Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Langevin-Gradient Rerandomization

Antônio Carlos Herling Ribeiro Junior

ahribeir@andrew.cmu.edu

Carnegie Mellon University

Research Article

Keywords: rerandomization, design of experiments, causal inference, stochastic gradient descent

Posted Date: April 23rd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9473741/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Langevin-Gradient Rerandomization

Antônio Carlos Herling Ribeiro Junior[1]

[1]Department of Statistics and Data Science, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, 15213, PA, United States.

Contributing authors: ahribeir@andrew.cmu.edu;

Abstract

Rerandomization is an experimental design technique that repeatedly randomizes treatment assignments until covariates are balanced between treatment groups. Rerandomization in the design stage of an experiment can lead to many asymptotic benefits in the analysis stage, such as increased precision, increased statistical power for hypothesis testing, reduced sensitivity to model specification, and mitigation of p-hacking. However, the standard implementation of rerandomization via rejection sampling faces a severe computational bottleneck in high-dimensional settings, where the probability of finding an acceptable randomization vanishes. Although alternatives based on Metropolis-Hastings and constrained optimization techniques have been proposed, these alternatives rely on discrete procedures that lack information from the gradient of the covariate balance metric, limiting their efficiency in high-dimensional spaces. We propose Langevin-Gradient Rerandomization (LGR), a new sampling method that mitigates this dimensionality challenge by navigating a continuous relaxation of the treatment assignment space using Stochastic Gradient Langevin Dynamics. We discuss the trade-offs of this approach, specifically that LGR samples from a non-uniform distribution over the set of balanced randomizations. We demonstrate how to retain valid inference under this design using randomization tests and empirically show that LGR generates acceptable randomizations orders of magnitude faster than current rerandomization methods in high dimensions.

Keywords: rerandomization, design of experiments, causal inference, stochastic gradient descent

Acknowledgments

The author would like to thank Zach Branson for helpful comments that improved the presentation of this work.

1 Introduction

Randomized controlled trials are widely regarded as the "gold standard" for estimating causal effects in a wide range of disciplines (Mosteller 2002; Carlin et al. 2007; Druckman et al. 2012; Athey and Imbens 2017; List 2024). While complete randomization ensures that treatment and control groups are balanced on both observed and unobserved covariates on average, it does not guarantees balance in any single realization (Rubin 2008; Rosenberger and Sverdlov 2008). In finite samples, chance imbalances in baseline covariates can significantly inflate the variance of treatment effect estimators and reduce the power of statistical tests (Morgan and Rubin 2012). Moreover, with as the number of covariates increase, the higher the probability of imbalance between treatment groups (Krieger et al. 2019; Morgan and Rubin 2012). To address such imbalances, other design of experiments can be considered, such as stratified (Miratrix et al. 2013; Tabord-Meehan 2022), blocked (Pashley and Miratrix 2021, 2022), matched-pair (Imai 2008; Bai 2022), and adaptive designs (Rosenberger and Lachin 2002; Hu and Rosenberger 2006). Another alternative is through rerandomization—a design strategy that repeatedly discards assignments failing to meet a pre-specified covariate balance criterion—has emerged as a powerful tool to improve estimation and inference efficiency (Morgan and Rubin 2012; Li et al. 2018; Li and Ding 2020).

The concept of rerandomization has been commented on since the 1950s (Grundy and Healy 1950; Jones 1958; Savage 1962) and has been used by many since then (Bruhn and McKenzie 2009) although it has not been formalized until the work of Morgan and Rubin (2012). After Morgan and Rubin (2012), rerandomization has been extended to many experimental designs, including 2K factorial, stratified, sequential, survey, and split plot (Branson et al. 2016; Li et al. 2020; Zhou et al. 2018; Yang et al. 2021; Shi et al. 2024). Using rerandomization in the design stage of an experiment has been shown to lead to many asymptotic benefits in the analysis stage, such as increased precision (Morgan and Rubin 2012; Li et al. 2018; Li and Ding 2020; Wang and Li 2024, 2025), increased statistical power of the hypothesis test (Branson et al. 2024), reduced sensitivity to model specification (Zhao and Ding 2024), and thus mitigation of p-hacking (Lu and Ding 2025).

However, rerandomization is typically implemented by acceptance-rejection sampling (Morgan and Rubin 2012; Li et al. 2018; Ribeiro Junior and Branson 2025). This implementation faces a severe "curse of dimensionality." As the number of covariates increases, the probability of a random assignment satisfying the balance criterion vanishes exponentially, making the search for a valid allocation computationally prohibitive for even moderately high-dimensional settings.

Recent work has sought to mitigate this bottleneck using more sophisticated sampling techniques. Pair-Switching Rerandomization (PSRR) (Zhu and Liu 2023) employs a Markov Chain Monte Carlo approach, iteratively swapping treatment assignment between pairs of units until the randomization is deemed balanced. While effective in low-dimensional settings, PSRR essentially operates as a local random walk with a fixed step size. In high-dimensional spaces, where the set of balanced randomizations represents a small region of the discrete hypercube, such local searches often fail to find acceptable assignments in a reasonable timeframe. On the other hand, Balanced Randomization via Integer Programming (BRAIN), a constrained optimization

approach proposed by Lu et al. (2025), can be much faster in high dimensions, but remains restricted to discrete moves in the treatment assignment space, which prevents it from directly exploiting gradient information of the covariate imbalance metric.

In this paper, we propose Langevin Gradient Rerandomization (LGR). Our approach shifts the problem from a discrete to a continuous sampling task. By relaxing the binary treatment assignment into a continuous latent space, we utilize Stochastic Gradient Langevin Dynamics (SGLD) to follow the gradient of the covariate imbalance measure toward the set of balanced randomizations. Unlike the "blind" search of rejection sampling, the "random walk" of PSRR, or the discrete optimization of BRAIN, LGR uses the gradient of the covariate imbalance metric to guide the sampling process.

We make two primary contributions. First, we prove that LGR leads to an unbiased and more precise difference-in-means treatment effect estimator despite sampling non-uniformly from the set of balanced randomizations. Since we sample non-uniformly on the balanced set, we use Fisher randomization tests to conduct finite-sample valid inference. These properties align with the finite-sample inference guarantees established for PSRR and BRAIN. Second, we empirically show that LGR samples balanced randomizations orders of magnitude faster than existing methods as the dimension size increases.

The rest of this paper is divided as follows. In Section 2 we cover the basics of rerandomization. Next, in Section 3 we define LGR, provide its statistical properties, and explain how to conduct finite-sample inference. We present our empirical results in Section 4. We conclude in Section 5.

2 Setup and Notation

Consider a randomized experiment with n units, where n_1 units are assigned to treatment and $n_0 = n - n_1$ to control, with $r_z = n_z/n, z = 0, 1$ the proportion of units under arm z. We observe a matrix of covariates $X = (X_1, \ldots, X_d)$, where $X_j \in \mathbb{R}^n$. The treatment assignment vector is $\mathcal{Z} = (Z_1, \ldots, Z_n)'$, with $Z_i = 1$ for treatment and $Z_i = 0$ for control. Under the potential outcomes framework Rubin (1974), let $Y_i(1)$ and $Y_i(0)$ be the potential outcomes for unit i under treatment and control, and their respective vectors $Y(1) = (Y_1(1), \ldots, Y_n(1))'$ and $Y(0) = (Y_1(0), \ldots, Y_n(0))'$. The observed outcome for unit i is $Y_i = Y_i(1)Z_i + Y_i(0)(1 - Z_i)$. The sole source of randomness is treatment assignment Z_i, while the covariates and potential outcomes are fixed. Ultimately, the goal is to estimate the Average Treatment Effect (ATE), defined by $\tau = \frac{1}{n}\sum_{i=1}^{n} \tau_i = \frac{1}{n}\sum_{i=1}^{n} Y_i(1) - Y_i(0)$, where τ_i are the individual treatment effects. Following most of the rerandomization literature, we use the difference-in-means estimator to estimate the ATE

$$\widehat{\tau} = \frac{1}{n_1}\sum_{i=1}^{n} Y_i Z_i - \frac{1}{n_0}\sum_{i=1}^{n} Y_i(1 - Z_i).$$

Rerandomization ensures balance between the treatment and control groups with respect to the observed covariates. Covariate balance can be measured in various ways, but following most of the rerandomization literature, we focus on the Mahalanobis

distance

$$M = \widehat{\tau}_{\mathrm{X}}' \Sigma^{-1} \widehat{\tau}_{\mathrm{X}}$$

where

$$\widehat{\tau}_{\mathrm{X}} = \left(\overline{X}_1 - \overline{X}_0\right) = \frac{1}{n_1} X'Z - \frac{1}{n_0} X'(1_n - Z)$$

with $1_n \in \mathbb{R}^n$ is a vector whose coefficients are all equal to one, and $\Sigma = \mathrm{cov}\left(\widehat{\tau}_{\mathrm{X}}\right)$. Under complete randomization, we can write $\Sigma = \frac{n}{n_1 n_0} S_X^2$ where $S_X^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \overline{X})(X_i - \overline{X})'$ is the sample covariance matrix of the covariates.

In a rerandomized design, a randomization Z is deemed balanced only if $M \leq a$, where $a > 0$ is a pre-specified threshold. Hence, we define the set of balanced randomizations as $\mathcal{Z}_a = \{Z : M \leq a\}$. Asymptotically, the Mahalanobis distance follows a χ_d^2 distribution (Morgan and Rubin 2012). Therefore, it is common to set a based on a acceptance probability $p_a = \mathbb{P}(\chi_d^2 \leq a)$, where p_a is commonly chosen to be 0.01 or 0.001.

3 Langevin-Gradient Rerandomization

3.1 The LGR Algorithm

To address the computational bottleneck of finding a balanced assignment $Z \in \mathcal{Z}_a$ in high-dimensional settings, we propose Langevin-Gradient Rerandomization (LGR). Unlike rejection sampling, which blindly draws randomizations from all possible treatment assignments, LGR utilizes the gradient of the Mahalanobis distance with respect to a continuous relaxation of the treatment assignment to actively guide the sampling towards the set of balanced randomizations $\mathcal{Z}_a$.

We introduce a vector of latent scores $\theta \in \mathbb{R}^n$, which are mapped to "soft" assignments $\tilde{z} \in (0,1)^n$ via a temperature-scaled logistic function:

$$\tilde{z}_i(\theta_i) = \sigma_\delta(\theta_i) = \frac{1}{1 + \exp(-\theta_i/\delta)} \tag{1}$$

where $\delta > 0$ controls the smoothness of the relaxation. This relaxation allows us to define a differentiable "soft" Mahalanobis distance, which is the Mahalanobis distance calculated with respect to the soft assignments $\tilde{z}$. The gradient of M with respect to the latent scores is calculated via the chain rule:

$$\frac{\partial M}{\partial \theta_i} = \frac{\partial M}{\partial \tilde{z}_i} \cdot \frac{\partial \tilde{z}_i}{\partial \theta_i}, \tag{2}$$

where the derivative of the sigmoid is $\frac{\partial \tilde{z}_i}{\partial \theta_i} = \frac{1}{\delta} \tilde{z}_i (1 - \tilde{z}_i)$.

The gradient of M with respect to the soft weights $\tilde{z}_i$, after applying the quotient rule to the difference-in-means terms, is:

$$\frac{\partial M}{\partial \tilde{z}_i} = 2 \left[\widehat{\Sigma}^{-1} \Delta(\tilde{z})\right]^\top \left[\frac{1}{n_{\tilde{1}}}(X_i - \overline{X}_{\tilde{1}}) + \frac{1}{n_{\tilde{0}}}(X_i - \overline{X}_{\tilde{0}})\right], \tag{3}$$

where $n_{\tilde{1}} = \sum \tilde{z}_i$ and $n_{\tilde{0}} = \sum(1 - \tilde{z}_i)$ are the soft treatment group sizes, and $\overline{X}_{\tilde{1}}$ and $\overline{X}_{\tilde{0}}$ are the covariate means with respect to the soft assignments.

The core of LGR, detailed in Algorithm 1, is the iterative evolution of the latent scores θ using Stochastic Gradient Langevin Dynamics (SGLD). The algorithm is initialized with $\theta^{(0)} \sim N(0, I_n)$. At each iteration t, we update the scores according to:

$$\theta^{(t)} \leftarrow \theta^{(t-1)} - \eta \nabla_\theta M \left(\theta^{(t-1)}\right) + \sqrt{2\eta\delta}\xi_t$$

where $\eta > 0$ is the learning rate and $\xi_t \sim N(0, I_n)$ is standard Gaussian noise. This update rule consists of two competing forces: the gradient term $-\eta\nabla_\theta M$ drives the latent scores toward values that minimize covariate imbalance, while the noise term $\sqrt{2\eta\delta}\xi_t$ injects stochasticity. This stochastic component is crucial; it prevents the algorithm from collapsing into a deterministic optimization and prohibiting randomization-based inference.

Algorithm 1 Langevin-Gradient Rerandomization

Input: Covariates X, Number of treated units n_1, Balance threshold a, Temperature δ (default value = 0.5), Learning rate η (default value = 1)
Output: A balanced assignment Z where $\sum Z_i = n_1$ and $M \leq a$

Initialize: $\theta^{(0)} \sim N(0, I_n)$
Pre-compute $\widehat{\Sigma}^{-1}$
while $M > a$ **do**
    1. Discrete Projection Check
    Let $\mathcal{S}$ be the indices of the n_1 largest elements of $\theta^{(t-1)}$
    Construct binary vector Z: $Z_i = 1$ if $i \in \mathcal{S}$, else 0
    Compute M
    if $M \leq a$ **then**
        return Z ▷ Found balanced randomization
    end if
    2. Soft Relaxation
    $\tilde{z} \leftarrow \sigma_\delta(\theta^{(t-1)})$
    $n_{\tilde{1}} \leftarrow \sum \tilde{z}_i, \quad n_{\tilde{0}} \leftarrow n - n_{\tilde{1}}$
    3. Gradient Computation
    $\Delta \leftarrow \left(\frac{1}{n_{\tilde{1}}} X'\tilde{z}\right) - \left(\frac{1}{n_{\tilde{0}}} X'(\mathbb{1}_n - \tilde{z})\right)$
    $g_M \leftarrow 2 \cdot \widehat{\Sigma}^{-1}\Delta$
    Scaling vector $\gamma \in \mathbb{R}^n$ where $\gamma_i = \frac{1}{\delta}\tilde{z}_i(1 - \tilde{z}_i)$
    $\nabla_\theta \leftarrow$ (Jacobian of M w.r.t $\tilde{z}$) $\cdot\, \gamma$ ▷ Using equation (2)
    4. Update Step (SGLD)
    $\xi_t \sim N(0, I_n)$
    $\theta^{(t)} \leftarrow \theta^{(t-1)} - \eta\nabla_\theta + \sqrt{2\eta\delta}\xi_t$
end while

While the SGLD updates occur in the continuous latent space, our goal is a binary assignment. We construct a candidate binary assignment Z by assigning the treatment to the units corresponding to the n_1 largest elements of $\theta^{(t)}$. If this candidate Z satisfies the balance criterion $M \leq a$, the algorithm terminates and returns Z.

The efficiency of LGR relies on the choice of the temperature δ and the learning rate η. We set the default temperature to $\delta = 0.5$. Since the latent scores are initialized from a standard normal distribution, $\theta^{(0)} \sim N(0, I_n)$, the input to the sigmoid function θ/δ effectively follows a distribution with variance $1/\delta^2 = 4$. This scaling spreads the initial scores across the active domain of the logistic function, preventing the soft assignments from saturating at 0 or 1 too early, which would cause the gradients to vanish. We set the default learning rate to $\eta = 1$. This value provides a balanced step size that allows for convergence toward the balanced region while maintaining sufficient variance in the Langevin noise to ensure valid coverage of the assignment space.

3.2 Statistical properties of LGR

Assumption 1 $n_1 = n_0 = n/2$.

Assumption 2 $Y_i(Z_i) = \beta_0 + \beta' X_i + \tau Z_i + \epsilon_i$, *where* $\beta_0 + \beta' X_i$ *is the linear projection of* $Y_i(0)$ *onto* $(1, X)$ *and* ϵ_i *is the deviation from the linear projection.*

Assumption 3 $\widehat{\tau}$ *and* $\overline{X}_1 - \overline{X}_0$ *are normally distributed.*

Theorem 4 (Unbiasedness) *Under Assumption 1 and LGR, we have that the difference-in-means estimator is unbiased:* $\mathbb{E}[\hat{\tau}] = \tau$.

Theorem 5 (Variance Reduction) *Under Assumptions 1-3 and LGR, we have that*

$$\frac{Var_{CR}(\widehat{\tau}) - Var_{LGR}(\widehat{\tau})}{Var_{CR}(\widehat{\tau})} \geq \left(1 - \frac{a}{d}\right) R^2$$

where

$$R^2 = \frac{n}{n_1 n_0} \frac{\beta' S_X^2 \beta}{Var_{CR}(\widehat{\tau})},$$

$Var_{CR}(\widehat{\tau})$ *the variance of* $\widehat{\tau}$ *under complete randomization, and* $Var_{LGR}(\widehat{\tau})$ *the variance of* $\widehat{\tau}$ *under LGR.*

Taken together, Theorems 4-5 show that, from a design-based perspective, LGR enjoys the same key finite-sample properties as existing rerandomization schemes such as PSRR and BRAIN. Thus, the main distinction between LGR and these approaches lies in the mechanism used to sample the space of balanced assignments, where LGR replaces discrete local moves with a gradient-guided Langevin sampler.

3.3 Inference under LGR

A key challenge in rerandomization designs that do not sample uniformly from the balanced set $\mathcal{Z}_a$—such as PSRR, BRAIN, and our proposed LGR—is that standard

asymptotic results for rerandomization (e.g., Li et al. (2018)) may not hold. Specifically, because LGR utilizes gradient information to steer the sampling, the resulting distribution of balanced assignments $P(Z|Z \in \mathcal{Z}_a)$ is not necessarily uniform. Consequently, standard theoretical results on asymptotic distributions derived in Morgan and Rubin (2012); Li et al. (2018) are not directly applicable.

To ensure valid inference without relying on asymptotic approximations that may be violated by the non-uniform sampling, we employ Fisher Randomization Tests (FRT). The FRT provides exact finite-sample inference by simulating the distribution of the test statistic under the sharp null hypothesis of no treatment effect, conditional on the specific randomization mechanism employed.

Given the vectors of potential outcomes $Y(1)$ and $Y(0)$, we wish to test the sharp null hypothesis $H_0 : Y_i(1) = Y_i(0)$ for all $i = 1, \ldots, n$. Under H_0, the observed outcomes Y are fixed regardless of the treatment assignment. We define a test statistic: $T(Z, Y)$. After sampling $B \gg 0$ balanced randomizations $Z^{(1)}, \ldots, Z^{(B)} \in \mathcal{Z}_a$, we define the p-value as

$$p = \frac{1}{B+1}\left[1 + \sum_{b=1}^{B} \mathbb{1}\left(|T(Z^{(b)}, Y)| > |T(Z, Y)|\right)\right]$$

where $\mathbb{1}$ is the indicator function.

To construct confidence intervals for the average treatment effect, we invert this test. While the standard FRT tests the sharp null hypothesis of no effect ($H_0 : \tau = 0$), we can extend this to test any constant additive treatment effect hypothesis $H_0(\tau_0)$: $Y_i(1) - Y_i(0) = \tau_0$ for all i, $\tau_0 \in \mathbb{R}$.

Under the hypothesis $H_0(\tau_0)$, the missing potential outcomes are imputed as:

$$Y_i(0) = \begin{cases} Y_i & \text{if } Z_i = 0 \\ Y_i - \tau_0 & \text{if } Z_i = 1 \end{cases} \quad \text{and} \quad Y_i(1) = \begin{cases} Y_i + \tau_0 & \text{if } Z_i = 0 \\ Y_i & \text{if } Z_i = 1 \end{cases}$$

This allows us to construct the full vector of potential outcomes under the null and generate the reference distribution of the test statistic $T(Z, Y(\tau_0))$ by repeatedly applying the LGR algorithm. The $(1 - \alpha)$ confidence interval is defined as the set of all values τ_0 such that the null hypothesis $H_0(\tau_0)$ is not rejected at level α:

$$\text{CI}_{1-\alpha} = \{\tau_0 : p(\tau_0) \geq \alpha\}$$

where $p(\tau_0)$ is the p-value obtained from the FRT under the hypothesized effect τ_0. In practice, this interval is approximated via a grid search over plausible values of τ. While this approach is computationally intensive for standard rejection sampling, the efficiency of LGR renders this inversion feasible even in high-dimensional settings.

4 Simulations

We evaluate the performance of LGR against standard Complete Randomization (CR), Acceptance Rejection Sampling Rerandomization (ARR), Pair-Switching Rerandomization (PSRR), and the BRAIN algorithm. We compare them in terms of (i) computational time (in seconds) to find a balanced randomization – where for CR is simply the time to draw a randomization and serve as a benchmark, (ii) bias and standard deviation of the treatment effect estimator, (iii) coverage and power of confidence intervals. We implement the PSRR algorithm as described in Algorithm 1 of Zhu and Liu (2023)), BRAIN according to Algorithm 1 of Lu and Ding (2025). In both of them, we use the default suggested values for the tuning parameters. We also use the default parameters' values to report the results of LGR. We run the simulations in a MacBook Air (Apple M2 Chip, 24 GB Memory, 8 Cores)[1].

We simulate a setting with d covariates generated from a multivariate normal distribution $X \sim N(0, I_d)$. The outcome follows a linear model $Y_i(Z_i) = \sum_{j=1}^{d} X_{ij} + \tau Z_i + \epsilon_i$, where $\tau = 0.5$ and $\epsilon_i \sim N(0, 1)$. We set $n_1 = n_0 = n/2$ and $n = 500$. For the rerandomization, we consider the acceptance probability $p_a = 0.01$, so the threshold a is the p_a-th quantile of a χ^2_d distribution.

We perform Fisher randomization tests with a significance level $\alpha = 0.05$ and construct confidence intervals with a nominal coverage rate of 95%. To estimate computational runtime and the bias and standard deviation of the treatment effect estimator, we conduct $B = 1000$ independent simulations. Within each simulation, we utilize the specific rerandomization algorithm to generate the balanced treatment assignment. To conduct randomization-based inference and construct confidence intervals, we generate a reference set of $B_{\text{frt}} = 100$ additional balanced randomizations for each simulation to approximate the null distribution. Since PSRR and ARR scale with the dimension size, we do not report any results that depend on randomization-based inference for these methods.

In Figure 1, we consider when d is small. In the left panel, each line considers the average time in seconds to find a balanced randomization by each rerandomization method as a function of the dimension d, or the time to draw a randomization completely at random. The shade around each line is a bootstrap 95% confidence interval calculated from the $B = 1000$ randomizations. The right panel represents the bias of the treatment effect estimator as a function of the dimension size. The lines are the average bias. The shaded region represents the standard deviation of the estimated bias. We find that initially, ARR is the slowest to find a balanced randomization, followed by our proposed method. However, as the dimension increases, PSRR turns into the slowest rerandomization method, and our proposed method is the fastest between the rerandomizations. All of the methods have similar biases and standard deviations of the estimated treatment effect, which are lower that the standard deviation from CR.

We extend this simulation to higher dimension size, and present it in Figure 2. We find that our proposed method is the fastest to find a balanced randomization while PSRR is the slowest. Interestingly, our method presents a U-shape curve in the

[1] All code for the simulations is available in this github repository.

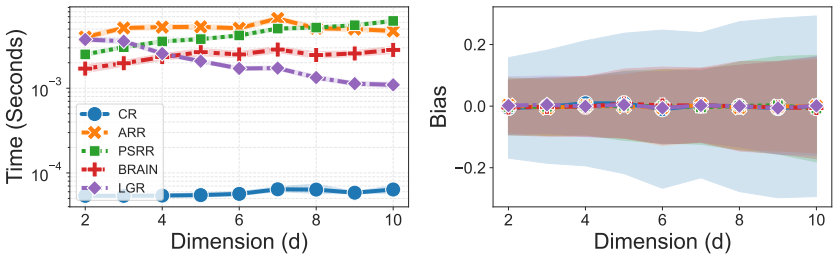


Fig. 1 The left panel represents the time to generate a balanced randomization using each rerandomization method or a randomization completely at random (blue line). The shaded region around each line is a bootstrap 95% confidence interval. The right panel shows the bias (lines) and standard deviation (shaded region) of the difference-in-means treatment effect estimator with each rerandomization method. We find that for the smallest dimensions ARR is the slowest method to find a balanced randomization, followed by our proposed method. However, as the dimension size increases, PSRR turns into the slowest method and LGR is the fastest method.

left plot. This might happen because calculating the gradient of the soft relaxation is an overhead in low dimensions making the algorithm slower, but proves to be computationally more efficient in higher dimensions. On the other hand, PSRR works as a "random walk" on the treatment space with a step of size one (only one unit on each treatment arm is swapped at each iteration). Hence, it takes longer to find the balanced region in the treatment space.

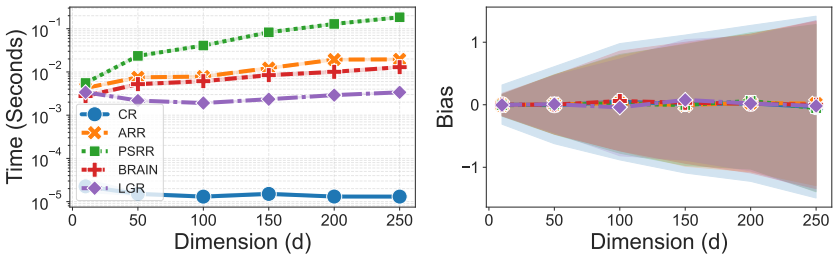


Fig. 2 The left panel represents the time to generate a balanced randomization using each rerandomization method or a randomization completely at random (blue line). The shaded region around each line is a bootstrap 95% confidence interval. The right panel shows the bias (lines) and standard deviation (shaded region) of the difference-in-means treatment effect estimator with each rerandomization method. For the higher dimensions, LGR is the fastest to generate a balanced randomization. Interestingly, it has a U-shape, suggesting that for low dimensions the overhead of calculating gradients, while for it is computationally beneficial to calculate the gradients.

Next, we do randomization-based inference with LGR, BRAIN, and CR and present the results in Figure 3. In the left panel, we show the coverage probability of each method as a function of the dimension size d. Nominal coverage at 95% is denoted by the dashed horizontal black line. While on the right panel, we show the power of each method as a function of the dimension size. Notice that all method achieve nominal coverage, while BRAIN and LGR are more powerful than CR, following the rerandomization literature.

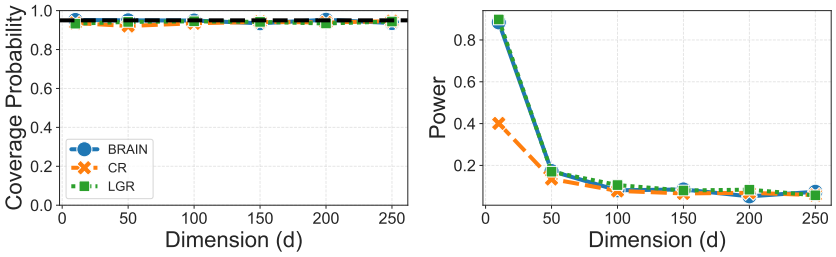


Fig. 3 The left panel represents the coverage probability of each method. Nominal coverage is represented by the dashed horizontal black line at 95%. Meanwhile the right panel shows the power. All methods achieve nominal coverage, while the rerandomization methods are more powerful than complete randomization.

In Appendix B, we conduct a sensitivity analysis of our proposed rerandomization method with respect to its temperature δ and learning rate η parameters. We find that extreme values of the parameters can be detrimental to the rerandomization method and make it slower than the established rerandomization algorithms in the literature.

5 Conclusion

Rerandomization is a powerful tool for improving precision in randomized experiments by achieving covariate balance at the design stage Morgan and Rubin (2012); Li et al. (2018). However, the standard implementation via acceptance–rejection sampling suffers from an exponential computational bottleneck as the dimension of covariates increases Ribeiro Junior and Branson (2025). Although recent alternatives such as PSRR and BRAIN have been proposed to mitigate this curse of dimensionality, these methods rely on discrete procedures—iterative local swaps or constrained optimization—that lack direct guidance from gradient information, limiting their efficiency in high-dimensional spaces.

In this paper, we propose Langevin-Gradient Rerandomization, a novel rerandomization method that exploits a continuous relaxation of the treatment assignment space to navigate toward balanced randomizations via Stochastic Gradient Langevin Dynamics. By reformulating the discrete assignment problem into a continuous optimization landscape, we leverage gradient information about covariate imbalance to guide the search process efficiently. We proved that despite sampling non-uniformly from the set of balanced randomizations, the difference-in-means treatment effect estimator remains unbiased (Theorem 4) and achieves variance reduction comparable to standard rerandomization methods (Theorem 5). To ensure valid finite-sample inference under non-uniform sampling, we employed Fisher Randomization Tests with confidence interval inversion, providing exact hypothesis tests conditional on the LGR sampling mechanism. Extensive empirical simulations across dimensions demonstrate that LGR achieves orders of magnitude speedups over existing methods in high-dimensional settings while maintaining unbiased and precise estimation as previous rerandomization methodologies. LGR leads to nominal coverage inference as powerful as previous rerandomization methods.

Future research directions include extending LGR to more general differentiable covariate balance metrics beyond the Mahalanobis distance, such as quadratic forms (Schindl and Branson 2024). Additionally, LGR could be generalized to other experimental designs where rerandomization is required, such as sequential designs where units arrive over time Zhou et al. (2018) and cluster randomized trials where balance must be maintained within and between clusters (Lu et al. 2023). Further refinements could include adaptive learning strategies to adjust LGR's parameters based on gradient magnitudes and covariance structure, eliminating the need for manual hyperparameter tuning.

Appendix A Proofs

A.1 Proof of Theorem 4

Proof Let Ω denote the set of all possible trajectories (chains) of the latent variable vectors generated by the LGR algorithm: $\left(\theta^{(0)}, ..., \theta^{(t)}, ..., \theta^{(T)}\right)$, where $\theta^{(t)} \in \mathbb{R}^n$. Let π be any permutation of the indices $\{1, ..., n\}$. Consider any realized chain $\vartheta = \left(\vartheta^{(0)}, ..., \vartheta^{(T)}\right) \in \Omega$. We compare the probability of this chain to its negation counterpart $-\vartheta = \left(-\vartheta^{(0)}, ..., -\vartheta^{(T)}\right)$.

The algorithm initializes $\theta^{(0)} \sim N(0, I_n)$. The standard multivariate normal distribution is exchangeable, $\mathbb{P}\left(\theta^{(0)} = \vartheta^{(0)}\right) = \mathbb{P}\left(\theta^{(0)} = -\vartheta^{(0)}\right)$.

The transition from $\theta^{(t-1)}$ to $\theta^{(t)}$ is governed by the SGD update:

$$\theta^{(t)} = \theta^{(t-1)} - \nabla J\left(\theta^{(t-1)}\right) + \xi_t \quad \xi_t \sim N(0, I_n).$$

The objective function $M(\theta)$ (soft Mahalanobis distance) is a Mahalanobis distance constructed from a sigmoid function σ. Hence, $M(\theta)$ is an even function, and its gradient is an odd function, meaning $\nabla M(-\theta) = -\nabla M(\theta)$.

As in the initialization of he algorithm, since the noise ξ_t is also isotropic (exchangeable), the transition probability density satisfies:

$$\mathbb{P}\left(\theta^{(t)} = \vartheta^{(t)} | \theta^{(t-1)} = \vartheta^{(t-1)}\right) = \mathbb{P}\left(\theta^{(t)} = -\vartheta^{(t)} | \theta^{(t-1)} = -\vartheta^{(t)}\right)$$

Combining all of these

$$\begin{aligned}
\mathbb{P}(\text{Chain} = \vartheta) &= \mathbb{P}\left(\theta^{(0)}\right) \prod_{t=1}^{T} \mathbb{P}\left(\theta^{(t)} | \theta^{(t-1)}\right) \\
&= \mathbb{P}\left(-\theta^{(0)}\right) \prod_{t=1}^{T} \mathbb{P}\left(-\theta^{(t)}) | -\theta^{(t-1)}\right) \\
&= \mathbb{P}(\text{Chain} = -\vartheta)
\end{aligned}$$

This implies that for every trajectory the algorithm takes, the negative trajectory is equally likely. The final assignment Z is a deterministic function of the final state $\theta^{(T)}$ (specifically, $Z_i = 1$ if $\theta_i^{(T)}$ is among the top n_1 values). If $\theta_i^{(T)}$ is in the top n_1, then the i-th element of $-\theta^{(T)}$ is in the bottom n_1. Thus, $Z\left(-\theta^{(T)}\right) = 1 - Z\left(\theta^{(T)}\right)$ and the marginal distribution of the final assignment Z must be

$$\mathbb{P}(Z = z) = 1 - \mathbb{P}(Z = z)$$

Summing over all possible permutations π, the marginal probability that any specific unit i is assigned to treatment must be identical for all units:

$$\mathbb{P}(Z_i = 1) = \mathbb{P}(Z_i = 0) = \frac{1}{2}, \forall i = 1, \ldots, n$$

Finally, with $\mathbb{E}(Z_i) = \mathbb{P}(Z_i = 1) = 1/2$, the expectation of the difference-in-means estimator is:

$$\begin{aligned}
\mathbb{E}[\hat{\tau}] &= \mathbb{E}\left[\frac{2}{n}\sum_{i=1}^{n} Z_i Y_i(1) - \frac{2}{n}\sum_{i=1}^{n}(1 - Z_i)Y_i(0)\right] \\
&= \frac{2}{n}\sum_{i=1}^{n} Y_i(1)\mathbb{E}(Z_i) - \frac{2}{n}\sum_{i=1}^{n} Y_i(0)\mathbb{E}(1 - Z_i) \\
&= \frac{1}{n}\sum_{i=1}^{n}(Y_i(1) - Y_i(0)) = \tau.
\end{aligned}$$

□

A.2 Proof of Theorem 5

Proof We can define

$$W = \sqrt{\frac{n}{n_1 n_0}} S_X^{-1}\left(\overline{X}_1 - \overline{X}_0\right) = \sqrt{\frac{n}{n_1 n_0}} S_X^{-1} X'\left(Z - \frac{n_1}{n}\mathbb{1}_n\right)$$

and $M = W'W = \sum_{j=1}^{d} W_j^2$. Therefore, under Assumption 1, $\mathbb{E}(W) = 0$ and $\mathbb{E}(W_j) = 0, \forall j = 1, \ldots, d$.

Note that if we exchange any W_i and W_j, M does not change. As a result, we have

$$\mathrm{Var}(W_j) = \mathbb{E}(W_j^2) = \frac{1}{d}\mathbb{E}\left[\sum_{j=1}^{d} W_j^2\right] = \frac{1}{d}\mathbb{E}(M) \leq \frac{a}{d}.$$

Moreover, if we change the sign of any W_j, the Mahalanobis distance M does not change. Hence this does not affect the distribution of W, and the distribution of W_j. Thus, $W_j|W_i$ and $-W_j|W_i$ are identically distributed. This implies,

$$\mathrm{cov}(W_j, W_i) = \mathbb{E}(W_j W_i) = \mathbb{E}\left[\mathbb{E}\left[W_j W_i | W_i\right]\right] = \mathbb{E}\left[W_i \mathbb{E}\left[W_j | W_i\right]\right] = \mathbb{E}\left[W_i \times 0\right] = 0$$

Therefore, $\mathrm{cov}(W) = \mathrm{Var}(W_j) I_d$,

$$\begin{aligned}
\mathrm{cov}\left(\overline{X}_1 - \overline{X}_0\right) &= \mathrm{cov}\left(\sqrt{\frac{n_1 n_0}{n}} S_X W\right) \\
&= \frac{n_1 n_0}{n} S_X \mathrm{cov}(W) S_X \\
&= \frac{n_1 n_0}{n} S_X \mathrm{Var}(W_j) S_X \\
&\leq \frac{a}{d}\frac{n_1 n_0}{n} S_X^2 \\
&= \frac{a}{d}\mathrm{Var}_{\mathrm{CR}}(\widehat{\tau})
\end{aligned}$$

and $\beta' \mathrm{cov}\left(\overline{X}_1 - \overline{X}_0\right)\beta \leq \frac{a}{d} R^2 \mathrm{Var}_{\mathrm{CR}}(\widehat{\tau})$.

Finally, by Assumption 2 the difference-in-means estimator can be expressed as

$$\widehat{\tau} = \tau + \beta'\left(\overline{X}_1 - \overline{X}_0\right) + \left(\overline{\epsilon}_1 - \overline{\epsilon}_0\right)$$

where $\epsilon_1 = \sum_{i:Z_i=1} \epsilon_i/n_1$ and $\epsilon_0 = \sum_{i:Z_i=0} \epsilon_i/n_0$. Since $\beta_0 + \beta' X_i$ is the projection of $Y_i(0)$ onto $(1, X)$, $\overline{X}_1 - \overline{X}_0$ and $\bar{\epsilon}_1 - \bar{\epsilon}_0$ are uncorrelated. By Assumption 3, $\widehat{\tau}$ and $\overline{X}_1 - \overline{X}_0$ are normally distributed, so $\overline{X}_1 - \overline{X}_0$ and $\bar{\epsilon}_1 - \bar{\epsilon}_0$ are independent. Since LGR does not affect $\bar{\epsilon}_1 - \bar{\epsilon}_0$, we have

$$\begin{aligned}
\mathrm{Var}_{\mathrm{LGR}}(\widehat{\tau}) &= \beta' \mathrm{cov}\left(\overline{X}_1 - \overline{X}_0\right) \beta + \mathrm{Var}\left(\bar{\epsilon}_1 - \bar{\epsilon}_0\right) \\
&\leq \frac{a}{d} R^2 \mathrm{Var}_{\mathrm{CR}}(\widehat{\tau}) + (1 - R^2) \mathrm{Var}_{\mathrm{CR}}(\widehat{\tau}) \\
&= \left[1 - \left(1 - \frac{a}{d}\right) R^2\right] \mathrm{Var}_{\mathrm{CR}}(\widehat{\tau})
\end{aligned}$$

Hence,

$$\frac{\mathrm{Var}_{\mathrm{CR}}(\widehat{\tau}) - \mathrm{Var}_{\mathrm{LGR}}(\widehat{\tau})}{\mathrm{Var}_{\mathrm{CR}}(\widehat{\tau})} \geq \left(1 - \frac{a}{d}\right) R^2.$$

□

Appendix B Additional Simulations

In addition to the simulation results in Section 4, we conduct a sensitivity analysis to understand whether the performance of LGR is affected by the choice of its parameters.

Both panels in the left side of Figure B1 show the results for LGR when varying its temperature $\delta \in \{0.01, 0.1, 0.5, 1.0, 10.0\}$, while considering the default value for the learning rate $\eta = 1$, and compare to the performance of BRAIN with its default parameters. The lines on the top left panel show the time in seconds for each setting to sample a balanced randomization, with its shaded region corresponding to a bootstrap 95% confidence interval. Notice that for extreme values of δ, LGR's performance is undermined, because the soft assignments defined in Equation (1) take extreme values, making the gradient in Equation (3) unstable and uninformative. Meanwhile, the bottom left panel shows the bias (lines) and standard deviation (shaded region) of the difference-in-means treatment effect estimator for each setting of the rerandomization methods. Note that all of them are similar, and hence the temperature does not seem to affect it.

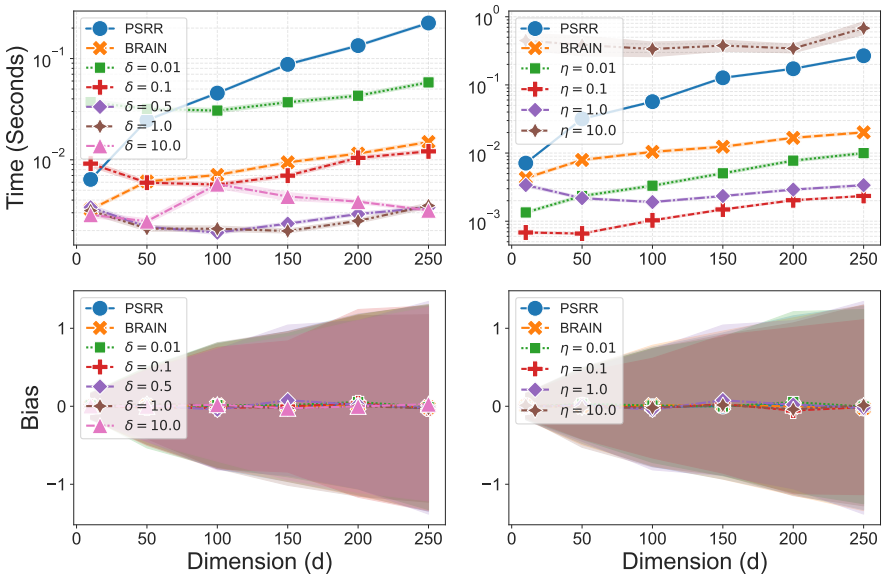


Fig. B1 Sensitivity analysis plot of the LGR method with respect to its temperature δ and η. Both panels in the left side show the results for LGR when varying its temperature $\delta \in \{0.01, 0.1, 0.5, 1.0, 10.0\}$, while considering the default value for the learning rate $\eta = 1$, and BRAIN method with its default parameters. The panels on the right side consider a fixed temperature at its default value $\delta = 0.5$, but varying $\eta \in \{0.01, 0.1, 1.0, 10.0\}$.The lines on the top panels show the time in seconds for each setting to sample a balanced randomization, with its shaded region corresponding to a bootstrap 95% confidence interval. Notice that for extreme values of δ and large values of η, LGR's performance is undermined. Meanwhile, the bottom right panel shows the bias (lines) and standard deviation (shaded region) of the difference-in-means treatment effect estimator for each setting of the rerandomization methods. Note that all of them are similar, and hence the parameters values do not seem to affect it.

The other two panels on the right side of Figure B1 show similar results. In this case, they consider LGR with default temperature $\delta = 0.5$ but varying learning rate $\eta \in \{0.01, 0.1, 1.0, 10.0\}$. The lines in the top right panel show the time in seconds for each setting to sample a balanced randomization, with its shaded region corresponding to a bootstrap 95% confidence interval. Notice that for large values of η, LGR's performance is undermined, this is because the SGLD update step is taking large step sizes, which ultimately does not exploit the gradient information. Meanwhile, the bottom right panel shows the bias (lines) and standard deviation (shaded region) of the difference-in-means treatment effect estimator for each setting of the rerandomization methods. Note that all of them are similar, and hence the learning rate does not seem to affect it.

References

Athey, S., Imbens, G.W.: Chapter 3 - the econometrics of randomized experiments. Handbook of Economic Field Experiments **1**, 73–140 (2017) https://doi.org/10.1016/bs.hefe.2016.10.003

Bai, Y.: Optimality of matched-pair designs in randomized controlled trials. American Economic Review **112**, 3911–3940 (2022) https://doi.org/10.1257/aer.20201856

Branson, Z., Dasgupta, T., Rubin, D.B.: Improving covariate balance in 2^K factorial designs via rerandomization with an application to a New York City Department of Education High School Study. The Annals of Applied Statistics **10**, 1958–1976 (2016) https://doi.org/10.1214/16-AOAS959

Branson, Z., Li, X., Ding, P.: Power and sample size calculations for rerandomization. Biometrika **111**, 355–363 (2024) https://doi.org/10.1093/biomet/asad027

Bruhn, M., McKenzie, D.: In pursuit of balance: Randomization in practice in development field experiments. American Economic Journal: Applied Economics **1**, 200–232 (2009) https://doi.org/10.1257/app.1.4.200

Carlin, B.P., Cook, T.D., Faraway, J.J., Zidek, J., Tanner, M.A., DeMets, D.L.: Introduction to Statistical Methods for Clinical Trials. Chapman & Hall/CRC, ??? (2007)

Druckman, J.N., Green, D.P., Kuklinski, J.H., Lupia, A.: Cambridge Handbook of Experimental Political Science. Cambridge University Press, ??? (2012)

Grundy, P.M., Healy, M.J.R.: Restricted randomization and quasi-latin squares. Journal of the Royal Statistical Society. Series B (Methodological) **12**, 286–291 (1950) https://doi.org/10.1111/j.2517-6161.1950.tb00062.x

Hu, F., Rosenberger, W.F.: The Theory of Response-Adaptive Randomization in Clinical Trials. John Wiley & Sons, ??? (2006)

Imai, K.: Variance identification and efficiency analysis in randomized experiments under the matched-pair design. Statistics in Medicine **27**, 4857–4873 (2008) https://doi.org/10.1002/sim.3337

Jones, H.L.: Inadmissible samples and confidence limits. Journal of the American Statistical Association **53**, 482–490 (1958) https://doi.org/10.1080/01621459.1958.10501453

Krieger, A.M., Azriel, D., Kapelner, A.: Nearly random designs with greatly improved balance. Biometrika **106**, 695–701 (2019) https://doi.org/10.1093/biomet/asz026

Li, X., Ding, P.: Rerandomization and regression adjustment. Journal of the Royal Statistical Society Series B: Statistical Methodology **82**, 241–268 (2020) https://doi.org/10.1111/rssb.12353

Lu, X., Ding, P.: Rerandomization for covariate balance mitigates p-hacking in regression adjustment. preprint (2025). arXiv:2505.01137

Li, X., Ding, P., Rubin, D.B.: Asymptotic theory of rerandomization in treatment–control experiments. Proc. Natl. Acad. Sci. U.S.A. **115**, 9157–9162 (2018) https://doi.org/10.1073/pnas.1808191115

Li, X., Ding, P., Rubin, D.B.: Rerandomization in 2^K factorial experiments. The Annals of Statistics **48**, 43–63 (2020) https://doi.org/10.1214/18-AOS1790

List, J.A.: Experimental Economics: Theory and Practice. The University of Chicago Press, ??? (2024)

Lu, X., Liu, T., Liu, H., Ding, P.: Design-based theory for cluster rerandomization. Biometrika **110**, 467–483 (2023) https://doi.org/10.1093/biomet/asac045

Lu, J., Liu, D., Lin, Z., Wang, X.: Fast Rerandomization via the BRAIN. preprint (2025). arXiv:2312.17230

Mosteller, F.: Evidence Matters: Randomized Trials in Education Research. Brookings Institution Press, ??? (2002)

Morgan, K.L., Rubin, D.B.: Rerandomization to improve covariate balance in experiments. Annals of Statistics **40**, 1263–1282 (2012) https://doi.org/10.1214/12-AOS1008

Miratrix, L.W., Sekhon, J.S., Yu, B.: Adjusting treatment effect estimates by post-stratification in randomized experiments. Journal of the Royal Statistical Society Series B: Statistical Methodology **75**, 369–396 (2013) https://doi.org/10.1111/j.1467-9868.2012.01048.x

Pashley, N.E., Miratrix, L.W.: Insights on variance estimation for blocked and matched pairs designs. Journal of Educational and Behavioral Statistics **46**, 271–296 (2021) https://doi.org/10.3102/10769982

Pashley, N.E., Miratrix, L.W.: Block what you can, except when you shouldn't. Journal of Educational and Behavioral Statistics **47** (2022) https://doi.org/10.3102/10769986211027240

Ribeiro Junior, A.C.H., Branson, Z.: Does Rerandomization Help Beyond Covariate Adjustment? A Review and Guide for Theory and Practice. preprint (2025). arXiv:2512.05290

Rosenberger, W.F., Lachin, J.M.: Randomization in Clinical Trials: Theory and Practice. John Wiley & Sons, ??? (2002)

Rosenberger, W.F., Sverdlov, O.: Handling covariates in the design of clinical trials. Statistical Science **23**, 404–419 (2008) https://doi.org/10.1214/08-STS269

Rubin, D.B.: Estimating causal effects of treatments in randomized and nonrandomized studies. Journal of Educational Psychology **66**, 688–701 (1974) https:

//doi.org/10.1037/h0037350

Rubin, D.B.: Comment: The design and analysis of gold standard randomized experiments. Journal of the American Statistical Association **103**, 1350–1353 (2008) https://doi.org/10.1198/016214508000001011

Savage, L.J.: The Foundations of Statistical Inference. John Wiley & Sons Inc., ??? (1962)

Schindl, K., Branson, Z.: A unified framework for rerandomization using quadratic forms. preprint (2024). arXiv:2403.12815

Shi, W., Zhao, A., Liu, H.: Rerandomization and covariate adjustment in split-plot designs. Journal of Business & Economic Statistics, 1–22 (2024) https://doi.org/10.1080/07350015.2024.2429464

Tabord-Meehan, M.: Stratification Trees for Adaptive Randomization in Randomized Controlled Trials. arxiv:1806.05127 (2022)

Wang, B., Li, F.: Asymptotic inference with flexible covariate adjustment under rerandomization and stratified rerandomization. preprint (2024). arXiv:2406.02834

Wang, Y., Li, X.: Asymptotic theory of the best-choice rerandomization using the Mahalanobis distance. Journal of Econometrics **251** (2025) https://doi.org/10.1016/j.jeconom.2025.106049

Yang, Z., Qu, T., Li, X.: Rejective sampling, rerandomization, and regression adjustment in survey experiments. Journal of the American Statistical Association **118**(542), 1207–1221 (2021) https://doi.org/10.1080/01621459.2021.1984926

Zhao, A., Ding, P.: No star is good news: A unified look at rerandomization based on p-values from covariate balance tests. Journal of Econometrics **241** (2024) https://doi.org/10.1016/j.jeconom.2024.105724

Zhou, Q., Ernst, P.A., Morgan, K.L., Rubin, D.B., Zhang, A.: Sequential rerandomization. Biometrika **105**, 745–752 (2018) https://doi.org/10.1093/biomet/asy031

Zhu, K., Liu, H.: Pair-Switching Rerandomization. Biometrics **79**, 2127–2142 (2023) https://doi.org/10.1111/biom.13712