=1}^{n} \frac{(-1)^k k!}{(1-M(s))^{k+1}} \\ &\quad \cdot B_{n,k}\Big(-M'(s), -M''(s), \ldots \\ &\qquad\qquad \ldots, -M^{(n-k+1)}(s)\Big),\end{aligned}$$

Using this identity along with the fact that $M^{(k)}(0) = k!\mu_k$, we have

$$\begin{aligned}u_n &= \frac{1}{n!} U^{(n)}(0) \\ &= \sum_{k=1}^{n} \frac{(-1)^k k!}{n!} B_{n,k}\Big(-1!\mu_1, -2!\mu_2, \ldots \\ &\qquad\qquad \ldots, -(n-k+1)!\mu_{n-k+1}\Big).\end{aligned}$$

A basic property of Bell polynomials (Charalambides, 2002, page 412) states that

$$\begin{aligned}&B_{n,k}\left(abx_1, a^2 b x_2, \ldots, a^{n-k+1} b x_{n-k+1}\right) \\ &\quad = a^n b^k B_{n,k}\left(x_1, x_2, \ldots, x_{n-k+1}\right).\end{aligned} \tag{7}$$

Setting $a = 1$ and $b = -1$, it follows that

$$\begin{aligned}u_n = \sum_{k=1}^{n} \frac{k!}{n!} B_{n,k}\Big(1!\mu_1, 2!\mu_2, \ldots \\ \ldots, (n-k+1)!\mu_{n-k+1}\Big),\end{aligned}$$

and the result follows by dividing by $n!$ and recalling that $u_n = \Pr[E_n \mid \mu]$. □

Theorem 3 allows us to compute $\Pr[E_n \mid \mu]$ for arbitrary choices of μ, which we now illustrate in the context of Poisson-distributed cluster sizes.

Example: ESC-Poisson.

Consider the case where $(\mu_n)_{n=0}^{\infty}$ is given by

$$\mu_k = \begin{cases} \frac{\lambda^{k-1} e^{-\lambda}}{(k-1)!} = \frac{k e^{-\lambda}}{\lambda} \frac{\lambda^k}{k!} & \text{if } k = 1, 2, \ldots \\ 0 & \text{if } k = 0. \end{cases}$$

That is, cluster sizes are shifted Poisson random variables. Applying Theorem 3,

$$\begin{aligned}&\Pr[E_n \mid \lambda] \\ &\quad = \sum_{k=1}^{n} \frac{k! e^{-k\lambda} \lambda^{n-k}}{n!} B_{n,k}\left(1, 2, \ldots, (n-k+1)\right),\end{aligned}$$

where we have used Equation (7). A basic identity (Comtet, 1974, page 135) states that

$$B_{n,k}\left(1, 2, \ldots, (n-k+1)\right) = \binom{n}{k} k^{n-k}. \tag{8}$$

Applying this identity, we conclude that

$$\begin{aligned}\Pr[E_n \mid \lambda] &= \sum_{k=1}^{n} \frac{e^{-k\lambda} (k\lambda)^{n-k}}{(n-k)!} \\ &= \sum_{k=1}^{n} \mathrm{Pois}(n-k; k\lambda),\end{aligned} \tag{9}$$

where $\mathrm{Pois}(\cdot; \lambda)$ denotes the probability mass function of a Poisson random variable with rate parameter λ.

2.2 Behavior of the number of clusters K_n

The number of clusters K_n is the (random) number k such that $\sum_{j=1}^{k} S_j = n$, again conditional on the event E_n to ensure that such a k exists. A natural choice under the ESC concerns the behavior of the random variable K_n.

Theorem 4. *Let $S_1, S_2, \ldots, S_{K_n}$ be cluster sizes generated according to an ESC model on n objects with cluster size distribution μ. Then for $k = 1, 2, \ldots, n$,*

$$\begin{aligned}&\Pr[K_n = k \mid E_n, \mu] \\ &\quad = \frac{k! B_{n,k}\left(1!\mu_1, 2!\mu_2, \ldots, (n-k+1)!\mu_{n-k+1}\right)}{n! \Pr[E_n \mid \mu]}.\end{aligned}$$

Proof. We begin by observing that $E_n = \cup_{k=1}^{n} \{K_n = k\}$, whence for $k = 1, 2, \ldots, n$,

$$\begin{aligned}
&\Pr[K_n = k \mid E_n, \mu] \\
&\quad = \sum_{s_1, s_2, \ldots, s_k} \frac{\Pr[S_1 = s_1, \ldots, S_k = s_k \mid \mu]}{\Pr[E_n \mid \mu]} \\
&\quad = \sum_{s_1, s_2, \ldots, s_k} \frac{\prod_{j=1}^{k} \mu_{s_j}}{\Pr[E_n \mid \mu]},
\end{aligned}$$

where the sum is over all $s_1, s_2, \ldots, s_k$ satisfying $\sum_{j=1}^{k} s_j = n$. Equivalently, using basic properties of partitions of $[n]$, we can express this sum as

$$\begin{aligned}
&\Pr[K_n = k \mid E_n, \mu] \\
&\quad = \frac{k!}{\Pr[E_n \mid \mu]} \sum_{j_1, j_2, \ldots, j_{n-k+1}} \prod_{i=1}^{n-k+1} \frac{\mu_i^{j_i}}{j_i!} \qquad (10)
\end{aligned}$$

where the sum is over all $j_1, j_2, \ldots, j_{n-k+1}$ satisfying $\sum_{i=1}^{n-k+1} j_i = k$ and $\sum_{i=1}^{n-k+1} i j_i = n$. The sum on the right-hand side of Equation (10) is known in the enumerative combinatorics literature as the ordinary Bell polynomial (Charalambides, 2002),

$$\begin{aligned}
&\hat{B}_{n,k}(\mu_1, \mu_2, \ldots, \mu_{n-k+1}) \\
&\quad = \sum_{j_1, j_2, \ldots, j_{n-k+1}} k! \prod_{i=1}^{n-k+1} \frac{\mu_i^{j_i}}{j_i!}, \qquad (11)
\end{aligned}$$

and can be related to the exponential Bell polynomial defined in Equation (5) according to

$$\begin{aligned}
&\hat{B}_{n,k}(\mu_1, \mu_2, \ldots, \mu_{n-k+1}) \\
&\quad = \frac{k!}{n!} B_{n,k}\left(1!\mu_1, 2!\mu_2, \ldots, (n-k+1)!\mu_{n-k+1}\right),
\end{aligned}$$

which completes the proof. □

Remark 2 (Asymptotic Runtime)**.** *As discussed above and explored experimentally in Section 3 below, the advantage of our novel sampling procedure in Algorithm 1 over the naïve sampling algorithm presented in Betancourt et al. (2022) lies in the fact that its runtime is not sensitive to the cluster size distribution μ. This is of particular importance when $\Pr[E_n \mid \mu]$ is small (e.g., because $\mathbb{E}[S_1 \mid \mu]$ is large compared to n). Thus, our concerns in this paper are largely with finite runtime comparison between these two methods. Nonetheless, it may be informative to compare the asymptotic runtimes required by these two algorithms to produce B random partitions, e.g., for use in prior calibration. We find that these two asymptotic runtimes are comparable in the setting where B is of the same order as n. See Section 2.4 for details.*

With Theorem 4 in hand, we can precisely describe the behavior of K_n for a particular choice of μ (or a prior over μ), in terms of Bell polynomials.

Example: ESC-Poisson, continued.

Under the ESC-Poisson distribution (see Section 2.1 above), cluster sizes are drawn according to a (shifted) Poisson. In Section 2.1, using Theorem 3, we computed $\Pr[E_n \mid \mu]$ for this distribution. Here, we use Theorem 4 to derive the distribution of the number of clusters K_n. By Theorem 4, for $k \in [n]$,

$$\begin{aligned}
\Pr[K_n = k \mid E_n, \mu] &= \frac{\hat{B}_{n,k}(\mu)}{\Pr[E_n \mid \mu]} \\
&= \frac{k! B_{n,k}\left(e^{-\lambda}, 2e^{-\lambda}\lambda, \ldots, (n-k+1)e^{-\lambda}\lambda^{n-k}\right)}{n! \Pr[E_n \mid \mu]} \\
&= \left(\sum_{\ell=1}^{n} \frac{e^{-\ell\lambda}(\ell\lambda)^{n-\ell}}{(n-\ell)!}\right)^{-1} \frac{e^{-k\lambda}(k\lambda)^{n-k}}{(n-k)!}
\end{aligned}$$

where we have used Equations (7), (8) and (9).

2.3 Illustrative Examples

Theorems 3 and 4 allow us to determine the behavior of ESC models for arbitrary choices of cluster size distribution, as we now demonstrate.

Negative Binomial Cluster Sizes.

The model that has received the most attention to date in the microclustering literature (see, e.g., Betancourt et al., 2022; Natarajan et al., to appear; Zanella et al., 2016) is the ESC-NB model, in which μ takes the form of a shifted negative binomial distribution,

$$\mu_k = \begin{cases} \binom{k+r-2}{k-1}(1-p)^r p^{k-1} & \text{if } k = 1, 2, \ldots \\ 0 & \text{if } k = 0, \end{cases}$$

where $p \in [0, 1]$ is the probability of success and $r > 0$ is the number of failures. To permit the

possibility that r is not an integer, we define

$$\binom{r}{m} = \frac{(r)_m}{m!},$$

where $(r)_m$ denotes the falling factorial, $(r)_m = r(r-1)(r-2)\cdots(r-m+1)$.

We first establish an expression for $\Pr[E_n \mid \mu]$. By the definition in Equation (11),

$$\begin{aligned}
&\hat{B}_{n,k}(\mu_1, \mu_2, \ldots, \mu_{n-k+1}) \\
&\quad = k! \sum_{j_1, j_2, \ldots, j_{n-k+1}} \prod_{i=1}^{n-k+1} \frac{\mu_i^{j_i}}{j_i!} \\
&\quad = \sum_{s_1, s_2, \ldots, s_k} \prod_{i=1}^{k} \mu_{s_k},
\end{aligned}$$

where the second sum is over all positive integers $s_1, s_2, \ldots, s_k$ summing to n. Plugging in our definition of μ, this becomes

$$\begin{aligned}
&\hat{B}_{n,k}(\mu_1, \mu_2, \ldots, \mu_{n-k+1}) \\
&\quad = \sum_{s_1, s_2, \ldots, s_k} \prod_{i=1}^{k} \binom{s_i + r - 2}{s_i - 1} (1-p)^r p^{s_i - 1} \\
&\quad = (1-p)^{rk} p^{n-k} \sum_{s_1, s_2, \ldots, s_k} \prod_{i=1}^{k} \binom{s_i + r - 2}{s_i - 1}.
\end{aligned}$$

After a change of variables, we have

$$\begin{aligned}
&\hat{B}_{n,k}(\mu_1, \mu_2, \ldots, \mu_{n-k+1}) \\
&\quad = (1-p)^{rk} p^{n-k} \sum_{s_1, s_2, \ldots, s_k} \prod_{i=1}^{k} \binom{s_i + r - 1}{s_i},
\end{aligned}$$

where now the sum is over all non-negative integers $s_1, s_2, \ldots, s_k$ summing to $n-k$. A basic identity for binomial coefficients (Graham, Knuth, & Patashnik, 1994, Equation 5.14) states that

$$\binom{t}{m} = (-1)^m \binom{m - t - 1}{m}, \tag{12}$$

which holds for all $t \in \mathbb{R}$ and non-negative integers m. Taking $t = s_i + r - 1$ and $m = s_i$,

$$\begin{aligned}
&\hat{B}_{n,k}(\mu_1, \mu_2, \ldots, \mu_{n-k+1}) \\
&\quad = \frac{(1-p)^{rk} p^n}{p^k} \sum_{s_1, s_2, \ldots, s_k} \prod_{i=1}^{k} (-1)^{s_i} \binom{-r}{s_i} \\
&\quad = \frac{(-1)^{n-k} (1-p)^{rk} p^n}{p^k} \sum_{s_1, s_2, \ldots, s_k} \prod_{i=1}^{k} \binom{-r}{s_i}.
\end{aligned}$$

Applying the generalized Vandermonde convolution identity (Graham et al., 1994), a second application of Equation (12) yields

$$\begin{aligned}
&\hat{B}_{n,k}(\mu_1, \mu_2, \ldots, \mu_{n-k+1}) \\
&\quad = (-1)^{n-k} (1-p)^{rk} p^{n-k} \binom{-kr}{n-k} \\
&\quad = (-1)^{2(n-k)} (1-p)^{rk} p^{n-k} \binom{n-k+kr-1}{n-k} \\
&\quad = p^n \left(\frac{(1-p)^r}{p}\right)^k \binom{n + k(r-1) - 1}{n-k}
\end{aligned}$$

Applying Theorem 3 and plugging in the definition given in Equation (11),

$$\begin{aligned}
&\Pr[E_n \mid \mu] \\
&\quad = p^n \sum_{k=1}^{n} \frac{(1-p)^{rk}}{p^k} \binom{n + k(r-1) - 1}{n-k}.
\end{aligned} \tag{13}$$

Applying Theorem 4 and the Bell polynomial expressions above,

$$\begin{aligned}
&\Pr[K_n = k \mid E_n, \mu] \\
&\quad = \frac{p^{n-k}(1-p)^{rk}}{\Pr[E_n \mid \mu]} \binom{n + k(r-1) - 1}{n-k},
\end{aligned} \tag{14}$$

where $\Pr[E_n \mid \mu]$ is as in Equation (13). We note that this recovers Proposition 2 in Natarajan et al. (to appear) as a special case.

Geometric Cluster Sizes

As another illustrative example, consider the setting where cluster sizes are distributed according to a geometric distribution,

$$\mu_k = \begin{cases} (1-p)^{k-1} p & \text{if } k = 1, 2, \ldots \\ 0 & \text{if } k = 0. \end{cases}$$

Of course, one could characterize this model using the fact that the geometric distribution is a special case of the negative binomial, but we include this example for the sake of its comparative simplicity.

Applying Theorem 3 followed by Equation (7),

$$\begin{aligned}&\Pr[E_n \mid \mu] \\ &= \sum_{k=1}^{n} \frac{k! p^k (1-p)^{n-k}}{n!} \\ &\qquad \cdot B_{n,k}\left(1!, 2!, \ldots, (n-k+1)!\right).\end{aligned}$$

By a basic identity (Comtet, 1974, page 135),

$$B_{n,k}\left(1!, 2!, \ldots, (n-k+1)!\right) = \binom{n-1}{k-1} \frac{n!}{k!}. \quad (15)$$

After a change of variables, we conclude that

$$\begin{aligned}\Pr[E_n \mid \mu] &= (1-p)^{n-1} p \sum_{\ell=0}^{n-1} \binom{n-1}{\ell} \left(\frac{p}{1-p}\right)^{\ell} \\ &= p.\end{aligned}$$

Turning to the cluster size distribution under this model, Theorem 4 states that

$$\begin{aligned}&\Pr[K_n = k \mid E_n, \mu] \\ &= \frac{k!}{n! p} B_{n,k}(1!\mu_1, 2!\mu_2, \ldots, (n-k+1)!\mu_{n-k+1}).\end{aligned}$$

Applying Equations (7) and (15) yields

$$\Pr[K_n = k \mid E_n, \mu] = \frac{(1-p)^n}{p} \left(\frac{p}{1-p}\right)^k \binom{n-1}{k-1}.$$

Zipf-distributed Cluster Sizes

A more interesting example is given by the case where cluster sizes are drawn according to a discrete power law. We consider, for $\alpha > 1$, a Zipfian cluster size distribution, given by

$$\mu_k = \frac{k^{-\alpha}}{\zeta(\alpha)} \text{ for } k = 0, 1, 2, \ldots,$$

where $\zeta(\cdot)$ denotes the Riemann zeta function. Our results above imply that

$$u_n = \sum_{k=1}^{n} \frac{k!}{n! \zeta(\alpha)^k} B_{n,k}\Big(1!, \frac{2!}{2^\alpha}, \ldots, \frac{(n-k+1)!}{(n-k+1)^\alpha}\Big).$$

It is not immediately clear how to simplify this probability using basic Bell polynomial identities. Nonetheless, from Theorem 4, we have that for $k \in [n]$,

$$\begin{aligned}&\Pr[K_n = k \mid E_n, \alpha] \\ &= \frac{k! \zeta^{-k}(\alpha) B_{n,k}\left(1!, \frac{2!}{2^\alpha}, \ldots, \frac{(n-k+1)!}{(n-k+1)^\alpha}\right)}{\sum_{\ell=1}^{n} \ell! \zeta^{-\ell}(\alpha) B_{n,\ell}\left(1!, \frac{2!}{2^\alpha}, \ldots, \frac{(n-\ell+1)!}{(n-\ell+1)^\alpha}\right)},\end{aligned}$$

and the Bell polynomials appearing on the right-hand side can be computed in quadratic time according to the recurrence relation (Charalambides, 2002, Equations 11.11, 11.12)

$$\begin{aligned}&B_{n,k}\left(\mu_1, \mu_2, \ldots, \mu_{n-k+1}\right) \\ &\quad = \sum_{j=1}^{n-k+1} \mu_j B_{n-j,k-1}\left(\mu_1, \mu_2, \ldots, \mu_{n-j-k}\right).\end{aligned}$$

Thus, even in the absence of a closed-form expression for $\Pr[K_n \mid E_n, \alpha]$, the distribution of K_n can be obtained numerically.

A host of interesting questions are raised by ESC models in which cluster sizes exhibit heavy-tailed behavior as in the Zipf distribution. In such settings, $\Pr[E_n \mid \mu]$ may converge very slowly or not at all, and the naïve sampling algorithm may be especially slow. Indeed, if the cluster size distribution μ has infinite mean, classical renewal theory results (Barbu & Limnios, 2009) do not apply, and we have no guarantee that $\Pr[E_n \mid \mu]$ converges to a finite limit as $n \to \infty$. Further, existing microclustering results (e.g., Betancourt et al., 2022, Theorem 3) do not apply to cluster size distributions that lack a finite mean, leaving open the question as to whether these models yield the microclustering property at all.

2.4 Asymptotic Runtime

Our concern in this work is the finite-sample behavior of our sampler and its sensitivity to the cluster size distribution in comparison to the naïve sampling algorithm. We compare these finite-sample runtimes in Section 3 below. Nonetheless, it is of interest to compare the asymptotic runtimes of Algorithm 1 and the naïve sampling algorithm. Toward this end, observe that the naïve algorithm presented in Betancourt et al. (2022) generates a sequence of random variables

$S_1, S_2, \ldots, S_{T_n}$ so that $\sum_{j=1}^{T_n} S_j \geq n$, where $T_n = \min\{k : \sum_{j=1}^{k} S_j \geq n\}$. The probability that these random variables sum to n exactly (and hence constitute a valid partition of n) is $\Pr[E_n \mid \mu]$. Thus, on average, the naïve sampling algorithm must generate $O(\mathbb{E}T_n / \Pr[E_n \mid \mu])$ random variables to produce a single random partition. Provided $\mathbb{E}[S_1 \mid \mu] < \infty$, Wald's equation (Levin, Peres, & Wilmer, 2009) states that

$$\mathbb{E}T_n = \frac{\mathbb{E}\sum_{k=1}^{T_n} S_k}{\mathbb{E}[S_1 \mid \mu]} \geq \frac{n}{\mathbb{E}[S_1 \mid \mu]}.$$

It follows by Lemma 1 that the naïve sampling algorithm requires time $\Omega(Bn)$, to generate B random partitions, since $\Pr[E_n \mid \mu] \to 1/\mathbb{E}[S_1 \mid \mu]$.

In contrast, consider our novel sampling procedure in Algorithm 1. Before it can be used to produce any samples, this procedure requires quadratic time to compute the terms $\Pr[X_k = s; m]$ for all $1 \leq s \leq m \leq n$ in terms of $\Pr[E_m \mid \mu]$ for $m = 1, 2, \ldots, n$. Having computed these probabilities, generating a single partition requires generating K_n random variables, where K_n is the (random) number of clusters in the generated partition. Thus, to generate B sample partitions, Algorithm 1 requires $O(n^2) + O(BK_n)$ runtime.

It is common in microclustering tasks that the total number of observations is often on the order of between a few hundreds and tens of thousands. In such settings, the number of Monte Carlo samples B is likely to be at least of the same order as the number of observations n, and the $O(n^2) + O(BK_n)$ runtime required by Algorithm 1 is essentially equivalent to the $O(Bn)$ runtime required by the naïve algorithm, since $K_n \to n/\mathbb{E}[S_1 \mid \mu]$ under microclustering models when $\mathbb{E}[S_1 \mid \mu] < \infty$ (Betancourt et al., 2022, Theorem 1). To the best of our knowledge, the asymptotic behavior of K_n when this expectation is infinite is not yet established. Should it turn out that $K_n = o(n)$ in such settings (we conjecture as much, but a proof is beyond the scope of the present work), the K_n-dependence of Algorithm 1 would be especially desirable compared to the naïve sampler.

3 Experiments

We now turn to a brief experimental investigation of our theoretical results. We note that all computations discussed above are performed in logarithmic space to avoid overflow or underflow. A Python implementation of our method is available in the digital supplement.

3.1 Behavior of K_n

We begin by verifying that the samples generated by Algorithm 1 match their intended ESC clustering distribution (i.e., verifying Theorem 2). Theorem 4 establishes the distribution of the number of clusters K_n under ESC models. In particular, Equation (14) gives the distribution of K_n under the ESC-NB model, in which the cluster sizes are distributed according to a Negative Binomial with success parameter $p \in [0, 1]$ and number of failures $r > 0$. Figure 1 shows this distribution for $p = 0.5$ and $r = 2.0$. The left-hand plot contains a histogram of 2000 draws of K_n, based on clusterings generated from the naïve ESC sampling method (Betancourt et al., 2022). The right-hand plot contains an analogous histogram based on clusterings generated from Algorithm 1, computing the $\Pr[E_n \mid \mu]$ terms using Bell polynomial identities. In both subplots, the solid black line indicates the distribution of K_n predicted by Theorem 4. We see that both the naïve and Bell polynomial-based algorithms yield clusterings in which the behavior of K_n matches that predicted by Theorem 4.

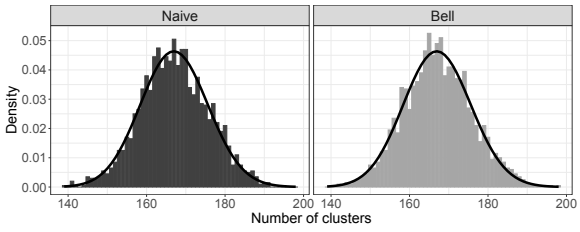

Fig. 1 Histogram of 2000 draws from the distribution of the number of clusters K_n under the ESC-NB model on $n = 500$ observations with Negative Binomial parameters $p = 0.5$ and $r = 2.0$ using the naïve (left) and Bell polynomial-based (right) sampling algorithms. The distribution predicted by Theorem 4 is indicated in black in both subplots.

3.2 Runtime Comparison

We now turn to a comparison of our proposed sampling algorithm with the naïve sampling approach

described in Section 2.1 and used in most previous microclustering work (see, e.g., Betancourt et al., 2022). For simplicity, we consider the ESC-Poisson model, in which cluster sizes are drawn according to a Poisson distribution with parameter λ.

Lemma 1 suggests that the runtime of the naïve sampling algorithm is likely to be sensitive to the mean of the cluster size distribution $\mathbb{E}S_1 = \lambda$. To examine this fact, we generated partitions of $n = 500$ objects under the ESC-Poisson model with Poisson parameter λ using both the naïve procedure and the procedure described in Algorithm 1. For varying values of the Poisson parameter λ, we performed 20 independent repetitions, recording the runtime required to generate clusterings under both methods. The mean runtime over these 20 replicates for these two methods are summarized in Figure 2, with the naïve method indicated by circles connected by solid lines and the Bell polynomial-based method indicated by triangles connected by dashed lines. We see that the runtime of the naïve sampling error depends sensitively on the mean λ of the cluster size distribution. Specifically, runtimes for the naïve method are orders of magnitude slower for values of λ that do not (exactly or approximately) divide $n = 500$. Under such circumstances, if $S_1, S_2, \ldots, S_k$ are such that $\sum_{j=1}^{k} S_j = n$, either all of the summands must be moderately far from the mean $\mathbb{E}S_1 = \lambda$ of the cluster size distribution, or, if most of the summands are close to $\mathbb{E}S_1$, one or more must deviate significantly from it. In either event, such sequences are of especially low probability, and thus many sequences $S_1, S_2, \ldots$ must be generated before the event E_n occurs, increasing the average runtime of the naïve procedure.

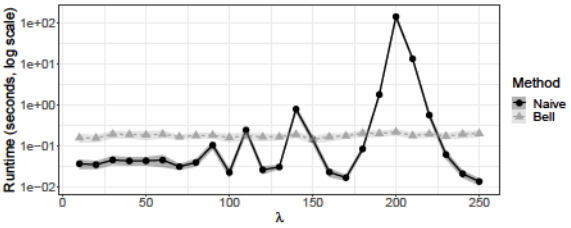

Fig. 2 Runtime in seconds required by the naïve (circles, solid line) and Bell polynomial-based (triangles, dashed line) methods to generate a partition of $n = 500$ objects under an ESC-Poisson model, as a function of the Poisson parameter λ. We see that the naïve sampling method is highly sensitive to the expected cluster size $\mathbb{E}S_1 = \lambda$.

Further examining Figure 2, we note that our proposed sampling method does not uniformly improve upon the naïve sampling method at all values of λ. This is owing to the fact that Algorithm 1 requires that we compute the probabilities

$$\Pr[X_k = s; m] = \frac{\mu_s \Pr[E_{m-s} \mid \mu]}{\Pr[E_m \mid \mu]} \tag{16}$$

for each $m \in [n]$ and each $s \in [m]$. Even with access to the sequences μ_m and $u_m = \Pr[E_m \mid \mu]$ for $m \in [n]$, constructing these probabilities ahead of time incurs a computational cost, which is included in the runtime reported in Figure 2.

Figure 3 compares the naïve sampling procedure and our proposed method, this time amortizing this up-front computational cost over 200 sample partitions. That is, each trial now consists of first calculating the probabilities in Equation (16), then using those probabilities to generate 200 clusterings from the ESC-Poisson model. We see that over a range of values of Poisson parameter λ and number of observations n, our proposed method improves upon the runtime of the naïve sampling method by an order of magnitude.

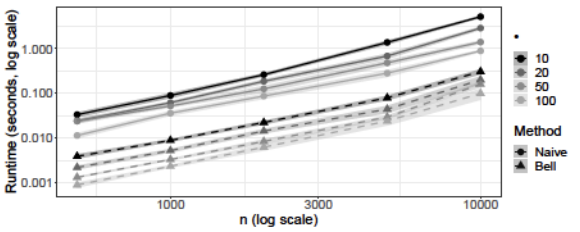

Fig. 3 Amortized runtime of the naïve ESC sampler (circles, solid line) and our proposed Bell polynomial-based method (triangles, dashed line), as a function of the number of observations n. Cluster sizes were generated according to a Poisson distribution with varying choices of mean λ (indicated by line shade). Each point corresponds to the mean runtime over 20 trials, with error bars indicating two standard errors of the mean. In each trial, 200 samples were generated from the ESC-Poisson model with parameter λ, and total runtime, including up-front computation required by the Bell polynomial-based method, was recorded.

4 Discussion and Conclusion

We have addressed two outstanding issues in ESC models: the behavior of the number of clusters K_n and the matter of sampling a priori from these

models. A number of natural follow-up questions present themselves. For example, all known results concerning the microclustering property in ESC models require that the cluster size distribution μ has finite expectation. It is natural to ask whether the microclustering property continues to hold if μ has infinite expectation, and how the size of the largest cluster grows in such situations. This question is the subject of ongoing work.

One possible criticism of Algorithm 1 is that it requires $O(n^2)$ up-front runtime to compute the probabilities $\Pr[X_k = s; m]$ for all $1 \leq s \leq m$. Absent particular structure in the cluster size distribution μ, it requires a new $O(n^2)$ runtime computation any time μ is updated. We stress that Algorithm 1 is not aimed at this situation, but rather is meant for faster a priori sampling, such as in the context of prior calibration. Nonetheless, future work should investigate speeding up these probability computations for use in Algorithm 1, perhaps using approximation techniques similar to those deployed in Bystrova et al. (2020). Such a speedup (even at the cost of approximation error) has the potential to yield the first feasible alternative to the chaperones algorithm for sampling a posteriori from ESC models.

Declarations

Competing interests: The authors have no completing interests to declare.

References

Aldous, D.J. (1985). Exchangeability and related topics. *École d'été de probabilités de saint-flour XIII-1983* (pp. 1–198). Springer.

Antoniak, C.E. (1974). Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *The Annals of Statistics*, *2*(6), 1152–1174,

Barbu, V.S., & Limnios, N. (2009). *Semi-Markov chains and hidden semi-Markov models toward applications: their use in reliability and DNA analysis* (Vol. 191). Springer.

Betancourt, B., Zanella, G., Steorts, R.C. (2022). Random partition models for microclustering tasks. *Journal of the American Statistical Association*, *117*(539), 1215–1227,

Binette, O., & Steorts, R.C. (2022). (almost) all of entity resolution. *Science Advances*, *8*(12), eabi8021,

Blei, D.M. (2012). Probabilistic topic models. *Communications of the ACM*, *55*(4), 77–84,

Blei, D.M., Ng, A.Y., Jordan, M.I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, *3*(4–5), 993–1022,

Bloem-Reddy, B., Foster, A., Mathieu, E., Teh, Y.W. (2018). Sampling and Inference for Beta Neutral-to-the-Left Models of Sparse Networks. *Proceedings of the thirty-fourth conference on uncertainty in artificial intelligence* (pp. 477–486).

Bystrova, D., Arbel, J., King, G.K.K., Deslandes, F. (2020). Approximating the clusters' prior distribution in bayesian nonparametric models. *3rd symposium of advances in approximate bayesian inference* (pp. 1–16).

Charalambides, C.A. (2002). *Enumerative combinatorics*. Chapman & Hall/CRC.

Comtet, L. (1974). *Advanced combinatorics*. D. Reidel Publishing Company.

Di Benedetto, G., Caron, F., Teh, Y.W. (2021). Non-exchangeable random partition models for microclustering. *The Annals of Statistics*, *49*(4), 1931–1957,

Falush, D., Stephens, M., Pritchard, J.K. (2003). Inference of population structure using multilocus genotype data: linked loci and correlated allele frequencies. *Genetics*, *164*(4), 1567–1587,

Gnedin, A., & Pitman, J. (2006). Exchangeable Gibbs partitions and Stirling triangles. *Journal of Mathematical Sciences*, *138*(3), 5674–5685,

Graham, R., Knuth, D., Patashnik, O. (1994). *Concrete mathematics: A foundation for computer science* (2nd ed.). Addison-Wesley.

Ishwaran, H., & James, L.F. (2003). Generalized weighted Chinese restaurant processes for species sampling mixture models. *Statistica Sinica*, *13*(4), 1211–1236,

Jitta, A., & Klami, A. (2018). On controlling the size of clusters in probabilistic clustering. *Proceedings of the thirty-second aaai conference on artificial intelligence* (pp. 3350–3357).

Klami, A., & Jitta, A. (2016). Probabilistic size-constrained microclustering. *Proceedings of the thirty-second conference on uncertainty in artificial intelligence* (pp. 329–338).

Kolchin, V.F. (1971). A problem of the allocation of particles in cells and cycles of random permutations. *Theory of Probability & Its Applications*, *16*(1), 74–90,

Lee, C.J., & Sang, H. (2022). Why the rich get richer? On the balancedness of random partition models. K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, & S. Sabato (Eds.), *Proceedings of the 39th international conference on machine learning* (Vol. 162, pp. 12521–12541).

Legramanti, S., Rigon, T., Durante, D., Dunson, D.B. (2022). Extended stochastic block models with application to criminal networks. *Annals of Applied Statistics*, *16*(4), 2369–2395,

Levin, D.A., Peres, Y., Wilmer, E.L. (2009). *Markov chains and mixing times* (1st ed.). American Mathematical Society.

Natarajan, A., De Iorio, M., Heinecke, A., Mayer, E., Glenn, S. (to appear). Cohesion and repulsion in Bayesian distance clustering. *Journal of the American Statistical Association*, ,

Pitman, J. (2006). *Combinatorial stochastic processes*. Springer.

Pritchard, J.K., Stephens, M., Donnelly, P. (2000). Inference of population structure using multilocus genotype data. *Genetics*, *155*(2), 945–959,

Sethuraman, J. (1994). A constructive definition of Dirichlet priors. *Statistica Sinica*, *4*, 639–650,

Silverman, J.D., & Silverman, R.K. (2017). The Bayesian sorting hat: A decision-theoretic approach to size-constrained clustering. *arXiv:1710.06047*, ,

Zanella, G., Betancourt, B., Wallach, H., Miller, J., Zaidi, A., Steorts, R.C. (2016). Flexible models for microclustering with application to entity resolution. D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, & R. Garnett (Eds.), *Proceedings of neural information processing systems 29.*