Computational Statistics
https://doi.org/10.1007/s00180-021-01113-3

ORIGINAL PAPER

A simulated annealing-based algorithm for selecting balanced samples

Roberto Benedetti[1] · **Maria Michela Dickson**[2] · **Giuseppe Espa**[2] · **Francesco Pantalone**[3] · **Federica Piersimoni**[4]

Received: 3 July 2020 / Accepted: 5 May 2021

Abstract
Balanced sampling is a random method for sample selection, the use of which is preferable when auxiliary information is available for all units of a population. However, implementing balanced sampling can be a challenging task, and this is due in part to the computational efforts required and the necessity to respect balancing constraints and inclusion probabilities. In the present paper, a new algorithm for selecting balanced samples is proposed. This method is inspired by simulated annealing algorithms, as a balanced sample selection can be interpreted as an optimization problem. A set of simulation experiments and an example using real data shows the efficiency and the accuracy of the proposed algorithm.

Keywords Balanced sampling · Auxiliary variables · Sampling algorithms · Simulated annealing

1 Introduction

Balanced sampling refers to a class of techniques aimed at randomly selecting units from a given population. The selection of balanced samples was first introduced by Gini (1928) and later recalled by Yates (1946) and Thionnet (1953), stimulating from then on an increasing interest in addressing, by means of balancing, several practical survey sampling problems (see for example Falorsi and Righi 2008; Grafström and Tillé 2013; Brus 2015; Benedetti and Piersimoni 2017; Chauvet 2017; Marazzi and

✉ Maria Michela Dickson
mariamichela.dickson@unitn.it

1 Department of Economic Studies, University "G. d'Annunzio" of Chieti-Pescara, Pescara 65127, Italy

2 Department of Economics and Management, University of Trento, Trento 38122, Italy

3 Department of Economics, University of Perugia, Perugia 06123, Italy

4 Directorate for Methodology and Statistical Process Design, Istat, Rome 00184, Italy

Published online: 26 May 2021

Tillé 2017; Tillé et al. 2018; Chauvet and Le Gleut 2019; Kermorvant et al. 2019). The peculiarity of balanced sampling consists of exploiting a priori auxiliary information available for all units of a population at the design stage, so that the expansion estimator (Narain 1951; Horvitz and Thompson 1952) returns the balancing variables totals known at the population level. The stronger the correlation among the variables of interest and the balancing variables, the higher the effiency of a balanced sampling design in estimating the population total.

Several contributions have been proposed in literature to select balanced samples (for a review see Valliant et al. 2000). All the methods are configured as partial solutions for a variety of reasons, the main of which are a considerable computing time when several balancing variables are used (e.g. in enumerative sampling; Ardilly 1991), and problems in respecting the original inclusion probabilities (e.g. in rejective sampling; Hájek 1981). A general solution for balanced sampling was finally proposed by Deville and Tillé (2004), whose *cube method* allows for the selection of balanced samples with equal or unequal inclusion probabilities and any number of auxiliary variables with fast execution time (Chauvet and Tillé 2006; Grafström and Lisic 2016). This method is based on a random transformation of the inclusion probabilities vector to draw a sample that exactly, or at least approximately, satisfies the original inclusion probabilities and balancing equations (Tillé 2011). Deville and Tillé (2004) provided a solution that allows for the selection of approximate balanced samples, while respecting the inclusion probabilities (for an exhaustive presentation of the cube method, see Deville and Tillé 2004; Tillé 2006).

The selection of balanced samples may be viewed as a constrained optimization problem with discrete variables and a solution may be borrowed from physics. In particular, a Markov chain Monte Carlo method aimed at solving complex combinatorial problems, such as e.g. the traveling salesman problem, is a well-known *simulated annealing* method (Kirkpatrick et al. 1983). This method uses the Metropolis-Hastings algorithm for approximating the global optimum of a given function and is a preferable alternative when the approximation of a global optimum is more important than finding a precise local one. For a review of the applications of simulated annealing, see e.g. Aarts and van Laarhoven (1987). In the present paper, we show that a deterministic version of simulated annealing may be applied in sampling context to select high quality, fixed-size balanced samples. A very fast algorithm called *Simulated Annealing for Balanced Sampling* (*sabs*) is presented and its quality in terms of sample balance and respect of inclusion probabilities is evaluated by means of Monte Carlo simulations. A comparison with the cube method is also carried out. The paper is structured as follows. In Sect. 2, preliminaries and notations are given. In Sect. 3, the new algorithm is presented and both technical and theoretical aspects are detailed. In Sect. 4, results of a simulation study exploring several scenarios are presented. Section 5 is devoted to the practical application of the algorithm to a real data-set. Finally, Sect. 6 discusses the results and concludes.

2 Preliminaries and notation

Let U be a finite population of size N composed by k units, with $k \in \{1, ..., N\}$. Let denote with y the variable of interest and with y_k, $k \in U$, the value of y in the $k-th$ population unit. The aim is to estimate the population total $Y = \sum_{k \in U} y_k$. Let S be a random fixed-size sample defined as a subset of U selected under a probability distribution $p(\cdot)$ and according to a without replacement sampling design, such that $Pr(S = s) = p(s)$ where $\sum_{s \subset U} p(s) = 1$, for each $s \in U$. A random sample may also be defined as a discrete random non-negative vector $\mathbf{a} = (a_1, ..., a_k, ..., a_N)^{\mathrm{T}}$, where a_k indicates the number of selections of the unit $k \in U$ in the sample. The variable a_k is an inclusion indicator, which is, in without replacement sampling, equal to 1 if the unit k is selected in the sample, and 0 otherwise (i.e. following a Bernoulli distribution). The inclusion probability π_k is the probability for the unit k to be selected in the sample. This probability is derived from the chosen sampling design as $\pi_k = Pr(k \in S) = E_p(a_k) = \sum_{s \subset U_{s \ni k}} p(s)$, for each $k \in U$. In a design-based perspective, it is possible to estimate the population total Y by using the expansion estimator (Narain 1951; Horvitz and Thompson 1952) of the total $\widehat{Y} = \sum_{k \in S} \frac{y_k}{\pi_k} = \sum_{k \in U} \frac{y_k a_k}{\pi_k}$, with $\pi_k > 0$.

Let $\mathbf{z}_k = (z_{k1}, ...z_{kj}, ..., z_{kJ})^{\mathrm{T}}$ be a vector of J auxiliary variables known for all the units in the population U, such that the knowledge of $\mathbf{z}_k$ allows for the computation of J totals as $Z_j = \sum_{k \in U} z_{kj}$, $j = 1, ..., J$. When a sample is selected, the expansion estimator of J auxiliary variables can be computed as $\hat{Z}_j = \sum_{k \in S} \frac{z_{kj}}{\pi_k}$. A sampling design $p(s)$ is said to be *balanced* on the vector of the auxiliary variables $z_k^{\mathrm{T}} = (z_1, ..., z_J)$ if and only if it satisfies the balancing equations

$$\sum_{k \in S} \frac{z_k}{\pi_k} = \sum_{k \in U} z_k.$$

The selection of a balanced sample may be a challenging task because it is a problem that cannot satisfy non-integer constraints. Indeed, there may exist a rounding problem that prevents the exact satisfaction of the balancing constraints. Therefore, the aim of this study is to find a design that exactly, or at least approximately, satisfies the balancing equations. The rounding problem becomes less relevant when the sample size is high, but this condition may not be compatible with the practical constraints of the conduction of a survey. In addition, the respect of the inclusion probabilities cannot be overlooked, as they are not always satisfied with some balanced sampling methods.

Moreover, balanced sampling may be formulated and solved as a linear programming problem. In this respect, to each sample is assigned a cost function $C(s)$, which is equal to zero if the sample is perfectly balanced and has greater values as the sample becomes increasingly unbalanced. The aim here is to find a sampling design $p(s)$ that minimizes the mean cost $\sum_{s \subset U} C(s)p(s)$, subject to the constraint to respect the inclusion probabilities $\sum_{s \subset U} p(s) = 1$ and $\sum_{s \subset U, s \ni k} p(s) = \pi_k$, with $k \in U$.

Deville and Tillé (1998) demonstrated that the solution to the problem leads to the selection of a minimal support design. Unfortunately, applying linear programming

is, in most cases, unfeasible, as it is necessary to enumerate all the possible samples. Therefore, the number of samples 2^N is too big to be a suitable solution. This reveals the need for a balanced sampling design to avoid such enumeration. In this respect, the cube method proposed by Deville and Tillé (2004) allows one to consider the 2^N possible samples as 2^N vectors in $\mathbb{R}^N$. This method provides a general solution to balance on several variables, with equal and unequal inclusion probabilities. The cube method is based on a random transformation of the inclusion probabilities vector and a sample is obtained when the inclusion probabilities are exactly satisfied and the balancing equations are satisfied as well as possible (Deville and Tillé 2004; Chauvet and Tillé 2006).

3 Simulated annealing-based algorithm

In balanced sampling, the aim, according to a multivariate distribution with conditions on the support, is to select a random sample of fixed-size. Simulated annealing may represent a suitable solution since it can be used to generate samples from any high-dimensional distribution if the probability function is known. Exploiting the idea at the basis of the method, a combinatorial optimization problem can be viewed as a stochastic process (Robert and Casella 2013), in which local conditional distributions are dependent from a global control parameter, i.e. the so-called temperature (for a readable explanation, see Geman and Geman 1984). Simulated annealing requires the generation of a finite sequence of decreasing values of the temperature (the annealing schedule) according to a probabilistic decreasing function (i.e. the Metropolis-Hastings algorithm), in order to converge toward a set of global optimal solutions (i.e. obtaining the minimum energy states). The parallelism with sampling problems easily follows. Specifically, let's define an energy function

$$f(\mathbf{a}) = \sqrt{\frac{\sum_{j=1}^{J}\left(\hat{Z}_j - Z_j\right)^2}{J}}.$$

This represents a mean quadratic distance function among the estimated totals and the known totals of the balancing variables, for any possible configuration of the sample $\mathbf{a} \in \{0, 1\}^N$. Note that any distance function can be used as an energy function and no restrictions exist on it. Clearly, if $f(\mathbf{a}) = 0$, the balance of the sample is exactly satisfied. The proposed sampling algorithm works to achieving this condition, by searching an optimal configuration. Hence, for configuration $\mathbf{a}^{(i)}$ obtained at the $i - th$ attempt of the algorithm, it may be possible to obtain another configuration $\mathbf{a}^{(e)}$, in which the inclusion indicator label is randomly exchanged among two units at each iteration, ensuring the respect of the sample size. The second configuration is preferred to the first if $f\left(\mathbf{a}^{(e)}\right) < f\left(\mathbf{a}^{(i)}\right)$, using a deterministic approach, following the idea of Besag (1986) in the image processing context. The maximum number of iterations is equal to $MAXITER$, where each iteration consists of N attempts, and where $MAXITER$ is chosen by the user and N is the population size. Therefore, the algorithm performs a maximum of $MAXITER \cdot N$ attempts. The procedure stops if a pre-fixed respect of minimal balancing constraints is reached. Thus, a final

configuration is always obtainable and no restrictions on summary index used in the convergence ($CONV$) check are needed.

Algorithm 1 sabs: simulated annealing-based algorithm for selecting balanced sampling.

Start with the initial configuration $\mathbf{a}^{(1)}$, given by a simple random sample of size n;
$d = \infty$;
$p = 1$;
$i = 1$;
while $d > CONV$ & $p <= MAXITER$ **do**
  c = 1;
  while $d > CONV$ & $c <= N$ **do**
    randomly select one unit k in the current configuration $\mathbf{a}^{(i)}$, i.e. $a_k^{(i)} = 1$;
    randomly select one unit l not selected in the current configuration $\mathbf{a}^{(i)}$, i.e. $a_l^{(i)} = 0$;
    define the new configuration $\mathbf{a}^{(e)}$, given by $\mathbf{a}^{(i)}$ but with the two previously selected units k and l interchanged, i.e. $a_k^{(e)} = 0$ and $a_l^{(e)} = 1$;
    if $f(\boldsymbol{a}^{(e)}) < f(\boldsymbol{a}^{(i)})$ **then**
      $\mathbf{a}^{(i+1)} = \mathbf{a}^{(e)}$;
    else
      $\mathbf{a}^{(i+1)} = \mathbf{a}^{(i)}$;
    end
    $d = \max\left(\frac{|\hat{\mathbf{Z}} - \mathbf{Z}|}{\mathbf{Z}}\right)$;
    c++;
    i++;
  end
  p++;
end
The last selected configuration represents the selected sample.

This procedure implies that a local minimum, not necessarily a global one, is reached. The way to avoid the entrapment in local minima lays in the use of a probabilistic decreasing rule, such as in traditional implementation of simulated annealing. Unfortunately, the main contraindication is to require strong computational efforts, because the temperature needs to be decreased very slowly so that all possible solutions may be visited. This makes this procedure difficult to be used in practice, especially with a high number of balancing variables and with large populations. In addition, the search for global optima may produce undesirable solutions in terms of the selected samples. However, according to our computational experience, this algorithm is based on a deterministic decreasing rule that ensures the balancing constraints can be satisfactorily achieved in a reasonable number of iterations and obtains first-order inclusion probabilities very close to those desired, as showed in the following section.

From a practical point of view, the sabs selects a set of fixed-size random samples without replacement, according to the initial vector of inclusion probabilities. This means that for each configuration, it is possible to resort to the expansion estimator

Table 1 Generated populations according to Uniform, Normal, Exponential, and Bimodal distributions, summarized for the population sizes

Distribution	N		
	1000	5000	10000
$Uniform$	U_1	U_5	U_9
$Normal$	U_2	U_6	U_{10}
$Exponential$	U_3	U_7	U_{11}
$Bimodal$	U_4	U_8	U_{12}

and to exploit its properties in total estimation, variance, and variance estimation (Horvitz and Thompson 1952).

4 Simulation experiments

In order to check the properties of the proposed sampling algorithm, several simulation experiments were carried out. Twelve populations have been generated according to different distributions and sizes. Indeed, an Uniform $\mathcal{U} \sim (0, 1)$, a Normal $\mathcal{N} \sim (0, 1) + 6$, an Exponential $Exp \sim (0.5)$, and a Bimodal defined as a mixture of normals $Bim \sim 0.4\mathcal{N}(2, 0.25) + 0.6\mathcal{N}(4, 0.25)$, each with population sizes equal to 1000, 5000 and 10000 units, have been considered. These populations are referred to as U_1 - U_{12}, and a summary of the characteristics is reported in Table 1.

For each population, $J = 3; 5; 10$ independent auxiliary variables have been generated, for a total of 36 scenarios. We considered a relevant number of auxiliary variables in order to evaluate the performance of the proposed method in the presence of a vast information asset.

For each of the above-mentioned scenarios, $M = 10000$ fixed-size samples have been selected with equal inclusion probabilities and with sampling fractions $\mathrm{f} = 0.01; 0.05; 0.1$, by means of the sabs and cube methods. Indeed, the proposed algorithm has been compared with the cube method since it is the most used sampling design for the selection of balanced samples; therefore, it is the most appropriate competitor. Moreover, in order to investigate some extreme cases as well, four small populations (U_{13} - U_{16}) of dimension $N = 100$ have been generated according to the aforementioned distributions, with $J = 3; 5; 10$ independent auxiliary variables. Here also $M = 10000$ fixed-size samples have been selected, this time with sampling fractions $\mathrm{f} = 0.1; 0.25$. Note that the sabs algorithm has been set to stop when the minimal balancing constraints equal to 0.1% is reached, with a maximum number of iterations equal to $10 \times N$.

The simulation focuses on two very important aspects in the evaluation of a balanced sampling design: respect of first-order inclusion probabilities and respect of balancing constraints. The former is investigated by means of the relative Root Mean Squared Error, defined as:

$$rRMSE_{\pi_k} = \frac{\sqrt{\frac{\sum_{k=1}^{N}\left(\frac{v_k}{M} - \pi_k\right)^2}{N}}}{\mathrm{f}},$$

where v_k is the number of times the k-th unit is selected in M Monte Carlo replicates. The difference in the respect of the balancing constraints has been defined as:

Table 2 Results of the Monte Carlo experiment on populations U_1, U_2, U_3, and U_4, generated according to Uniform, Normal, Exponential, and Bimodal distributions with $J = 10$ for three sampling fractions f = 0.01; 0.05; 0.1. Monte Carlo replicates $M = 10000$

$rRMSE_{\pi_k}$								
	$U_1\ (Uniform)$		$U_2(Normal)$		$U_3(Exponential)$		$U_4(Bimodal)$	
f	*sabs*	*cube*	*sabs*	*cube*	*sabs*	*cube*	*sabs*	*cube*
0.01	0.1658	0.0980	0.2034	0.0975	0.2407	0.0974	0.2375	0.1011
0.05	0.0480	0.0423	0.0527	0.0443	0.0573	0.0449	0.0514	0.0434
0.1	0.0313	0.0295	0.0331	0.0295	0.0341	0.0296	0.0314	0.0304
CD								
0.01	0.0520	0.1486	0.0141	0.0437	0.0700	0.3480	0.0144	0.0560
0.05	0.0103	0.0288	0.0028	0.0088	0.0124	0.0652	0.0028	0.0112
0.1	0.0051	0.0143	0.0014	0.0044	0.0062	0.0331	0.0014	0.0056

Table 3 Results of the Monte Carlo experiment on populations U_9, U_{10}, U_{11}, and U_{12}, generated according to Uniform, Normal, Exponential, and Bimodal distributions with $J = 10$ for three sampling fractions f = 0.01; 0.05; 0.1. Monte Carlo replicates $M = 10000$

$rRMSE_{\pi_k}$								
	$U_9\ (Uniform)$		$U_{10}(Normal)$		$U_{11}(Exponential)$		$U_{12}(Bimodal)$	
f	*sabs*	*cube*	*sabs*	*cube*	*sabs*	*cube*	*sabs*	*cube*
0.01	0.0998	0.0995	0.1005	0.0987	0.1007	0.1002	0.1012	0.0988
0.05	0.0439	0.0435	0.0436	0.0437	0.0443	0.0434	0.0436	0.0437
0.1	0.0301	0.0304	0.0299	0.0299	0.0299	0.0210	0.0297	0.0296
CD								
0.01	0.0038	0.0142	0.0010	0.0044	0.0044	0.0338	0.0010	0.0056
0.05	0.0009	0.0028	0.0009	0.0009	0.0009	0.0067	0.0009	0.0011
0.1	0.0009	0.0014	0.0009	0.0004	0.0009	0.0034	0.0009	0.0006

$$CD = mean(\mathbf{d}),$$

with $\mathbf{d} = \{d_1, ..., d_m, ..., d_M\}$ and $d_m = max_j \left(\frac{\hat{Z}_{j,m} - Z_j}{Z_j} \right)$.

Moreover, a comparison in terms of computational time has been performed. In particular, as an example, we report the elapsed time to select one sample on the populations generated by $\mathcal{U} \sim (0, 1)$.

4.1 Respect of the inclusion probabilities and of the balancing constraints

The most relevant results of the Monte Carlo experiments on U_1 - U_4 and U_9 - U_{12} are reported in Tables 2 and 3 and are graphically summarized in Figs. 1, 2, 3 and 4, while remaining simulation results are reported in the Supplementary Material.

The simulation results motivate the following comments.

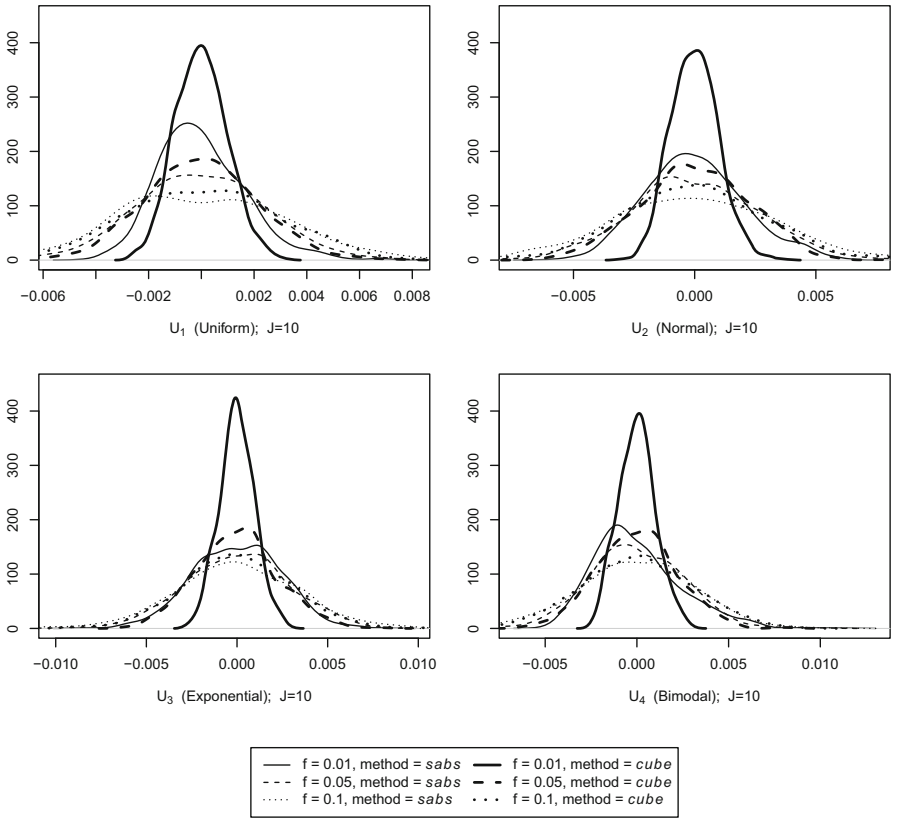

Fig. 1 Results of the Monte Carlo experiment regarding the respect of the inclusion probabilities. Simulations were performed on populations U_1, U_2, U_3, and U_4, generated according to Uniform, Normal, Exponential, and Bimodal distributions with $J = 10$ for three sampling fractions f $= 0.01; 0.05; 0.1$. Monte Carlo replicates $M = 10000$. X-axis: rRMSE values; Y-axis: frequency (density) of selected samples

With respect to the inclusion probabilities, the cube method showed better performance than the sabs method. Furthermore, the performance difference between the two methods decreased when the sampling fraction increased. Note that the largest sampling fraction was considered equal to 10% of the population size. This behaviour was detected on all of the considered populations.

With respect to the balancing constraints, the performances of the sabs was very encouraging. The method showed errors very near to zero, for all three population sizes and for the smaller sampling fractions considered. The differences in performance among the sabs and cube methods increased with an increased population size (see results reported in the Supplementary Material).

The considerations made up until this point were valid for all of the considered distributions used to generate the populations presented so far.

With regard to the populations of size equal to 100 units, some simulation results are reported in Table 4, while the extended set-up is reported in the Supplementary

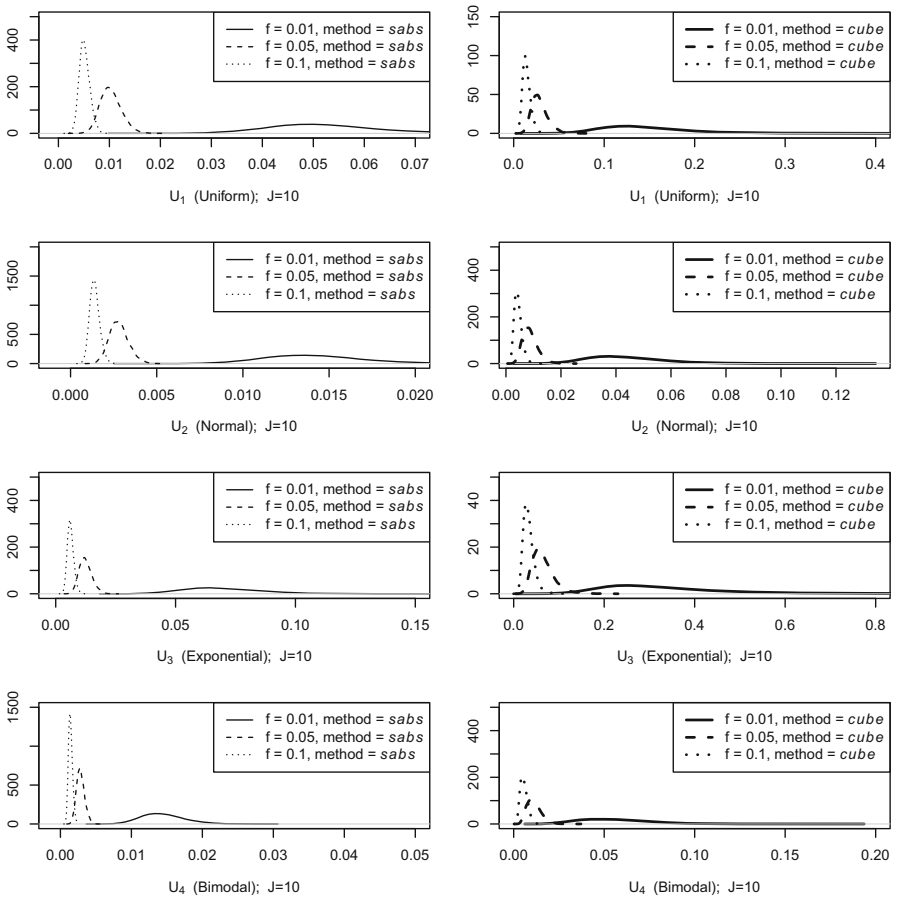

Fig. 2 Results of the Monte Carlo experiment regarding the respect of the balancing constraints. Simulations were performed on populations U_1, U_2, U_3, and U_4, generated according to Uniform, Normal, Exponential, and Bimodal distributions with $J = 10$ for three sampling fractions f $= 0.01; 0.05; 0.1$. Monte Carlo replicates $M = 10000$. X-axis: rRMSE values; Y-axis: frequency (density) of selected samples

Material. Sampling fractions have been increased compared to previous examples in order to provide the balancing on the same number of constraints.

The latter populations represent a very particular case due to the very small population sizes, which in turn implies very small sample sizes. Neverthless, the behaviour of the sabs method is confirmed. The performances in respect of the inclusion probabilities remain the best for the cube method, even as they become very similar with the growth of the sampling fractions. It should be noted that with the bimodal distribution, the $rRMSE_{\pi_k}$ is practically identical for both methods. In respect of the balancing constraints, the performances of the sabs were again confirmed as the best for every distribution and every sampling fraction considered.

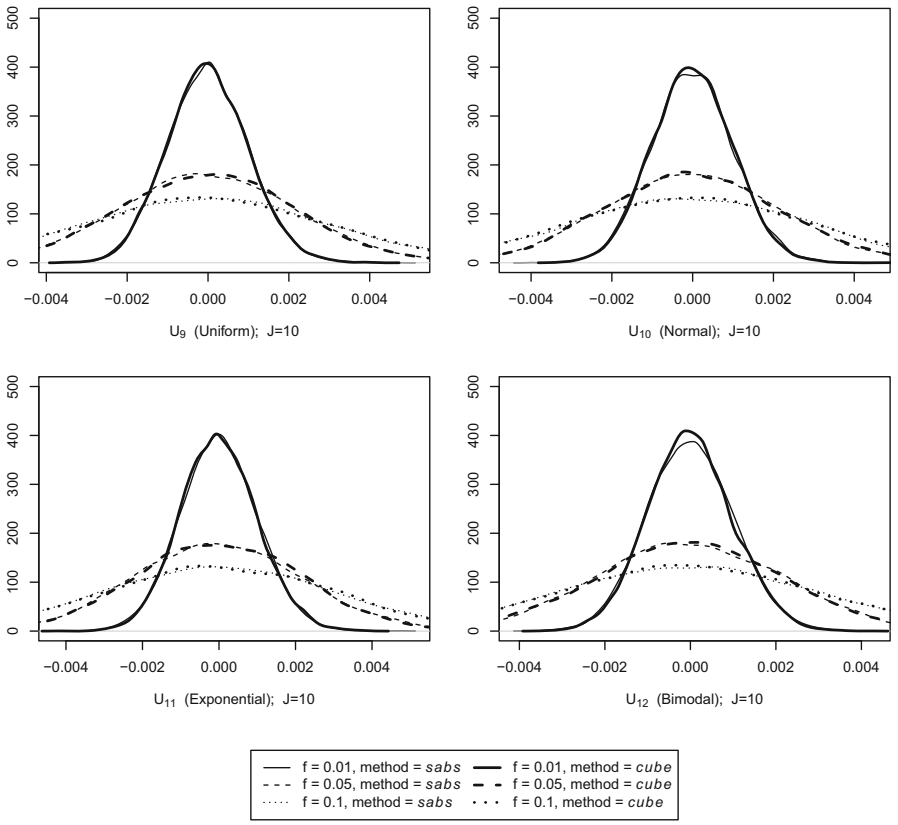

Fig. 3 Results of the Monte Carlo experiment regarding the respect of the inclusion probabilities. Simulations were performed on the populations U_9, U_{10}, U_{11}, and U_{12}, generated according to Uniform, Normal, Exponential, and Bimodal distributions with $J = 10$ for three sampling fractions f $= 0.01; 0.05; 0.1$. Monte Carlo replicates $M = 10000$. X-axis: rRMSE values; Y-axis: frequency (density) of selected samples

Table 4 Results of the Monte Carlo experiment on the populations U_{13}, U_{14}, U_{15}, and U_{16}, generated according to Uniform, Normal, Exponential, and Bimodal distributions with $J = 10$ for two sampling fractions f $= 0.1; 0.25$. Monte Carlo replicates $M = 10000$.

$rRMSE_{\pi_k}$								
	U_{13} $(Uniform)$		$U_{14}(Normal)$		$U_{15}(Exponential)$		$U_{16}(Bimodal)$	
f	*sabs*	*cube*	*sabs*	*cube*	*sabs*	*cube*	*sabs*	*cube*
0.1	0.1032	0.0326	0.1522	0.0281	0.2174	0.0306	0.0293	0.0285
0.25	0.0272	0.0192	0.0403	0.0173	0.0584	0.0176	0.0712	0.0712
CD								
0.1	0.0731	0.1559	0.0195	0.0447	0.1060	0.3707	0.0221	0.0625
0.25	0.0289	0.0604	0.0077	0.0180	0.0385	0.1369	0.0085	0.0244

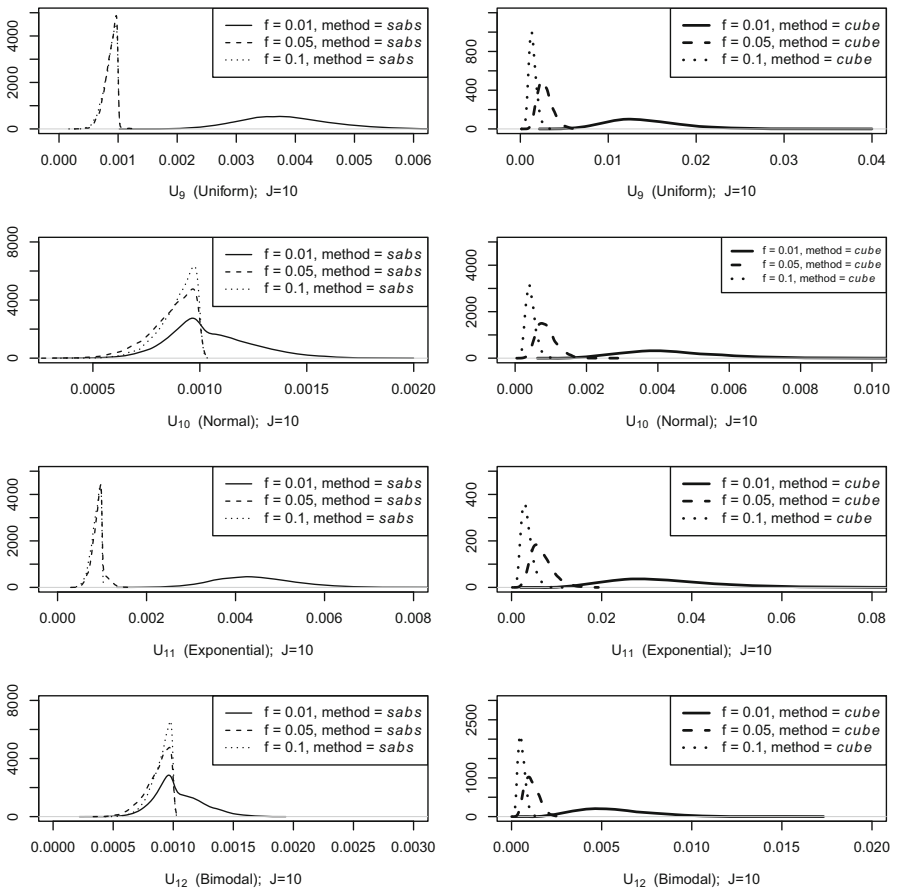

Fig. 4 Results of the Monte Carlo experiment regarding the respect of the balancing constraints. Simulations were performed on the populations U_9, U_{10}, U_{11}, and U_{12}, generated according to Uniform, Normal, Exponential, and Bimodal distributions with $J = 10$ for three sampling fractions $f = 0.01; 0.05; 0.1$. Monte Carlo replicates $M = 10000$. X-axis: rRMSE values; Y-axis: frequency (density) of selected samples

4.2 Time of computation

The computing time needed for the selection of samples for a given sampling method could be an important factor in both surveys and simulations, especially if it is highly dependent on N and n. Table 5 reports the average CPU time in seconds taken by each of the algorithms to select a sample for different populations and sample sizes. Simulations were performed on a 2.3 GHz Dual-Core Intel Core i5. Samples were selected using R software. In particular, the sabs algorithm was implemented in C++ via `Rcpp`; the cube method was implemented by means of the `BalancedSampling` package (Grafström and Lisic 2016), which is a fast implementation in C++ via `Rcpp`, and by means of the `sampling` package (Tillé and Matei 2009).

Table 5 Elapsed time in seconds for the selection of one sample (average over 10 Monte Carlo replicates) by means of the sabs algorithm and of two versions of the cube method. Populations used were U_1, U_5, and U_9. Balancing variables: $J = 10$. Sampling fractions f $= 0.01; 0.05; 0.1$

N	f	*sabs* (C++)	*cube* (C++)	*cube* (R)
1000	0.01	0.0007	0.0029	0.1028
	0.05	0.0008	0.0029	0.1015
	0.1	0.0009	0.0028	0.0994
5000	0.01	0.0047	0.0136	0.5236
	0.05	0.0049	0.0135	0.5226
	0.1	0.0024	0.0133	0.5387
10,000	0.01	0.0099	0.0259	1.1659
	0.05	0.0031	0.0275	1.1312
	0.1	0.0016	0.0260	1.0600

The sabs algorithm was the least computationally intensive method used, as it was sensibly quicker than the cube method implemented in both ways. It increased gradually with n and only proportionally to N, mainly because the number of attempts was set equal to N. Thus, we can be confident that the sabs algorithm can be effectively applied without extensive difficulties regarding large population sizes and without any storage or RAM limitations other than those represented by the size of the frame from which we wanted to select the sample.

5 An experiment on real data

Real data was used in order to investigate the efficiency of the sampling design when estimating a target variable in a real context. The dataset we used is freely available from www.statbel.fgov.be and concerns fiscal statistics on income subject to personal income tax. Data are related to the 581 Belgian municipalities for the period between 2005–2018. By focusing attention on information related to the last available year, we aim to estimate the total Belgian net income (Y) by exploiting the three following auxiliary variables: the number of declarations with a total net taxable income greater than zero (z_1), the number of declarations where the amount of total taxes is greater than zero (z_2), and the number of residents per municipality (z_3). We selected $M = 10000$ fixed-size samples with equal inclusion probabilities, for sampling fractions f $= 0.01; 0.05; 0.1$, by means of the sabs algorithm, the cube method and simple random sampling without replacement (srswor). The latter was a traditionally used benchmark in survey sampling. Beside to the computation of $rRMSE_{\pi_k}$ and CD, the population total of Y was estimated by the expansion estimator $\hat{Y}$. We computed the relative Root Mean Square Error for the estimator, which is defined as $rRMSE_{\hat{Y}} = \frac{\sqrt{\frac{\sum_{m=1}^{M}(\hat{Y_m}-Y)^2}{M}}}{Y}$. Table 6 shows the results of $rRMSE_{\pi_k}$, CD, and $rRMSE_{\hat{Y}}$, obtained by selecting samples with the three sampling methods considered.

The results of $rRMSE_{\pi_k}$ and CD were in line with those seen in previous simulations. Indeed, the sabs algorithm demonstrated greater efficiency in respecting balancing constraints, while also performing worse in the respect of inclusion proba-

Table 6 Results of the Monte Carlo experiment on the Belgian municipalities data for three sampling fractions f = 0.01; 0.05; 0.1. Monte Carlo replicates $M = 10000$

	f	*sabs*	*cube*	*srswor*
$rRMSE_{\pi_k}$	0.01	0.7109	0.1029	0.0932
	0.05	0.3149	0.0437	0.0442
	0.1	0.1466	0.0310	0.0307
CD	0.01	0.0098	0.3353	0.4561
	0.05	0.0022	0.1378	0.2194
	0.1	0.0010	0.0892	0.1594
$rRMSE_{\hat{Y}}$	0.01	0.1348	0.4794	0.5792
	0.05	0.0354	0.1685	0.2558
	0.1	0.0268	0.1053	0.1776

bilities compared to srswor and the cube method. Clearly srswor shows considerable superiority compared to other balanced sampling methods, but this advantage is outclassed by its relevant inferiority in producing efficient estimates of a target variable. In fact, the ability of the sabs and cube method to efficiently estimate the Y variable is evident, especially in regard to the former method. Hence, the efficiency of the proposed algorithm has been showed once again, together with its expendability in practical contexts.

6 Dicussion and conclusions

In the present paper, a new method of selecting balanced samples by means of a simulated annealing-based algorithm has been proposed. By exploiting the possibility of interpreting balanced sampling as an optimization problem, we showed that the sabs algorithm allows for a quicker selection of well-balanced samples on a wider set of auxiliary variables, not neglecting a rigorous respect of inclusion probabilities and a strong efficiency in estimation. An in-depth investigation about both aspects is necessary in order to prove the employability of a sampling method in practical situations. In fact, a good respect of balancing constraints results in a possible reduction of estimation variance, while the respect of inclusion probabilities means a reduction in estimation bias. Hence, a crucial issue concerns finding a sampling method capable of combining these two desirable properties. In the present paper, we proved through extensive simulation experiments, both on simulated and real data, that the sabs algorithm achieves this goal, resulting in a valid and efficient alternative to the well-known cube method.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s00180-021-01113-3.

Funding Open access funding provided by Università degli Studi di Trento within the CRUI-CARE Agreement.

References

Aarts EH, van Laarhoven PJ (1987) Simulated annealing: a pedestrian review of the theory and some applications. In Pattern recognition theory and applications, pages 179–192. Springer

Ardilly P (1991) Échantillonnage représentatif optimum à probabilités inégales. Annales d'Economie et de Statistique 91–113

Benedetti R, Piersimoni F (2017) A spatially balanced design with probability function proportional to the within sample distance. Biom J 59(5):1067–1084

Besag J (1986) On the statistical analysis of dirty pictures. J R Stat Soc Ser B 48(3):259–279

Brus DJ (2015) Balanced sampling: a versatile sampling approach for statistical soil surveys. Geoderma 253:111–121

Chauvet G (2017) A comparison of pivotal sampling and unequal probability sampling with replacement. Stat Prob Lett 121:1–5

Chauvet G, Le Gleut R (2019) Inference under pivotal sampling: properties, variance estimation, and application to tesselation for spatial sampling. Scand J Stat 48:108

Chauvet G, Tillé Y (2006) A fast algorithm for balanced sampling. Comput Stat 21(1):53–62

Deville J-C, Tillé Y (1998) Unequal probability sampling without replacement through a splitting method. Biometrika 85(1):89–101

Deville J-C, Tillé Y (2004) Efficient balanced sampling: the cube method. Biometrika 91(4):893–912

Falorsi PD, Righi P (2008) A balanced sampling approach for multi-way stratification designs for small area estimation. Survey Methodol 34(2):223–234

Geman S, Geman D (1984) Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. IEEE Trans Pattern Anal Mach Intell 6:721–741

Gini C (1928) Une application de la methode representative aux materiaux du dernier recensement de la population italienne (ler decembre 1921). Bull Int Stat Inst 23(2):198–215

Grafström A, Lisic J (2016) Balancedsampling: balanced and spatially balanced sampling. R package version 1(2):

Grafström A, Tillé Y (2013) Doubly balanced spatial sampling with spreading and restitution of auxiliary totals. Environmetrics 24(2):120–131

Hájek J (1981) Sampling from a finite population. Marcel Dekker, New York

Horvitz DG, Thompson DJ (1952) A generalization of sampling without replacement from a finite universe. J Am Stat Ass 47(260):663–685

Kermorvant C, Damico F, Bru N, Caill-Milly N, Robertson B (2019) Spatially balanced sampling designs for environmental surveys. Environ Monit Assess 191(8):524

Kirkpatrick S, Gelatt CD, Vecchi MP et al (1983) Optimization by simulated annealing. Science 220(4598):671–680

Marazzi A, Tillé Y (2017) Using past experience to optimize audit sampling design. Rev Quant Financ Account 49(2):435–462

Narain R (1951) On sampling without replacement with varying probabilities. J Indian Soc Agric Stat 3:169–174

Robert C, Casella G (2013) Monte Carlo statistical methods. Springer Science & Business Media, Berlin

Thionnet P (1953) La théorie des sondages. INSEE, Imprimerie Nationale

Tillé Y (2006) Sampling algorithms. Springer-Verlag, New York

Tillé Y (2011) Ten years of balanced sampling with the cube method: an appraisal. Surv Methodol 37(2):215–226

Tillé Y, Dickson MM, Espa G, Giuliani D (2018) Measuring the spatial balance of a sample: a new measure based on morans i index. Sp Stat 23:182–192
Tillé Y, Matei A (2009) Sampling: survey sampling. R package version, 2
Valliant R, Dorfman AH, Royall RM (2000) Finite population sampling and inference: a prediction approach. Wiley, New York
Yates F (1946) A review of recent statistical developments in sampling and sampling surveys. J R Stat Soc 109(1):12–43

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH ("Springer Nature").

Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users ("Users"), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use ("Terms"). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.

These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.

We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.

While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.

These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.

To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.

Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.

If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com