WALKING LOGISTICALLY, BUT LAZILY, FOR ONLINE DISCREPANCY MINIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

We study online discrepancy minimization in the oblivious setting, where vectors $v_1, \dots, v_T \in \mathbb{R}^d$ satisfying $\|v_t\|_2 \le 1$ are fixed in advance and revealed one at a time. The goal is to assign signs in $\{-1, +1\}$ upon arrival while keeping all prefix sums small in ℓ_∞ norm. Liu et al. (2022) showed that a parity obstruction prevents a one-dimensional Markov chain with steps in $\{-1, +1\}$ from preserving a Gaussian distribution exactly. By allowing coefficients in $\{-1, 0, +1\}$, they constructed a Gaussian fixed-point walk with at least 96% nonzero coefficients in expectation and an $O(\sqrt{\log T})$ prefix discrepancy bound with high probability, matching the optimal order for fully signed online discrepancy. More recently, Aden-Ali (2026) introduced the Gaussian triplet walk, which couples three Gaussian fixed-point walks to assign signs in $\{-1, +1\}$, achieving the optimal bound in linear time. We propose two complementary random walks. The first is a Metropolis-adjusted logistic walk, parametrized by β, that achieves the optimal discrepancy order while assigning coefficients in $\{-1, 0, +1\}$. The parameter β controls both the subgaussian parameter of the partial sums and the uniform upper bound on the expected fraction of zero coefficients. The second is a fully signed, unadjusted logistic walk that lacks a proved optimality guarantee but is empirically competitive. Across seven real datasets, the unadjusted walk is 2.28–$2.77\times$ faster than the Gaussian triplet walk and has 3.5%–11.6% lower mean maximum prefix discrepancy on six datasets, averaging 7.9% over those six. At $\beta = 2$, the adjusted walk is 1.89–$2.22\times$ faster than the Gaussian triplet walk and also has lower mean maximum prefix discrepancy on six datasets, while incurring a higher, though still less than 1.8%, observed fraction of zero coefficients than the Gaussian fixed-point walk.

1 INTRODUCTION

Discrepancy minimization studies how to assign signs to vectors so that their signed sum remains small. This combinatorial optimization technique allows for simplifying large objects to ones with lower complexity according to some metric. In the online setting, vectors arrive one at a time, and each sign must be chosen irrevocably upon arrival. We consider an oblivious adversary, which fixes a sequence $\boldsymbol{v}_1, \dots, \boldsymbol{v}_T \in \mathbb{R}^d$ with $\|\boldsymbol{v}_t\|_2 \le 1$ independently of the algorithm's randomness. The goal is to keep the infinity norm of every partial sum small.

Spencer (1977) was among the first to study a closely related max-norm vector-balancing problem. Banaszczyk subsequently developed powerful Gaussian-measure and signed-series bounds (Banaszczyk, 1998; 2012), and Bansal et al. (2019) gave an algorithm matching Banaszczyk's Komlós bound. These results provide the offline benchmarks against which online algorithms are compared.

A complementary line of work considers stochastic or structured input sequences. Bansal & Spencer (2020) treated i.i.d. uniform Rademacher vectors. Bansal et al. (2020) then allowed dependencies among coordinates and obtained applications to online interval and Tusnády discrepancy. Subsequently, Bansal et al. (2021) developed a broader framework for independent draws from arbitrary distributions and general target norms, matching offline bounds up to polylogarithmic factors. Recent works give matching bounds in a sparse i.i.d. binary model (Altschuler & Tikhomirov, 2025)

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

and online Beck–Fiala guarantees in the near-logarithmic sparsity regime (Altschuler & Tikhomirov, 2026). We refer the reader to Bansal & Nikolov (2026) for a recent overview.

Online discrepancy minimization has found a growing range of applications. With advances in AI systems and large-scale sequential data processing, it has become relevant to KV-cache compression and streaming-attention approximation (Kochetkova et al., 2026). Other applications include online experimental design (Arbour et al., 2022), discrepancy with recourse for vectors and graphs (Gupta et al., 2022), and dynamic carpooling with recourse (Efron et al., 2025).

In the oblivious online setting, Alweiss et al. (2021) were the first to introduce a linear-time algorithm, called the self-balancing walk, based on a linear sign-selection rule. Their convex-domination argument yields the suboptimal $O(\log(T/\delta))$ prefix discrepancy bound. An optimal algorithm was later presented by Kulkarni et al. (2024), albeit with exponential running time. They also prove that an oblivious adversary can force $\Omega(\sqrt{\log T})$ prefix discrepancy with high probability for every online algorithm (Kulkarni et al., 2024, Theorem 1.4), establishing the matching lower bound in its dependence on T. The same optimal order has since been extended to online multicolor discrepancy against an oblivious adversary (Halpern et al., 2025).

A related Gaussian relaxation replaces binary signs by correlated Gaussian variables; Chewi et al. (2022) gave a fast online algorithm in this model. A different question is how few zero updates are needed to preserve an exact Gaussian distribution. Liu et al. (2022) addressed this question with a Gaussian fixed-point walk that assigns coefficients in $\{-1, 0, +1\}$. At their standard Gaussian scale, at least 96% of its coefficients are nonzero with high probability, while its prefix discrepancy is $O(\sqrt{\log T})$ with high probability. The zero updates resolve the parity obstruction for a Gaussian-preserving walk restricted to unit signed steps. Their transition probabilities are defined by convergent series, and increasing the Gaussian scale reduces the probability of a zero update at the cost of a larger discrepancy bound. Our walk approaches the same question from a different direction: it uses a logistic proposal derived from a two-Gaussian mixture and a Metropolis correction to preserve a common Gaussian target. More recently, Aden-Ali (2026) coupled three Gaussian walks to obtain a walk with coefficients in $\{-1, +1\}$, $O(dT)$ running time, and the optimal $O(\sqrt{\log(T/\delta)})$ prefix discrepancy bound.

Relatedly, Smirnov & Vershynin (2026) study an online walk that applies a Metropolis correction to independently signed proposals and bounds the expected number of rejections via Fisher information. Our walk instead uses a state-dependent logistic proposal derived from a Gaussian mixture; the distinction lies in the proposal rather than the correction itself. For the Gaussian target considered here with unit updates and $\beta = 2$, the independently signed proposal rejects 38.3% of steps in stationarity, whereas our proved upper bound is 2.73%, a factor of approximately 14 in this comparison.

Our construction starts from a symmetric two-Gaussian mixture whose components are exchanged by opposite signed updates. The posterior probabilities of the component labels give the logistic sign-selection rule, which exactly preserves the mixture for a fixed update vector. Near the origin, this rule recovers the linear bias of Alweiss et al. (2021) to first order. We choose the component covariance so that the mixture matches the first two moments of a common Gaussian target, and then use a Metropolis correction to preserve that target as the input vectors change. Consequently, each transition depends only on $\langle \mathbf{x}_{t-1}, \boldsymbol{v}_t \rangle$ and $\|\boldsymbol{v}_t\|_2^2$, requires one auxiliary state, and avoids the alternating series of Liu et al. (2022). An accepted proposal produces a signed update, while a rejection produces coefficient zero.

We prove that every prefix sum is β-subgaussian and that the walk matches the optimal fully signed discrepancy order. The constant parameter $\beta > 1$ controls the tradeoff between discrepancy and the frequency of rejected updates. Our lazy-step result bounds the expected fraction of rejected updates. Moreover, Proposition 3.2 shows that this limitation is intrinsic: unlike the Gaussian fixed-point walk of Liu et al. (2022), our walk cannot satisfy a uniform exponential-in-T concentration bound for the fraction of zero steps. We complement the theory with experiments comparing the adjusted and unadjusted logistic walks with the self-balancing, Gaussian fixed-point, and Gaussian triplet walks.

We state the algorithm and its discrepancy guarantee below. The analysis is given in Section 2.

Algorithm 1 METROPOLIS-ADJUSTEDLOGISTICWALK($\boldsymbol{v}_1, \dots, \boldsymbol{v}_T, \beta$)

$\mathbf{x}_0 \sim \mathcal{N}(0, \beta^2 \boldsymbol{I}_d/4)$
$\mathbf{w}_0 \leftarrow 0$
for $t = 1, \dots, T$ **do**
  Define $p_t^+(\boldsymbol{x}) \leftarrow \mathrm{sigmoid}\left(-\frac{4\langle \boldsymbol{x}, \boldsymbol{v}_t\rangle}{\beta^2 - \|\boldsymbol{v}_t\|_2^2}\right), \quad p_t^-(\boldsymbol{x}) \leftarrow \mathrm{sigmoid}\left(\frac{4\langle \boldsymbol{x}, \boldsymbol{v}_t\rangle}{\beta^2 - \|\boldsymbol{v}_t\|_2^2}\right)$ ▷ Logistic rule
  Set $\varepsilon_t \leftarrow +1$ with prob. $p_t^+(\mathbf{x}_{t-1})$, and $\varepsilon_t \leftarrow -1$ with prob. $p_t^-(\mathbf{x}_{t-1})$
  Set $a_t \leftarrow \min\left\{1\,,\ \frac{\phi_\beta(\mathbf{x}_{t-1} + \varepsilon_t \boldsymbol{v}_t)\, p_t^{-\varepsilon_t}(\mathbf{x}_{t-1} + \varepsilon_t \boldsymbol{v}_t)}{\phi_\beta(\mathbf{x}_{t-1})\, p_t^{\varepsilon_t}(\mathbf{x}_{t-1})}\right\}$ ▷ Metropolis correction
  Set $\delta_t \leftarrow \varepsilon_t$ with probability a_t, and $\delta_t \leftarrow 0$ with probability $1 - a_t$
  $\mathbf{x}_t \leftarrow \mathbf{x}_{t-1} + \delta_t \boldsymbol{v}_t$
  $\mathbf{w}_t \leftarrow \mathbf{x}_t - \mathbf{x}_0$
return $(\delta_1, \dots, \delta_T)$

Theorem 1.1. *Let $\boldsymbol{v}_1, \dots, \boldsymbol{v}_T \in \mathbb{R}^d$ be an oblivious sequence satisfying $\|\boldsymbol{v}_t\|_2 \le 1$, and fix a constant $\beta > 1$. For every $\delta \in (0,1)$, algorithm* METROPOLIS-ADJUSTEDLOGISTICWALK *assigns $\delta_t \in \{-1, 0, 1\}$ and returns $\|\mathbf{w}_t\|_\infty = O\Big(\sqrt{\log(T/\delta)}\Big)$ for every $t \in [T]$, with probability at least $1 - \delta$.*

2 ANALYSIS

2.1 PRELIMINARIES

We first introduce the notation used throughout the paper.

For vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^d$, define the inner product $\langle \boldsymbol{u}, \boldsymbol{v}\rangle = \sum_{j=1}^d \boldsymbol{u}_j \boldsymbol{v}_j$. For $p \in [1, \infty)$, define $\|\boldsymbol{v}\|_p = (\sum_{j=1}^d |\boldsymbol{v}_j|^p)^{1/p}$ and $\|\boldsymbol{v}\|_\infty = \max_{j \in [d]} |\boldsymbol{v}_j|$.

For a random vector $\mathbf{x} \in \mathbb{R}^d$, we write $\mathbb{E}[\mathbf{x}]$ for its expectation and $\mathrm{Cov}(\mathbf{x}) = \mathbb{E}[(\mathbf{x} - \mathbb{E}[\mathbf{x}])(\mathbf{x} - \mathbb{E}[\mathbf{x}])^\top]$ for its covariance matrix. We write $\mathbf{x} \sim \mathcal{D}$ to mean that $\mathbf{x}$ is distributed according to $\mathcal{D}$; in particular, $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ means that $\mathbf{x}$ is Gaussian with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. For a symmetric matrix $\boldsymbol{A}$, we write $\boldsymbol{A} \succeq 0$ if $\boldsymbol{A}$ is positive semidefinite and $\boldsymbol{A} \succ 0$ if $\boldsymbol{A}$ is positive definite. Throughout, $\mathbf{g} \sim \mathcal{N}(0, \boldsymbol{I}_d)$ and $\mathrm{b} \sim \mathrm{Rad}(1/2)$ satisfy $\Pr[\mathrm{b} = +1] = \Pr[\mathrm{b} = -1] = 1/2$; whenever they appear together, $\mathbf{g}$ and b are independent. For $\boldsymbol{\Sigma} \succ 0$, denote the density of $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ by $\phi_{\boldsymbol{\mu}, \boldsymbol{\Sigma}}(\boldsymbol{x}) = \exp(-\langle \boldsymbol{x} - \boldsymbol{\mu}, \boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\mu})\rangle/2)/((2\pi)^{d/2} \det(\boldsymbol{\Sigma})^{1/2})$.

A random vector $\mathbf{x} \in \mathbb{R}^d$ is K-subgaussian if $\mathbb{E}[\exp(\langle \boldsymbol{u}, \mathbf{x} - \mathbb{E}[\mathbf{x}]\rangle)] \le \exp(K^2 \|\boldsymbol{u}\|_2^2/2)$ for every $\boldsymbol{u} \in \mathbb{R}^d$. Finally, we define the standard logistic, or sigmoid, function by $\mathrm{sigmoid}(x) = 1/(1+e^{-x})$. Its Taylor expansion around the origin is $\mathrm{sigmoid}(x) = 1/2 + x/4 - x^3/48 + O(x^5)$.

2.2 FROM GAUSSIAN MIXTURES TO LOGISTIC SIGN SELECTION

We first seek a sign-selection rule for updates in $\{-\boldsymbol{s}, +\boldsymbol{s}\}$, where $\boldsymbol{s} \in \mathbb{R}^d$ is fixed, nonzero, and $\|\boldsymbol{s}\|_2 \le 1$, together with an invariant distribution having a constant subgaussian parameter. A natural candidate is the equal mixture of two Gaussians, understood as a convex combination of probability measures:

$$\mathcal{M}_{\boldsymbol{s}, \boldsymbol{I}_d} := \frac{1}{2}\mathcal{N}\Big(-\frac{\boldsymbol{s}}{2}, \boldsymbol{I}_d\Big) + \frac{1}{2}\mathcal{N}\Big(\frac{\boldsymbol{s}}{2}, \boldsymbol{I}_d\Big), \tag{1}$$

Equivalently, if $\mathbf{x} \sim \mathcal{M}_{\boldsymbol{s}, \boldsymbol{I}_d}$, for independent $\mathbf{g} \sim \mathcal{N}(0, \boldsymbol{I}_d)$ and $\mathrm{b} \sim \mathrm{Rad}(1/2)$,

$$\mathbf{x} := \mathbf{g} + \frac{\mathrm{b}\boldsymbol{s}}{2} \sim \mathcal{M}_{\boldsymbol{s}, \boldsymbol{I}_d}. \tag{2}$$

As we prove in the next subsection, this mixture is constant-subgaussian, and its component posterior gives the probabilities

$$p^+_{\boldsymbol{s}, \boldsymbol{I}_d}(\boldsymbol{x}) = \mathrm{sigmoid}(-\langle \boldsymbol{x}, \boldsymbol{s}\rangle), \qquad p^-_{\boldsymbol{s}, \boldsymbol{I}_d}(\boldsymbol{x}) = \mathrm{sigmoid}(\langle \boldsymbol{x}, \boldsymbol{s}\rangle).$$

That is, the rule chooses $+\boldsymbol{s}$ when $\boldsymbol{x}$ is more likely to have come from the component centered at $-\boldsymbol{s}/2$, and $-\boldsymbol{s}$ when the component centered at $\boldsymbol{s}/2$ is more likely. This posterior interpretation is the first justification for using the logistic sign-selection rule.

The logistic rule also preserves the properties used by Alweiss et al. (2021) to motivate their linear rule. When the current state $\boldsymbol{w}$ and update vector $\boldsymbol{v}$ are orthogonal, it chooses each sign with probability $1/2$. As $\langle \boldsymbol{w}, \boldsymbol{v} \rangle$ increases, the probability of choosing $-\boldsymbol{v}$ increases, producing a stronger bias against the current alignment. A particularly interesting observation about the logistic rule is its natural connection to their linear sign-selection rule. For any constant $c > 0$, the Taylor expansion

$$\operatorname{sigmoid}\left(\mp \frac{2\langle \boldsymbol{w}, \boldsymbol{v} \rangle}{c}\right) = \frac{1}{2} \mp \frac{\langle \boldsymbol{w}, \boldsymbol{v} \rangle}{2c} + O\left(\left|\frac{\langle \boldsymbol{w}, \boldsymbol{v} \rangle}{c}\right|^3\right)$$

shows that the first-order Taylor polynomial of the logistic probabilities is exactly their linear sign-selection rule. This gives a second justification for the logistic rule: it is a nonlinear extension of their balancing rule near the origin. Their proof, however, is based on a convex-domination argument that relies on the linearity of the conditional mean update and therefore does not apply directly to our nonlinear rule. For completeness we provide their algorithm in Appendix C.

2.3 A FIXED-VECTOR LOGISTIC WALK

We now prove that the logistic update preserves this mixture for an arbitrary component covariance $\boldsymbol{\Sigma}$.

Lemma 2.1. *Fix $\boldsymbol{s} \in \mathbb{R}^d$ and $\boldsymbol{\Sigma} \succ 0$, and let $\mathbf{x} \sim \mathcal{M}_{\boldsymbol{s},\boldsymbol{\Sigma}}$. Conditionally on $\mathbf{x} = \boldsymbol{x}$, choose $\varepsilon \in \{-1, +1\}$ according to*

$$\begin{aligned} \Pr[\varepsilon = +1 \mid \mathbf{x} = \boldsymbol{x}] &=: p^{+}_{\boldsymbol{s},\boldsymbol{\Sigma}}(\boldsymbol{x}) = \operatorname{sigmoid}(-\langle \boldsymbol{x}, \boldsymbol{\Sigma}^{-1}\boldsymbol{s} \rangle), \\ \Pr[\varepsilon = -1 \mid \mathbf{x} = \boldsymbol{x}] &=: p^{-}_{\boldsymbol{s},\boldsymbol{\Sigma}}(\boldsymbol{x}) = \operatorname{sigmoid}(\langle \boldsymbol{x}, \boldsymbol{\Sigma}^{-1}\boldsymbol{s} \rangle). \end{aligned} \tag{3}$$

Then $\mathbf{x} + \varepsilon \boldsymbol{s} \sim \mathcal{M}_{\boldsymbol{s},\boldsymbol{\Sigma}}$.

Proof. Let ϕ_- and ϕ_+ denote the densities of $\mathcal{N}(-\boldsymbol{s}/2, \boldsymbol{\Sigma})$ and $\mathcal{N}(\boldsymbol{s}/2, \boldsymbol{\Sigma})$, respectively. Since $\phi_+(\boldsymbol{x})/\phi_-(\boldsymbol{x}) = \exp(\langle \boldsymbol{x}, \boldsymbol{\Sigma}^{-1}\boldsymbol{s} \rangle)$, we have $p^{+}_{\boldsymbol{s},\boldsymbol{\Sigma}}(\boldsymbol{x}) = \phi_-(\boldsymbol{x})/(\phi_-(\boldsymbol{x}) + \phi_+(\boldsymbol{x}))$ and $p^{-}_{\boldsymbol{s},\boldsymbol{\Sigma}}(\boldsymbol{x}) = \phi_+(\boldsymbol{x})/(\phi_-(\boldsymbol{x}) + \phi_+(\boldsymbol{x}))$. Let f denote the density of $\mathbf{x}$ and g the density of $\mathbf{y} := \mathbf{x} + \varepsilon \boldsymbol{s}$. Then

$$\begin{aligned} g(\boldsymbol{y}) &= f(\boldsymbol{y} - \boldsymbol{s}) p^{+}_{\boldsymbol{s},\boldsymbol{\Sigma}}(\boldsymbol{y} - \boldsymbol{s}) + f(\boldsymbol{y} + \boldsymbol{s}) p^{-}_{\boldsymbol{s},\boldsymbol{\Sigma}}(\boldsymbol{y} + \boldsymbol{s}) \\ &= \frac{1}{2}\phi_-(\boldsymbol{y} - \boldsymbol{s}) + \frac{1}{2}\phi_+(\boldsymbol{y} + \boldsymbol{s}) \\ &= \frac{1}{2}\phi_+(\boldsymbol{y}) + \frac{1}{2}\phi_-(\boldsymbol{y}) = f(\boldsymbol{y}). \end{aligned}$$

Hence $\mathbf{y} \sim \mathcal{M}_{\boldsymbol{s},\boldsymbol{\Sigma}}$. □

This general-covariance form will allow us to choose the component covariance to match a common Gaussian target in Subsection 2.4.

Lemma 2.2. *If $\|\boldsymbol{s}\|_2 \le 1$ and $\mathbf{x} \sim \mathcal{M}_{\boldsymbol{s},\boldsymbol{I}_d}$, then $\mathbf{x}$ is $\sqrt{5}/2$-subgaussian.*

Proof. For every $\boldsymbol{u} \in \mathbb{R}^d$,

$$\begin{aligned} \mathbb{E}\left[\exp\left(\langle \boldsymbol{u}, \mathbf{x} \rangle\right)\right] &= \frac{1}{2}\exp\left(\frac{\|\boldsymbol{u}\|_2^2}{2} - \frac{\langle \boldsymbol{u}, \boldsymbol{s} \rangle}{2}\right) + \frac{1}{2}\exp\left(\frac{\|\boldsymbol{u}\|_2^2}{2} + \frac{\langle \boldsymbol{u}, \boldsymbol{s} \rangle}{2}\right) \\ &= \exp\left(\frac{\|\boldsymbol{u}\|_2^2}{2}\right)\cosh\left(\frac{\langle \boldsymbol{u}, \boldsymbol{s} \rangle}{2}\right) \\ &\le \exp\left(\frac{\|\boldsymbol{u}\|_2^2}{2} + \frac{\langle \boldsymbol{u}, \boldsymbol{s} \rangle^2}{8}\right) \le \exp\left(\frac{5}{8}\|\boldsymbol{u}\|_2^2\right). \end{aligned}$$

The claim follows from $\cosh(x) \le \exp(x^2/2)$, Cauchy–Schwarz, and $\|\boldsymbol{s}\|_2 \le 1$. □

The remaining obstacle is that an online sequence uses different vectors $\boldsymbol{v}_i$, and hence different invariant mixtures $\mathcal{M}_{\boldsymbol{v}_i,\boldsymbol{I}_d}$. Invariance for one vector does not establish invariance under the update for another. Although we do not prove a discrepancy guarantee for the resulting unadjusted logistic walk, it maintains low discrepancy in every setting considered in Section 4. To obtain a theoretical guarantee, we use the fixed-vector transition proved here as a proposal and correct it so that every step preserves the same Gaussian target.

2.4 A METROPOLIS-ADJUSTED LOGISTIC WALK

Fix a constant $\beta > 1$. A Metropolis correction preserves the common Gaussian target $\mathcal{N}(0, \beta^2\boldsymbol{I}_d/4)$ as the incoming vector changes. Since each rejection produces a zero step, we choose the auxiliary mixture to match the target's first two moments, with the aim of keeping rejections rare.

We now use the representation in equation 2 with a general component covariance. For an arbitrary update vector $\boldsymbol{v} \in \mathbb{R}^d$ with $\|\boldsymbol{v}\|_2 \le 1$, let $\mathbf{z} := \boldsymbol{\Sigma}^{1/2}\mathbf{g} + \frac{\mathrm{b}\boldsymbol{v}}{2} \sim \mathcal{M}_{\boldsymbol{v},\boldsymbol{\Sigma}}$. Since $\mathbf{z}$ is centered and $\mathbf{g}$ and b are independent, $\mathrm{Cov}(\mathbf{z}) = \boldsymbol{\Sigma} + \frac{\boldsymbol{v}\boldsymbol{v}^\top}{4}$. Matching this covariance to that of the Gaussian target therefore gives

$$\boldsymbol{\Sigma}_{\boldsymbol{v}} := \frac{\beta^2\boldsymbol{I}_d - \boldsymbol{v}\boldsymbol{v}^\top}{4} \succ 0$$

The subtraction compensates for the covariance contributed by the random shift $\mathrm{b}\boldsymbol{v}/2$. Although the resulting mixture $\mathcal{M}_{\boldsymbol{v},\boldsymbol{\Sigma}_{\boldsymbol{v}}}$ matches the target's mean and covariance, it is not Gaussian when $\boldsymbol{v} \neq 0$. Moreover, $\boldsymbol{v}$ is an eigenvector of $\boldsymbol{\Sigma}_{\boldsymbol{v}}$ with eigenvalue $(\beta^2 - \|\boldsymbol{v}\|_2^2)/4$, and hence

$$\boldsymbol{\Sigma}_{\boldsymbol{v}}^{-1}\boldsymbol{v} = \frac{4\boldsymbol{v}}{\beta^2 - \|\boldsymbol{v}\|_2^2}$$

Substitution into equation 3 gives exactly the logistic coefficient used in Algorithm 1. At step i, we specialize to $\boldsymbol{v} = \boldsymbol{v}_i$ and apply the Metropolis acceptance rule below. The reverse of a proposal with sign ε_i uses sign $-\varepsilon_i$, which explains the reverse-proposal factor $p_i^{-\varepsilon_i}(\boldsymbol{x} + \varepsilon_i\boldsymbol{v}_i)$ in the acceptance ratio. The following lemma proves Gaussian invariance.

Lemma 2.3. *Fix $\beta > 1$ and an oblivious sequence $\boldsymbol{v}_1, \dots, \boldsymbol{v}_T \in \mathbb{R}^d$ with $\|\boldsymbol{v}_i\|_2 \le 1$. Let ϕ_β denote the density of $\mathcal{N}(0, \beta^2\boldsymbol{I}_d/4)$. For every $i \in [T]$, set $\boldsymbol{\Sigma}_i := (\beta^2\boldsymbol{I}_d - \boldsymbol{v}_i\boldsymbol{v}_i^\top)/4$, and write $p_i^{\pm} := p^{\pm}_{\boldsymbol{v}_i,\boldsymbol{\Sigma}_i}$ for the general-covariance proposal probabilities in equation 3. Define*

$$a_i^+(\boldsymbol{x}) := \min\left\{1, \frac{\phi_\beta(\boldsymbol{x} + \boldsymbol{v}_i)p_i^-(\boldsymbol{x} + \boldsymbol{v}_i)}{\phi_\beta(\boldsymbol{x})p_i^+(\boldsymbol{x})}\right\} \tag{4}$$

$$a_i^-(\boldsymbol{x}) := \min\left\{1, \frac{\phi_\beta(\boldsymbol{x} - \boldsymbol{v}_i)p_i^+(\boldsymbol{x} - \boldsymbol{v}_i)}{\phi_\beta(\boldsymbol{x})p_i^-(\boldsymbol{x})}\right\} \tag{5}$$

Starting from $\mathbf{x}_0 \sim \mathcal{N}(0, \beta^2\boldsymbol{I}_d/4)$, conditionally on $\mathbf{x}_{i-1} = \boldsymbol{x}$, choose a proposal $\varepsilon_i \in \{-1, +1\}$ with $p_i^{\pm}(\boldsymbol{x})$. Then,

$$\Pr[\delta_i = +1 \mid \mathbf{x}_{i-1} = \boldsymbol{x}] = p_i^+(\boldsymbol{x})a_i^+(\boldsymbol{x}), \tag{6}$$

$$\Pr[\delta_i = -1 \mid \mathbf{x}_{i-1} = \boldsymbol{x}] = p_i^-(\boldsymbol{x})a_i^-(\boldsymbol{x}), \tag{7}$$

$$\Pr[\delta_i = 0 \mid \mathbf{x}_{i-1} = \boldsymbol{x}] = 1 - p_i^+(\boldsymbol{x})a_i^+(\boldsymbol{x}) - p_i^-(\boldsymbol{x})a_i^-(\boldsymbol{x}), \tag{8}$$

and set $\mathbf{x}_i := \mathbf{x}_{i-1} + \delta_i\boldsymbol{v}_i$. Then $\mathbf{x}_i \sim \mathcal{N}(0, \beta^2\boldsymbol{I}_d/4)$ for every $i \in [T]$.

Proof. For every nonzero $\boldsymbol{u} \in \mathbb{R}^d$, $\boldsymbol{u}^\top\boldsymbol{\Sigma}_i\boldsymbol{u} \ge (\beta^2 - \|\boldsymbol{v}_i\|_2^2)\|\boldsymbol{u}\|_2^2/4 > 0$. Hence $\boldsymbol{\Sigma}_i \succ 0$, and $\boldsymbol{\Sigma}_i^{-1}\boldsymbol{v}_i = 4\boldsymbol{v}_i/(\beta^2 - \|\boldsymbol{v}_i\|_2^2)$. The general-covariance extension of Lemma 2.1 shows that p_i^+ and p_i^- form the invariant sigmoid proposal for $\mathcal{M}_{\boldsymbol{v}_i,\boldsymbol{\Sigma}_i}$.

Fix $i \in [T]$. For $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{v}_i$, the definition of a_i^+ gives

$$\begin{aligned}\phi_\beta(\boldsymbol{x})p_i^+(\boldsymbol{x})a_i^+(\boldsymbol{x}) &= \min\left\{\phi_\beta(\boldsymbol{x})p_i^+(\boldsymbol{x}), \phi_\beta(\boldsymbol{y})p_i^-(\boldsymbol{y})\right\} \\ &= \phi_\beta(\boldsymbol{y})p_i^-(\boldsymbol{y})a_i^-(\boldsymbol{y}).\end{aligned}$$

The analogous identity for $\boldsymbol{y} = \boldsymbol{x} - \boldsymbol{v}_i$ is

$$\phi_\beta(\boldsymbol{x})p_i^-(\boldsymbol{x})a_i^-(\boldsymbol{x}) = \phi_\beta(\boldsymbol{y})p_i^+(\boldsymbol{y})a_i^+(\boldsymbol{y}).$$

Thus the accepted moves satisfy detailed balance with respect to ϕ_β, while rejected proposals remain at their current state. Hence every transition preserves ϕ_β, and induction proves the claim. □

2.5 PREFIX SUMS

The Gaussian invariance concerns the marginal law of $\mathbf{x}_t$. It does not imply that $\mathbf{w}_t = \mathbf{x}_t - \mathbf{x}_0$ is Gaussian, since the two states are dependent. The following lemma instead derives the required subgaussian bound directly.

Lemma 2.4. *Under the assumptions of Lemma 2.3, define* $\mathbf{w}_t := \sum_{s=1}^{t} \delta_s \boldsymbol{v}_s$. *Then* $\mathbb{E}[\mathbf{w}_t] = 0$, *and for every* $t \in [T]$ *and* $\boldsymbol{u} \in \mathbb{R}^d$,

$$\mathbb{E}\left[\exp\left(\langle \boldsymbol{u}, \mathbf{w}_t \rangle\right)\right] \le \exp\left(\frac{\beta^2 \|\boldsymbol{u}\|_2^2}{2}\right).$$

Consequently, every $\mathbf{w}_t$ *is* β*-subgaussian.*

Proof. By construction, $\mathbf{w}_t = \mathbf{x}_t - \mathbf{x}_0$. Lemma 2.3 gives $\mathbf{x}_t \sim \mathcal{N}(0, \beta^2 \boldsymbol{I}_d/4)$ for every $t \in [T]$. Since $\mathbf{x}_0$ has the same centered Gaussian distribution, we have $\mathbb{E}[\mathbf{w}_t] = \mathbb{E}[\mathbf{x}_t] - \mathbb{E}[\mathbf{x}_0] = 0$. Moreover, Cauchy–Schwarz gives

$$\begin{aligned}
\mathbb{E}\left[\exp\left(\langle \boldsymbol{u}, \mathbf{w}_t \rangle\right)\right] &= \mathbb{E}\left[\exp\left(\langle \boldsymbol{u}, \mathbf{x}_t - \mathbf{x}_0 \rangle\right)\right] \\
&\le \mathbb{E}\left[\exp\left(2\langle \boldsymbol{u}, \mathbf{x}_t \rangle\right)\right]^{1/2} \mathbb{E}\left[\exp\left(-2\langle \boldsymbol{u}, \mathbf{x}_0 \rangle\right)\right]^{1/2} \\
&= \exp\left(\frac{1}{2} \cdot \frac{1}{2} (2\boldsymbol{u})^\top \frac{\beta^2}{4} \boldsymbol{I}_d (2\boldsymbol{u})\right) \exp\left(\frac{1}{2} \cdot \frac{1}{2} (-2\boldsymbol{u})^\top \frac{\beta^2}{4} \boldsymbol{I}_d (-2\boldsymbol{u})\right) \\
&= \exp\left(\frac{\beta^2 \|\boldsymbol{u}\|_2^2}{4}\right) \exp\left(\frac{\beta^2 \|\boldsymbol{u}\|_2^2}{4}\right) = \exp\left(\frac{\beta^2 \|\boldsymbol{u}\|_2^2}{2}\right).
\end{aligned}$$

□

Now we are ready to prove the main theorem. The projection-and-trace argument below follows the proof of Theorem 2 of Aden-Ali (2026); we apply it to the subgaussian bound above.

Proof of Theorem 1.1. For $t \in [T]$, let $E_t := \operatorname{span}\{\boldsymbol{v}_1, \dots, \boldsymbol{v}_t\}$ and let P_t be the orthogonal projection onto E_t. Since $\mathbf{w}_t \in E_t$, we have $\langle \boldsymbol{u}, \mathbf{w}_t \rangle = \langle P_t \boldsymbol{u}, \mathbf{w}_t \rangle$ for every $\boldsymbol{u} \in \mathbb{R}^d$. Lemma 2.4, applied with $P_t \boldsymbol{u}$, therefore gives

$$\mathbb{E}\left[\exp\left(\langle \boldsymbol{u}, \mathbf{w}_t \rangle\right)\right] \le \exp\left(\frac{\beta^2 \|P_t \boldsymbol{u}\|_2^2}{2}\right). \tag{9}$$

Let $\boldsymbol{e}_k$ be the kth standard basis vector and define $q_{t,k} := \|P_t \boldsymbol{e}_k\|_2^2$. If $q_{t,k} = 0$, then $(\mathbf{w}_t)_k = 0$. Otherwise, the standard subgaussian tail bound applied to equation 9 yields $\Pr[|(\mathbf{w}_t)_k| > c] \le 2\exp(-c^2/(2\beta^2 q_{t,k}))$ for every $c > 0$. Set $x := c^2/(2\beta^2)$. For $x \ge 1$ and $q \in (0, 1]$, we have $e^{-x/q} \le q e^{-x}$ because $q^{-1} - 1 \ge -\log q$ and hence $x(q^{-1} - 1) \ge -\log q$. Consequently,

$$\Pr[\|\mathbf{w}_t\|_\infty > c] \le 2e^{-x} \sum_{k=1}^{d} q_{t,k} = 2e^{-x} \operatorname{tr}(P_t) \le 2te^{-x}$$

where $\operatorname{tr}(P_t) = \dim(E_t) \le t$. Now, choose $c := \beta\sqrt{2\log(\mathrm{e}\, T(T+1)/\delta)}$. Then $x = \log(\mathrm{e}\, T(T+1)/\delta) \ge 1$, and a union bound over $t \in [T]$ gives

$$\Pr\left[\max_{t \in [T]} \|\mathbf{w}_t\|_\infty > c\right] \le 2e^{-x} \sum_{t=1}^{T} t = T(T+1)e^{-x} = \frac{\delta}{\mathrm{e}} \le \delta$$

This proves the theorem. □

3 LAZY STEPS

In this section we address the lazy step probability induced by our random process.

3.1 How often is the correction used?

The distinction from the walk of Liu et al. (2022) is important when quantifying lazy steps. Their construction controls the zero-step probability uniformly over the current state, whereas the conditional rejection probability of our Metropolis correction can be arbitrarily close to one for some states. Consequently, we do not claim a worst-case-over-state bound. Instead, we average over the invariant Gaussian marginal of the auxiliary state, which is its distribution at every step of the run.

We bound the zero-step probability uniformly over $\|\boldsymbol{v}_t\|_2 \le 1$. This is an along-the-run marginal statement: at each time t, the probability averages over $\mathbf{x}_{t-1} \sim \mathcal{N}(0, \beta^2 \boldsymbol{I}_d/4)$ rather than holding uniformly for every realized state.

Lemma 3.1 (Zero-step upper bound)**.** *Fix $\beta > 1$ and $\boldsymbol{v}_t \in \mathbb{R}^d$ with $\|\boldsymbol{v}_t\|_2 \le 1$. Suppose $\mathbf{x}_{t-1} \sim \mathcal{N}(0, \beta^2 \boldsymbol{I}_d/4)$ and apply the Metropolis-adjusted logistic step with proposal probabilities $p_t^{\pm}$ and acceptance probabilities $a_t^{\pm}$. Then*

$$\Pr[\delta_t = 0] \le \min\left\{1, \left(\frac{e^{1/(\beta^2+1)} + e^{-1/(\beta^2-1)}}{2\sqrt{1-\beta^{-4}}} - 1\right)^{1/2}\right\}. \tag{10}$$

The proof is presented in Appendix A

The next proposition shows that the marginal guarantee above cannot, in general, be strengthened to exponential concentration in T. Let $N_0(T) := |\{t \in [T] : \delta_t = 0\}|$ denote the number of zero steps.

Proposition 3.2 (Failure of exponential retention concentration)**.** *For every fixed $\beta > 1$ and all sufficiently large T, there exists an oblivious sequence $\boldsymbol{v}_1, \dots, \boldsymbol{v}_T$ such that the* Metropolis-AdjustedLogisticWalk *satisfies*

$$\Pr[N_0(T) = T] \ge \exp\big(-O_\beta(\log^2 T)\big).$$

Consequently, for fixed $c > 0$ and $\varepsilon \in (0, 1)$, no uniform bound of the form

$$\Pr[N_0(T) \ge \varepsilon T] \le e^{-cT}$$

can hold over all oblivious input sequences.

The proof is presented in Appendix A.

Thus, the marginal zero-step bound does not imply exponential concentration of the total number of rejections. Proposition 3.2 shows that this limitation is inherent to the transition rule. Although Lemma 3.1 is weaker than the high-probability retention guarantee of Liu et al. (2022), the experiments in Section 4 show low observed zero-step frequencies for the Metropolis-adjusted logistic walk.

3.2 Parameter choice and numerical evaluation

Any real $\beta > 1$ is allowed, ensuring $\beta^2 - \|\boldsymbol{v}_t\|_2^2 > 0$, so the logistic proposal is well-defined and $\boldsymbol{\Sigma}_t$ is positive definite. Generally, the transition requires $\beta > \max_t \|\boldsymbol{v}_t\|_2$.

Evaluating the closed-form Cauchy–Schwarz bound gives zero-step upper bounds of 2.729194%, 0.510704%, and 0.160138% for $\beta = 2, 3, 4$, respectively. These are upper bounds, not predicted zero-step frequencies. The choice of β trades off the discrepancy guarantee $\beta\sqrt{2\log(\mathrm{e}\,T(T+1)/\delta)}$ against the bound on zero steps.

4 Experiments

We compare the adjusted logistic walk at $\beta \in \{2, 3, 4\}$, the fully signed unadjusted walk at $\beta = 2$, SelfBalance (Alweiss et al., 2021), the Gaussian fixed-point walk (Liu et al., 2022), and Gaussian triplet (Aden-Ali, 2026). We use seven real vector collections and five synthetic datasets, always sharing the same input order across methods and averaging over 20 seeds.

We report maximum prefix discrepancy, zero-step percentage, and runtime. Since the adjusted, Gaussian fixed-point, and Gaussian triplet walks have the same optimal asymptotic discrepancy

Table 1: Mean maximum prefix discrepancy over 20 seeds for all methods. Bold marks the lowest value in each row; complete mean ± standard deviation results are in Appendix B.

Dataset	Metropolis-adjusted logistic $\beta = 2$	$\beta = 3$	$\beta = 4$	Unadjusted logistic	Gaussian fixed-point	Gaussian triplet	SelfBalance
MNIST	**4.940**	6.237	7.332	4.981	5.008	5.163	16.173
NYTimes	**6.920**	9.868	12.954	6.977	6.994	7.449	62.017
SIFT1M	**6.850**	10.304	13.746	7.109	6.934	8.041	71.026
GIST	5.189	6.844	8.069	5.203	5.298	**5.026**	17.217
Llama	6.198	8.828	11.184	6.168	**6.030**	6.530	44.320
ArXiv	**6.994**	10.381	13.071	7.188	7.001	8.062	53.137
LAION	**7.075**	10.314	13.398	7.255	7.195	8.040	74.195

Table 2: Mean loop runtime in seconds over 20 seeds for all methods. Bold marks the lowest value in each row. Timing conventions and complete results are given in Appendix B.

Dataset	Metropolis-adjusted logistic $\beta = 2$	$\beta = 3$	$\beta = 4$	Unadjusted logistic	Gaussian fixed-point	Gaussian triplet	SelfBalance
MNIST	0.96	1.04	0.97	0.82	1.43	1.95	**0.70**
NYTimes	3.79	3.84	3.82	**3.04**	5.53	8.43	3.15
SIFT1M	14.00	13.85	13.79	11.61	18.56	29.14	**10.93**
GIST	18.62	19.07	18.96	16.31	34.47	37.15	**14.46**
Llama	3.58	3.52	3.51	2.91	5.01	7.46	**2.65**
ArXiv	24.45	24.37	23.12	20.13	30.70	46.20	**18.76**
LAION	17.59	17.76	16.94	14.24	22.75	35.18	**13.77**

order, the experiments primarily compare constants and computational tradeoffs. Tables 1 and 2 give the main real-data summary. Dataset construction, normalization, implementation details, standard deviations, paired comparisons, matched-retention experiments, and all synthetic results are deferred to Appendix B.

Main findings. At $\beta = 2$, the adjusted logistic walk has the lowest mean discrepancy on five of the seven real datasets and is generally close to the Gaussian fixed-point and unadjusted walks. It usually improves on Gaussian triplet, although the gaps are modest and not uniform. Increasing β sharply reduces the zero-step fraction at the expected cost of larger discrepancy.

Both logistic variants are faster than the Gaussian fixed-point and triplet implementations on every real dataset. The matched-retention experiments, however, favor the Gaussian fixed-point walk in discrepancy, so the empirical advantage of the logistic transition is computational rather than a uniformly better discrepancy–retention tradeoff. The synthetic experiments reinforce the same picture: the logistic walks are competitive with the Gaussian fixed-point walk, while the pointwise paired intervals in Table 9 favor both logistic walks over Gaussian triplet on every synthetic family and batch. Detailed comparisons and uncertainty estimates are given in Appendix B.

The fully signed unadjusted walk is particularly suggestive: it remains competitive with the proved optimal-order methods despite lacking a common Gaussian invariant for changing input vectors. This motivates the following conjecture.

Conjecture 4.1. *The (unadjusted) logistic walk, or a closely related fully signed logistic walk, achieves the optimal $O(\sqrt{\log T})$ prefix discrepancy bound with high probability.*

5 Conclusion

We derived a logistic sign-selection rule from the posterior probabilities of a two-Gaussian mixture and used a Metropolis correction to obtain an explicit single-state, mixture-based walk with a common Gaussian invariant distribution. Each transition requires only an inner product and a vector update, giving an $O(d)$ implementation with optimal-order prefix discrepancy. The correction permits zero steps, and our retention guarantee is in expectation rather than with high probability, although we show that this limitation is intrinsic to the transition rule: no uniform exponential-in-T concentration bound on the fraction of zero steps can hold over all oblivious input sequences.

In the experiments, the adjusted walk has competitive discrepancy and lower observed runtime than the Gaussian fixed-point and triplet baselines, although the matched-retention comparison shows no

discrepancy–sparsity advantage over the Gaussian fixed-point walk. The fully signed unadjusted walk is simpler and faster still, while remaining empirically competitive with Gaussian triplet. This leaves a concrete question: does the unadjusted logistic walk, or a closely related fully signed logistic transition, achieve optimal prefix discrepancy against oblivious input sequences?

Reproducibility statement

The algorithms and all assumptions used in the analysis are stated in the main text, with the remaining proofs in Appendix A. Appendix B documents the datasets, normalization, implementations, random seeds, hardware, statistical procedures, and complete experimental results. Code, configuration files, and run-level outputs are provided with the anonymous supplementary material in Appendix.

AI use statement

In this work, we used OpenAI Codex to aid and polish writing and LaTeX formatting; retrieve and discover related work; support research ideation and execution; draft portions of the paper, particularly Section 4 and Appendix B; assist with experimental design and implementation; generate synthetic datasets; and assist in proving mathematical claims. We also used it for statistical reporting and for interpreting and presenting experimental results. The algorithm emerged through several rounds of interaction with the LLM. The authors originated the idea of replacing the linear rule of Alweiss et al. (2021) with the logistic rule formalized in Lemma 2.1. The mixture-based viewpoint for the auxiliary state and the analysis through Gaussian-invariant walks were inspired by Aden-Ali (2026) and adapted to the logistic construction in Lemmas 2.1 and 2.3. The authors proposed allowing lazy steps, while the LLM suggested implementing them through a Metropolis correction; the resulting rejection bound is analyzed in Lemma 3.1. The authors reviewed every AI-assisted change: generated code was tested using validity checks and reproducible run records, mathematical derivations and calculations were verified, citations were checked against their sources, and the compiled manuscript was inspected before inclusion. We take responsibility for the final content of this work, including text, claims, code, data, and other artifacts produced with the aid of generative AI.

References

Ishaq Aden-Ali. Optimal online discrepancy minimization in linear time. *arXiv preprint arXiv:2607.04388*, 2026.

Dylan J Altschuler and Konstantin Tikhomirov. A threshold for online balancing of sparse iid vectors. *arXiv preprint arXiv:2509.02432*, 2025.

Dylan J Altschuler and Konstantin Tikhomirov. Online beck–fiala down to logarithmic sparsity. *arXiv preprint arXiv:2607.14238*, 2026.

Ryan Alweiss, Yang P. Liu, and Mehtaab Sawhney. Discrepancy minimization via a self-balancing walk. In Samir Khuller and Virginia Vassilevska Williams (eds.), *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 14–20. ACM, 2021. doi: 10.1145/3406325.3450994.

David Arbour, Drew Dimmery, Tung Mai, and Anup Rao. Online balanced experimental design. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 844–864. PMLR, 17–23 Jul 2022. URL `https://proceedings.mlr.press/v162/arbour22a.html`.

Wojciech Banaszczyk. Balancing vectors and gaussian measures of n-dimensional convex bodies. *Random Structures & Algorithms*, 12(4):351–360, 1998.

Wojciech Banaszczyk. On series of signed vectors and their rearrangements. *Random Structures & Algorithms*, 40(3):301–316, 2012.

Nikhil Bansal and Aleksandar Nikolov. Discrepancy theory: An algorithmic and geometric perspective. *arXiv preprint arXiv:2608.00140*, 2026.

Nikhil Bansal and Joel H Spencer. On-line balancing of random inputs. *Random Structures & Algorithms*, 57(4):879–891, 2020.

Nikhil Bansal, Daniel Dadush, and Shashwat Garg. An algorithm for komlós conjecture matching banaszczyk's bound. *SIAM Journal on Computing*, 48(2):534–553, 2019.

Nikhil Bansal, Haotian Jiang, Sahil Singla, and Makrand Sinha. Online vector balancing and geometric discrepancy. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 1139–1152, 2020.

Nikhil Bansal, Haotian Jiang, Raghu Meka, Sahil Singla, and Makrand Sinha. Online discrepancy minimization for stochastic arrivals. In *Proceedings of the Thirty-Second Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '21, pp. 2842–2861, USA, 2021. Society for Industrial and Applied Mathematics. ISBN 9781611976465.

Sinho Chewi, Patrik Gerber, Philippe Rigollet, and Paxton Turner. Gaussian discrepancy: A probabilistic relaxation of vector balancing. *Discrete Applied Mathematics*, 322:123–141, 2022.

Yuval Efron, Shyamal Patel, and Cliff Stein. A simple algorithm for dynamic carpooling with recourse. In *2025 Symposium on Simplicity in Algorithms (SOSA)*, pp. 196–201. SIAM, 2025.

Anupam Gupta, Vijaykrishna Gurunathan, Ravishankar Krishnaswamy, Amit Kumar, and Sahil Singla. Online discrepancy with recourse for vectors and graphs. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 1356–1383. SIAM, 2022.

Daniel Halpern, Alexandros Psomas, Paritosh Verma, and Daniel Xie. Online envy minimization and multicolor discrepancy: Equivalences and separations. *arXiv preprint arXiv:2502.14624*, 2025.

Ekaterina Kochetkova, Kshiteej Jitesh Sheth, Insu Han, Amir Zandieh, and Michael Kapralov. Streaming attention approximation via discrepancy theory. *Advances in Neural Information Processing Systems*, 38:20935–20972, 2026.

Janardhan Kulkarni, Victor Reis, and Thomas Rothvoss. Optimal online discrepancy minimization. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pp. 1832–1840, 2024.

Yang P. Liu, Ashwin Sah, and Mehtaab Sawhney. A gaussian fixed point random walk. In Mark Braverman (ed.), *13th Innovations in Theoretical Computer Science Conference, ITCS 2022*, volume 215 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 101:1–101:10. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi: 10.4230/LIPIcs.ITCS.2022.101.

Gleb Smirnov and Roman Vershynin. Discrepancy and fisher information. *arXiv preprint arXiv:2605.13107*, 2026.

Joel Spencer. Balancing games. *Journal of Combinatorial Theory, Series B*, 23(1):68–74, 1977.

A PROOFS OF SECTION 3

Proof of Lemma 3.1. If $\boldsymbol{v}_t = 0$, the rejection probability is zero. Otherwise, with $\boldsymbol{\Sigma}_t = (\beta^2 \boldsymbol{I}_d - \boldsymbol{v}_t \boldsymbol{v}_t^\top)/4$, the logistic rule gives

$$\begin{aligned}\left(\frac{1}{2}\sum_{\varepsilon=\pm 1}\phi_{\varepsilon \boldsymbol{v}_t/2,\boldsymbol{\Sigma}_t}(\boldsymbol{x})\right) p_t^+(\boldsymbol{x}) &= \frac{1}{2}\phi_{-\boldsymbol{v}_t/2,\boldsymbol{\Sigma}_t}(\boldsymbol{x}) \\ &= \frac{1}{2}\phi_{\boldsymbol{v}_t/2,\boldsymbol{\Sigma}_t}(\boldsymbol{x}+\boldsymbol{v}_t) \\ &= \left(\frac{1}{2}\sum_{\varepsilon=\pm 1}\phi_{\varepsilon \boldsymbol{v}_t/2,\boldsymbol{\Sigma}_t}(\boldsymbol{x}+\boldsymbol{v}_t)\right) p_t^-(\boldsymbol{x}+\boldsymbol{v}_t).\end{aligned}$$

Substituting the Metropolis rule, pairing opposite rejected moves, and subtracting the equal terms above yields

$$\begin{aligned}\Pr[\delta_t = 0] &= \sum_{\varepsilon=\pm 1}\int_{\mathbb{R}^d}\phi_\beta(\boldsymbol{x}) p_t^\varepsilon(\boldsymbol{x})(1 - a_t^\varepsilon(\boldsymbol{x}))\,d\boldsymbol{x} \\ &= \int_{\mathbb{R}^d}\left|\phi_\beta(\boldsymbol{x}) p_t^+(\boldsymbol{x}) - \phi_\beta(\boldsymbol{x}+\boldsymbol{v}_t) p_t^-(\boldsymbol{x}+\boldsymbol{v}_t)\right| d\boldsymbol{x} \\ &\le \int_{\mathbb{R}^d}\left|\phi_\beta(\boldsymbol{x}) - \frac{1}{2}\sum_{\varepsilon=\pm 1}\phi_{\varepsilon \boldsymbol{v}_t/2,\boldsymbol{\Sigma}_t}(\boldsymbol{x})\right| d\boldsymbol{x}.\end{aligned}$$

The inequality is the triangle inequality, followed by translation of the reverse term and $p_t^+ + p_t^- = 1$.

This absolute density difference is bounded by its unit-vector case. To see this, scale the coordinate along $\boldsymbol{v}_t$ by $\|\boldsymbol{v}_t\|_2$ and add independent Gaussian noise with variance $\beta^2(1 - \|\boldsymbol{v}_t\|_2^2)/4$. This maps the unit-vector mixture to the required mixture while preserving the Gaussian target; applying the same random transformation cannot increase the absolute density difference integrated over space.

We may therefore take $\boldsymbol{v}_t = \boldsymbol{e}_1$. The mixture's first coordinate has means $\pm 1/2$ and variance $(\beta^2 - 1)/4$ in each component, whereas the target has variance $\beta^2/4$. The common independent Gaussian coordinates integrate to one. Setting $z = 2x_1/\beta$ and applying Cauchy–Schwarz against the standard Gaussian density gives

$$\begin{aligned}\Pr[\delta_t = 0]^2 &\le \frac{1}{\sqrt{2\pi}}\int_{\mathbb{R}} e^{-z^2/2}\left(\frac{\beta}{\sqrt{\beta^2-1}} e^{-\frac{z^2+1}{2(\beta^2-1)}}\cosh\left(\frac{\beta z}{\beta^2-1}\right) - 1\right)^2 dz \\ &= \frac{\beta^2 e^{-1/(\beta^2-1)}}{(\beta^2-1)\sqrt{2\pi}}\int_{\mathbb{R}} e^{-\frac{\beta^2+1}{2(\beta^2-1)}z^2}\cosh^2\left(\frac{\beta z}{\beta^2-1}\right) dz - 1.\end{aligned}$$

For the equality, the expression before the minus one inside the square is the mixture-to-Gaussian density ratio. Its Gaussian-weighted integral is one, so expanding the square leaves its second moment minus one.

Using $\cosh^2(x) = (1 + \cosh(2x))/2$, the remaining integral is a sum of two Gaussian integrals. Completing the square gives

$$\int_{\mathbb{R}} e^{-\frac{\beta^2+1}{2(\beta^2-1)}z^2}\,dz = \sqrt{\frac{2\pi(\beta^2-1)}{\beta^2+1}},$$

$$\int_{\mathbb{R}} e^{-\frac{\beta^2+1}{2(\beta^2-1)}z^2}\cosh\left(\frac{2\beta z}{\beta^2-1}\right) dz = \sqrt{\frac{2\pi(\beta^2-1)}{\beta^2+1}} e^{2\beta^2/(\beta^4-1)}.$$

Substitution therefore yields

$$\begin{aligned}\Pr[\delta_t = 0]^2 &\le \frac{\beta^2 e^{-1/(\beta^2-1)}}{2\sqrt{\beta^4-1}}\left(1 + e^{2\beta^2/(\beta^4-1)}\right) - 1 \\ &= \frac{e^{-1/(\beta^2-1)} + e^{1/(\beta^2+1)}}{2\sqrt{1-\beta^{-4}}} - 1.\end{aligned}$$

Taking square roots and using $\Pr[\delta_t = 0] \le 1$ proves equation 10. □

Proof of Proposition 3.2. Take $d = 1$ and $\boldsymbol{v}_t = 1$ for every t. For $x \geq 0$, the probability of proposing and accepting a positive update is at most

$$p^+(x) \leq \exp\left(-\frac{4x}{\beta^2 - 1}\right).$$

By the Metropolis detailed-balance identity, the probability of an accepted negative update is

$$\begin{aligned}\Pr[\delta_t = -1 \mid \mathbf{x}_{t-1} = x] &= \min\left\{p^-(x), \frac{\phi_\beta(x-1)}{\phi_\beta(x)} p^+(x-1)\right\} \\ &\leq \exp\left(\frac{4}{\beta^2 - 1} - \frac{2}{\beta^2} - \frac{4x}{\beta^2(\beta^2 - 1)}\right).\end{aligned}$$

Thus, the probability of any accepted update decays exponentially in x, at a rate depending only on β. In particular, it is at most $1/(2T)$ when x is a sufficiently large β-dependent multiple of $\log T$. If $\mathbf{x}_0$ exceeds this threshold, the state remains unchanged until the first accepted update, and a union bound gives

$$\Pr[N_0(T) = T \mid \mathbf{x}_0 \text{ exceeds the threshold}] \geq \frac{1}{2}.$$

Finally, $\mathbf{x}_0 \sim \mathcal{N}(0, \beta^2/4)$, so a standard Gaussian tail lower bound shows that the probability of exceeding this $\Theta_\beta(\log T)$ threshold is

$$\exp\big(-O_\beta(\log^2 T)\big).$$

Multiplying the last two bounds proves the first claim. Since $\{N_0(T) = T\} \subseteq \{N_0(T) \geq \varepsilon T\}$ and $\log^2 T = o(T)$, the claimed exponential bound is impossible. □

B Experimental details and additional results

This appendix provides the experimental and reproducibility details omitted from Section 4. We first describe the real datasets used and report the complete results, then present the matched-retention study, and finally document the synthetic benchmarks. Throughout, adjusted logistic refers to the Metropolis-adjusted logistic walk.

Code and data. The experiment code, configuration files, and associated data for all experiments reported in the main text and Appendix B are available in the anonymous repository at https://anonymous.4open.science/r/LogisticWalks-4539/.

B.1 Real Datasets and results

Datasets and preprocessing. We use only the `train` arrays from the HDF5 files provided by ANN-Benchmarks (MNIST, NYTimes, SIFT1M, and GIST) and VIBE (Llama, ArXiv, and LAION, using `llama-128-ip.hdf5`, `arxiv-nomic-768-normalized.hdf5`, and `laion-clip-512-normalized.hdf5`). Query, distance, and neighbor arrays are unused. All available training rows are processed in stored order; we do not center, individually normalize, shuffle, or subsample the vectors. Table 3 records the common scaling divisor for each dataset; a divisor of one means that no rescaling is applied. When rescaling is needed, the divisor is rounded upward by one floating-point unit, after which we verify that the maximum row norm is at most one. SIFT1M and GIST have the same sequence length but differ in dimension and geometry, so comparisons across them do not isolate either factor.

Methods, implementation, and randomness. We use `float64` arithmetic and walk seeds 1000–1019 for every dataset–configuration pair. Thus, the 20 repeats vary the initialization and transition randomness while keeping the input vectors and their order fixed. The adjusted and unadjusted logistic walks use the same Gaussian initialization $\mathbf{x}_0 \sim \mathcal{N}(0, \beta^2 \boldsymbol{I}_d/4)$ and proposal rule; the unadjusted walk accepts every proposal. Metropolis acceptance is evaluated in log-space. We run the logistic walks for all T steps. SelfBalance uses $\delta = 0.01$ and threshold $30\log(dT/0.01)$; all reported runs completed successfully. Gaussian triplet uses the three-way coupling of Aden-Ali (2026). The Gaussian fixed-point implementation follows Liu et al. (2022) at $\sigma = 1$ and evaluates its alternating-series probabilities with truncation tolerance 10^{-15}. We verify complete output lengths and valid coefficient sets for every run. The unadjusted walk is stated in Appendix D.

Table 3: Real-dataset dimensions and common normalization divisors. A divisor of one indicates that the stored vectors already satisfy the unit norm constraint.

Dataset	Vectors T	Dimension d	Divisor
MNIST	60,000	784	4380.656686
NYTimes	290,000	256	1.203668
SIFT1M	1,000,000	128	511.353107
GIST	1,000,000	960	7.934343
Llama	256,921	128	33.394101
ArXiv	1,344,643	768	1.000000
LAION	1,000,448	512	1.000000

Timing protocol. All runs execute sequentially on an Apple M1 processor with 8 GB of RAM and one BLAS thread. For in-memory datasets, loop timings include discrepancy tracking but exclude data loading, normalization, validation, allocation, and random-number preparation. GIST, ArXiv, and LAION are processed in batches of 8,192 rows to fit in memory; their timings additionally include HDF5 reads, `float64` conversion, and common-factor scaling. Walk and random-number-generator states persist across batches. The reported times therefore compare these implementations under the stated protocol rather than intrinsic algorithmic complexity or end-to-end cost.

Paired comparisons. We pair discrepancy measurements by seed. For methods A and B, let $D_i = M_{A,i} - M_{B,i}$, where $M_{A,i}$ is the maximum prefix discrepancy of method A at seed i. We report the mean paired difference and its pointwise 95% confidence interval

$$\bar{D} \pm t_{19,0.975}\frac{s_D}{\sqrt{20}}, \qquad t_{19,0.975} = 2.093. \tag{11}$$

These intervals quantify variation over walk randomness for a fixed dataset and row order. They are pointwise rather than simultaneous and are not adjusted for multiple comparisons; we therefore interpret them descriptively rather than as familywise-error-controlled tests.

Table 4: Paired differences in maximum prefix discrepancy over seeds 1000–1019. Each entry is $\bar{D}\,[L, U]$, where $[L, U]$ is the pointwise 95% confidence interval in equation 11. Adjusted and unadjusted logistic use $\beta = 2$, and fixed-point uses $\sigma = 1$. A positive value means that the method named first has larger discrepancy.

Dataset	Unadjusted − Adjusted	Fixed-point − Adjusted	Triplet − Adjusted	Triplet − Unadjusted
MNIST	0.041 [−0.125, 0.208]	0.068 [−0.181, 0.317]	0.223 [−0.019, 0.466]	0.182 [−0.079, 0.443]
NYTimes	0.057 [−0.118, 0.232]	0.074 [−0.214, 0.362]	0.529 [0.271, 0.787]	0.473 [0.171, 0.774]
SIFT1M	0.259 [0.154, 0.364]	0.084 [−0.113, 0.281]	1.192 [1.025, 1.358]	0.933 [0.715, 1.150]
GIST	0.015 [−0.072, 0.101]	0.110 [−0.146, 0.365]	−0.162 [−0.396, 0.072]	−0.177 [−0.391, 0.037]
Llama	−0.029 [−0.193, 0.134]	−0.167 [−0.434, 0.099]	0.332 [−0.066, 0.731]	0.362 [−0.033, 0.756]
ArXiv	0.193 [0.035, 0.351]	0.007 [−0.201, 0.215]	1.067 [0.767, 1.368]	0.874 [0.537, 1.212]
LAION	0.180 [−0.004, 0.364]	0.121 [−0.074, 0.315]	0.965 [0.767, 1.163]	0.785 [0.616, 0.954]

Paired-comparison summary. Table 4 favors adjusted over unadjusted logistic on SIFT1M and ArXiv, where the corresponding intervals exclude zero. The triplet-minus-adjusted intervals exclude zero on NYTimes, SIFT1M, ArXiv, and LAION, whereas none of the fixed-point-minus-adjusted intervals does. The matched-retention study below provides a complementary comparison with the Gaussian fixed-point walk.

Complete real-data tables. Tables 5, 6, and 7 report all methods on all seven datasets under the same protocol. Entries are the mean ± sample standard deviation over 20 runs; these standard deviations describe variation across seeds, not confidence intervals. Zero-step values are percentages. In the β column, a dash means that the parameter is not applicable; the Gaussian fixed-point walk instead uses $\sigma = 1$. In the zero-step column, a dash denotes an identically zero rate for a fully signed method.

Table 5: Complete results for the million-row ANN benchmarks SIFT1M and GIST. Entries are mean ± sample standard deviation over 20 walk seeds. Bold marks the lowest mean discrepancy and runtime within each dataset.

SIFT1M: $T = 1{,}000{,}000, d = 128$				
Algorithm	β	Max discrepancy	Zero steps (%)	Time (s)
Metropolis-adjusted logistic	2	$\mathbf{6.850 \pm 0.254}$	1.7049 ± 0.0100	14.00 ± 1.19
Metropolis-adjusted logistic	3	10.304 ± 0.560	0.2104 ± 0.0059	13.85 ± 0.73
Metropolis-adjusted logistic	4	13.746 ± 1.033	0.0489 ± 0.0028	13.79 ± 0.91
Unadjusted logistic	2	7.109 ± 0.315	–	11.61 ± 1.38
Gaussian fixed-point	–	6.934 ± 0.335	0.8680 ± 0.0087	18.56 ± 0.86
Gaussian triplet	–	8.041 ± 0.278	–	29.14 ± 1.19
SelfBalance	–	71.026 ± 8.048	–	$\mathbf{10.93 \pm 1.01}$
GIST: $T = 1{,}000{,}000, d = 960$				
Algorithm	β	Max discrepancy	Zero steps (%)	Time (s)
Metropolis-adjusted logistic	2	5.189 ± 0.291	0.009440 ± 0.000737	18.62 ± 1.06
Metropolis-adjusted logistic	3	6.844 ± 0.751	0.001135 ± 0.000281	19.07 ± 2.22
Metropolis-adjusted logistic	4	8.069 ± 0.877	0.000245 ± 0.000173	18.96 ± 1.23
Unadjusted logistic	2	5.203 ± 0.258	–	16.31 ± 1.16
Gaussian fixed-point	–	5.298 ± 0.402	0.000005 ± 0.000022	34.47 ± 2.09
Gaussian triplet	–	$\mathbf{5.026 \pm 0.404}$	–	37.15 ± 0.77
SelfBalance	–	17.217 ± 1.643	–	$\mathbf{14.46 \pm 0.28}$

Table 6: Complete results for the remaining in-memory datasets: MNIST, NYTimes, and Llama. Entries are mean ± sample standard deviation over 20 walk seeds. Bold marks the lowest mean discrepancy and runtime within each dataset.

MNIST: $T = 60{,}000, d = 784$				
Algorithm	β	Max discrepancy	Zero steps (%)	Time (s)
Metropolis-adjusted logistic	2	$\mathbf{4.940 \pm 0.357}$	0.1019 ± 0.0134	0.96 ± 0.16
Metropolis-adjusted logistic	3	6.237 ± 0.397	0.0136 ± 0.0041	1.04 ± 0.41
Metropolis-adjusted logistic	4	7.332 ± 0.528	0.0033 ± 0.0025	0.97 ± 0.22
Unadjusted logistic	2	4.981 ± 0.470	–	0.82 ± 0.28
Gaussian fixed-point	–	5.008 ± 0.414	0.0012 ± 0.0014	1.43 ± 0.10
Gaussian triplet	–	5.163 ± 0.486	–	1.95 ± 0.09
SelfBalance	–	16.173 ± 1.825	–	$\mathbf{0.70 \pm 0.07}$
NYTimes: $T = 290{,}000, d = 256$				
Algorithm	β	Max discrepancy	Zero steps (%)	Time (s)
Metropolis-adjusted logistic	2	$\mathbf{6.920 \pm 0.452}$	0.6850 ± 0.0183	3.79 ± 0.19
Metropolis-adjusted logistic	3	9.868 ± 0.572	0.0852 ± 0.0056	3.84 ± 0.25
Metropolis-adjusted logistic	4	12.954 ± 0.984	0.0203 ± 0.0031	3.82 ± 0.31
Unadjusted logistic	2	6.977 ± 0.413	–	$\mathbf{3.04 \pm 0.12}$
Gaussian fixed-point	–	6.994 ± 0.418	0.1156 ± 0.0061	5.53 ± 0.33
Gaussian triplet	–	7.449 ± 0.384	–	8.43 ± 0.46
SelfBalance	–	62.017 ± 4.960	–	3.15 ± 0.76
Llama: $T = 256{,}921, d = 128$				
Algorithm	β	Max discrepancy	Zero steps (%)	Time (s)
Metropolis-adjusted logistic	2	6.198 ± 0.576	0.5622 ± 0.0159	3.58 ± 0.32
Metropolis-adjusted logistic	3	8.828 ± 0.812	0.0689 ± 0.0058	3.52 ± 0.12
Metropolis-adjusted logistic	4	11.184 ± 1.286	0.0170 ± 0.0021	3.51 ± 0.13
Unadjusted logistic	2	6.168 ± 0.575	–	2.91 ± 0.11
Gaussian fixed-point	–	$\mathbf{6.030 \pm 0.401}$	0.0773 ± 0.0055	5.01 ± 0.20
Gaussian triplet	–	6.530 ± 0.510	–	7.46 ± 0.17
SelfBalance	–	44.320 ± 6.935	–	$\mathbf{2.65 \pm 0.11}$

Table 7: Complete results for the batched ArXiv and LAION experiments. Entries are mean ± sample standard deviation over 20 walk seeds. Bold marks the lowest mean discrepancy and runtime within each dataset. Timings include batch loading, conversion, and scaling.

ArXiv: $T = 1{,}344{,}643$, $d = 768$				
Algorithm	β	Max discrepancy	Zero steps (%)	Time (s)
Metropolis-adjusted logistic	2	$\mathbf{6.994 \pm 0.276}$	1.7551 ± 0.0119	24.45 ± 2.37
Metropolis-adjusted logistic	3	10.381 ± 0.620	0.2169 ± 0.0048	24.37 ± 2.23
Metropolis-adjusted logistic	4	13.071 ± 0.888	0.0504 ± 0.0021	23.12 ± 1.14
Unadjusted logistic	2	7.188 ± 0.386	–	20.13 ± 1.16
Gaussian fixed-point	–	7.001 ± 0.410	0.9126 ± 0.0098	30.70 ± 2.16
Gaussian triplet	–	8.062 ± 0.587	–	46.20 ± 2.90
SelfBalance	–	53.137 ± 4.254	–	$\mathbf{18.76 \pm 1.30}$
LAION: $T = 1{,}000{,}448$, $d = 512$				
Algorithm	β	Max discrepancy	Zero steps (%)	Time (s)
Metropolis-adjusted logistic	2	$\mathbf{7.075 \pm 0.299}$	1.7605 ± 0.0146	17.59 ± 0.81
Metropolis-adjusted logistic	3	10.314 ± 0.469	0.2155 ± 0.0038	17.76 ± 1.30
Metropolis-adjusted logistic	4	13.398 ± 0.600	0.0502 ± 0.0021	16.94 ± 0.55
Unadjusted logistic	2	7.255 ± 0.340	–	14.24 ± 0.35
Gaussian fixed-point	–	7.195 ± 0.235	0.9178 ± 0.0083	22.75 ± 1.72
Gaussian triplet	–	8.040 ± 0.286	–	35.18 ± 2.61
SelfBalance	–	74.195 ± 5.643	–	$\mathbf{13.77 \pm 1.99}$

B.2 Matched-retention comparison with the Gaussian fixed-point walk

The main real-data comparison uses adjusted logistic at $\beta = 2$ and the Gaussian fixed-point walk at $\sigma = 1$. Both initial states then have covariance $\boldsymbol{I}_d$, which matches their initialization scales but not their observed zero-step rates. To compare the discrepancy–retention tradeoff more directly, we use the first $\min\{T, 100{,}000\}$ rows of MNIST, GIST, and Llama and divide all selected rows of each dataset by a single maximum-norm factor. We fix adjusted logistic at $\beta = 2$ and run walk seeds 1000–1009 for Gaussian fixed-point scales $\sigma \in \{0.5, 0.6, 0.7, 0.75, 0.8, 0.85, 0.9, 1\}$.

For each dataset, Table 8 reports the tested fixed-point scale whose mean zero-step percentage is closest to that of adjusted logistic. The paired difference is adjusted-logistic discrepancy minus fixed-point discrepancy, with a pointwise 95% paired t interval over the ten common seeds. Because the fixed-point scale is selected using the observed mean retention, the intervals do not account for this selection step. All three intervals exclude zero in favor of fixed-point, whereas adjusted logistic is faster in these implementations. Thus, this experiment supports a runtime advantage for the logistic transition but not a better observed discrepancy–retention tradeoff.

Table 8: Closest observed retention matches for adjusted logistic at $\beta = 2$. "Zero" is the mean zero-step percentage and "Disc." is the mean maximum prefix discrepancy. The difference is adjusted minus fixed-point, and the time ratio is fixed-point time divided by adjusted time.

	Adjusted logistic ($\beta = 2$)		Gaussian fixed-point match				
Dataset	Zero (%)	Disc.	σ	Zero (%)	Disc.	Difference [95% CI]	Time ratio
MNIST	0.0963	4.813	0.75	0.0898	4.172	0.641 [0.309, 0.972]	1.46×
GIST	0.0125	3.174	0.60	0.0160	2.529	0.645 [0.351, 0.940]	1.43×
Llama	0.5930	5.812	0.85	0.6494	4.890	0.922 [0.481, 1.363]	1.39×

B.3 Synthetic benchmarks

Sequence construction. We generate five oblivious sequences of $T = 100{,}000$ vectors at each of $d = 128$ (batch 1) and $d = 512$ (batch 2). Thus, the batches differ only in dimension; both use input seed 1000, the same sequence length, and the same generation rules. NumPy's `default_rng` with seed 1000 generates each entire family before any walk begins. To generate a uniform unit direction, we draw $\mathbf{g} \sim \mathcal{N}(0, \boldsymbol{I}_d)$ and set $\boldsymbol{u} = \mathbf{g}/\|\mathbf{g}\|_2$. In the stored `float64` arrays, every

normalized direction is multiplied by $1 - 8\varepsilon_{\mathrm{mach}}$ to remain within the unit ball under rounding. The five sequences are:

- *Basis cycle:* sample one uniform random permutation π of $[d]$ and set $\boldsymbol{v}_i = \boldsymbol{e}_{\pi(1+((i-1) \bmod d))}$. Thus, the same permuted basis repeats throughout the sequence.
- *Blocks:* independently draw one uniform unit direction $\boldsymbol{u}_b$ for each block $b = 0, \ldots, \lceil T/64 \rceil - 1$ and set $\boldsymbol{v}_i = \boldsymbol{u}_{\lfloor (i-1)/64 \rfloor}$. The block length is 64; directions are identical within a block and independently redrawn between blocks.
- *Correlated:* draw one shared uniform unit direction $\boldsymbol{u}$. Independently for each i, draw $\boldsymbol{\xi}_i \sim \mathcal{N}(0, 0.05^2 \boldsymbol{I}_d)$ and set $\boldsymbol{v}_i = (\boldsymbol{u} + \boldsymbol{\xi}_i)/\|\boldsymbol{u} + \boldsymbol{\xi}_i\|_2$, followed by the rounding factor above. The noise standard deviation is 0.05 per coordinate; because each row is normalized, the resulting correlation depends on d.
- *Sphere:* draw an independent uniform unit direction at each arrival.
- *Varying norms:* draw independent uniform unit directions $\boldsymbol{u}_i$ and set $\boldsymbol{v}_i = a_{1+((i-1) \bmod 6)} \boldsymbol{u}_i$, where $(a_1, \ldots, a_6) = (0.05, 0.1, 0.25, 0.5, 0.75, 1)$. The lengths cycle deterministically rather than being sampled randomly.

The sphere and varying-norm generators use the same seed and identical underlying directions within a batch, pairing the directions while isolating the effect of vector lengths. No further normalization or row shuffling is applied.

Configurations and seed conventions. For each family and batch, the input sequence is fixed across methods and across the 20 walk seeds 1000–1019. Each walk seed generates a new initial state and transition randomness; it does not generate a new input sequence. We compare adjusted logistic at $\beta \in \{2, 3, 4\}$, unadjusted logistic at $\beta = 2$, Gaussian fixed-point at $\sigma = 1$, Gaussian triplet, and SELFBALANCE, for 700 runs per batch. Every configuration processes the same ordered vectors. Tables 10 and 11 report means and sample standard deviations of maximum prefix discrepancy, zero-step percentage, and runtime.

Paired comparisons. For each batch–family pair, we compute seed-paired discrepancy differences using the same convention and pointwise interval as in equation 11. Because each synthetic input is fixed across the 20 walk seeds, these intervals quantify only walk randomness, not variation across independently generated sequences. As above, the intervals are pointwise and unadjusted for multiple comparisons.

Table 9: Seed-paired maximum prefix discrepancy differences on synthetic inputs. Each cell reports the mean difference and pointwise 95% interval over 20 walk seeds. A and U are adjusted and unadjusted logistic at $\beta = 2$, F is Gaussian fixed-point at $\sigma = 1$, and G is Gaussian triplet. Positive values mean that the method named first has larger discrepancy.

Batch	Family	U − A	F − A	G − A	G − U
1	Basis cycle	−0.100 [−0.309, 0.109]	0.100 [−0.200, 0.400]	0.850 [0.501, 1.199]	0.950 [0.595, 1.305]
1	Blocks	0.158 [0.013, 0.302]	0.098 [−0.183, 0.378]	0.946 [0.591, 1.301]	0.789 [0.424, 1.153]
1	Correlated	−0.070 [−0.254, 0.113]	−0.089 [−0.361, 0.183]	0.800 [0.541, 1.059]	0.870 [0.665, 1.075]
1	Sphere	0.060 [−0.161, 0.281]	0.058 [−0.342, 0.458]	0.802 [0.479, 1.125]	0.742 [0.530, 0.953]
1	Varying norms	−0.199 [−0.423, 0.025]	−0.217 [−0.558, 0.125]	0.385 [0.055, 0.716]	0.585 [0.335, 0.834]
2	Basis cycle	−0.050 [−0.234, 0.134]	0.150 [−0.125, 0.425]	1.050 [0.695, 1.405]	1.100 [0.800, 1.400]
2	Blocks	0.072 [−0.146, 0.290]	0.150 [−0.080, 0.380]	1.068 [0.714, 1.422]	0.996 [0.705, 1.287]
2	Correlated	0.013 [−0.173, 0.200]	0.182 [−0.110, 0.473]	0.947 [0.622, 1.272]	0.934 [0.642, 1.225]
2	Sphere	−0.066 [−0.264, 0.132]	−0.111 [−0.318, 0.095]	0.705 [0.448, 0.963]	0.771 [0.568, 0.975]
2	Varying norms	−0.129 [−0.353, 0.095]	−0.029 [−0.338, 0.279]	0.386 [0.069, 0.703]	0.515 [0.253, 0.777]

The G − A and G − U intervals exclude zero in favor of the logistic walks on all ten fixed inputs; this is not a simultaneous 95% statement. Only the batch-1 blocks comparison has a U − A interval excluding zero, and it favors adjusted logistic. No F − A interval excludes zero. Thus, the small gaps between the logistic and Gaussian fixed-point walks remain unresolved by these 20 trajectories, while the separation from Gaussian triplet is consistent across both dimensions and all five geometries.

Complete synthetic results. Tables 10 and 11 report mean ± sample standard deviation over the 20 walk seeds. The input is fixed within each batch–family pair, so the standard deviations describe walk randomness only. Gaussian fixed-point uses $\sigma = 1$; the zero-step percentage of each fully signed method is identically zero and is shown by a dash.

Table 10: Complete synthetic results for batch 1 ($T = 100{,}000$, $d = 128$). Entries are mean ± sample standard deviation over 20 walk seeds. Bold marks the lowest mean discrepancy and runtime within each sequence family.

Algorithm	β	Max discrepancy	Zero steps (%)	Time (s)
Basis cycle				
Metropolis-adjusted logistic	2	5.950 ± 0.394	1.7571 ± 0.0339	1.406 ± 0.064
Metropolis-adjusted logistic	3	9.100 ± 0.718	0.2222 ± 0.0169	1.403 ± 0.071
Metropolis-adjusted logistic	4	12.100 ± 1.071	0.0538 ± 0.0078	1.392 ± 0.064
Unadjusted logistic	2	$\mathbf{5.850 \pm 0.489}$	–	$\mathbf{1.167 \pm 0.119}$
Gaussian fixed-point	–	6.050 ± 0.510	0.8896 ± 0.0506	1.902 ± 0.071
Gaussian triplet	–	6.800 ± 0.523	–	2.927 ± 0.085
SelfBalance	–	57.000 ± 5.840	–	1.184 ± 0.312
Blocks				
Metropolis-adjusted logistic	2	$\mathbf{5.727 \pm 0.427}$	1.7499 ± 0.0409	0.819 ± 0.056
Metropolis-adjusted logistic	3	8.613 ± 0.799	0.2196 ± 0.0153	0.817 ± 0.056
Metropolis-adjusted logistic	4	11.568 ± 1.034	0.0548 ± 0.0060	0.813 ± 0.039
Unadjusted logistic	2	5.884 ± 0.398	–	0.665 ± 0.053
Gaussian fixed-point	–	5.824 ± 0.498	0.9012 ± 0.0263	1.105 ± 0.035
Gaussian triplet	–	6.673 ± 0.518	–	1.739 ± 0.067
SelfBalance	–	55.387 ± 5.288	–	$\mathbf{0.616 \pm 0.015}$
Correlated				
Metropolis-adjusted logistic	2	6.314 ± 0.377	1.7494 ± 0.0345	1.369 ± 0.074
Metropolis-adjusted logistic	3	9.014 ± 0.652	0.2156 ± 0.0163	1.380 ± 0.101
Metropolis-adjusted logistic	4	12.065 ± 0.985	0.0512 ± 0.0062	1.581 ± 0.845
Unadjusted logistic	2	6.244 ± 0.293	–	1.169 ± 0.181
Gaussian fixed-point	–	$\mathbf{6.225 \pm 0.366}$	0.9130 ± 0.0318	1.817 ± 0.175
Gaussian triplet	–	7.114 ± 0.384	–	3.022 ± 0.638
SelfBalance	–	41.211 ± 7.713	–	$\mathbf{1.079 \pm 0.254}$
Sphere				
Metropolis-adjusted logistic	2	$\mathbf{6.651 \pm 0.557}$	1.7456 ± 0.0299	1.549 ± 0.572
Metropolis-adjusted logistic	3	9.764 ± 0.517	0.2192 ± 0.0164	1.695 ± 0.802
Metropolis-adjusted logistic	4	12.453 ± 0.742	0.0529 ± 0.0067	1.493 ± 0.278
Unadjusted logistic	2	6.711 ± 0.386	–	1.236 ± 0.321
Gaussian fixed-point	–	6.710 ± 0.520	0.9038 ± 0.0302	1.998 ± 0.308
Gaussian triplet	–	7.453 ± 0.287	–	3.184 ± 0.527
SelfBalance	–	57.499 ± 5.985	–	$\mathbf{1.143 \pm 0.248}$
Varying norms				
Metropolis-adjusted logistic	2	6.514 ± 0.573	0.3644 ± 0.0171	0.787 ± 0.015
Metropolis-adjusted logistic	3	9.150 ± 0.708	0.0463 ± 0.0073	0.838 ± 0.219
Metropolis-adjusted logistic	4	12.080 ± 1.095	0.0116 ± 0.0050	0.786 ± 0.013
Unadjusted logistic	2	6.315 ± 0.389	–	0.636 ± 0.009
Gaussian fixed-point	–	$\mathbf{6.297 \pm 0.516}$	0.1537 ± 0.0129	2.222 ± 0.024
Gaussian triplet	–	6.900 ± 0.352	–	1.697 ± 0.016
SelfBalance	–	41.377 ± 6.310	–	$\mathbf{0.610 \pm 0.014}$

Table 11: Complete synthetic results for batch 2 ($T = 100{,}000$, $d = 512$). Entries are mean $\pm$ sample standard deviation over 20 walk seeds. Bold marks the lowest mean discrepancy and runtime within each sequence family.

Algorithm	β	Max discrepancy	Zero steps (%)	Time (s)
Basis cycle				
Metropolis-adjusted logistic	2	6.100 ± 0.308	1.7464 ± 0.0400	0.986 ± 0.116
Metropolis-adjusted logistic	3	9.200 ± 0.696	0.2207 ± 0.0153	0.929 ± 0.068
Metropolis-adjusted logistic	4	12.200 ± 0.951	0.0524 ± 0.0064	0.944 ± 0.098
Unadjusted logistic	2	$\mathbf{6.050 \pm 0.394}$	–	0.791 ± 0.106
Gaussian fixed-point	–	6.250 ± 0.550	0.9135 ± 0.0289	1.399 ± 0.282
Gaussian triplet	–	7.150 ± 0.587	–	2.131 ± 0.293
SelfBalance	–	43.250 ± 3.683	–	$\mathbf{0.706 \pm 0.066}$
Blocks				
Metropolis-adjusted logistic	2	$\mathbf{5.217 \pm 0.322}$	1.7599 ± 0.0448	1.039 ± 0.154
Metropolis-adjusted logistic	3	7.810 ± 0.413	0.2185 ± 0.0156	1.057 ± 0.200
Metropolis-adjusted logistic	4	10.489 ± 1.035	0.0546 ± 0.0081	1.070 ± 0.190
Unadjusted logistic	2	5.289 ± 0.250	–	0.881 ± 0.224
Gaussian fixed-point	–	5.368 ± 0.376	0.9104 ± 0.0356	1.378 ± 0.158
Gaussian triplet	–	6.286 ± 0.651	–	2.201 ± 0.342
SelfBalance	–	40.471 ± 3.513	–	$\mathbf{0.783 \pm 0.108}$
Correlated				
Metropolis-adjusted logistic	2	$\mathbf{6.584 \pm 0.400}$	1.7486 ± 0.0378	1.027 ± 0.156
Metropolis-adjusted logistic	3	9.585 ± 0.542	0.2172 ± 0.0190	1.045 ± 0.166
Metropolis-adjusted logistic	4	12.136 ± 0.737	0.0533 ± 0.0082	1.036 ± 0.164
Unadjusted logistic	2	6.598 ± 0.331	–	0.854 ± 0.119
Gaussian fixed-point	–	6.766 ± 0.422	0.9092 ± 0.0284	1.447 ± 0.265
Gaussian triplet	–	7.531 ± 0.557	–	2.243 ± 0.305
SelfBalance	–	33.538 ± 2.590	–	$\mathbf{0.737 \pm 0.069}$
Sphere				
Metropolis-adjusted logistic	2	6.832 ± 0.391	1.7581 ± 0.0375	0.895 ± 0.077
Metropolis-adjusted logistic	3	9.830 ± 0.511	0.2208 ± 0.0169	0.908 ± 0.127
Metropolis-adjusted logistic	4	12.914 ± 0.799	0.0538 ± 0.0080	0.897 ± 0.103
Unadjusted logistic	2	6.766 ± 0.368	–	0.758 ± 0.167
Gaussian fixed-point	–	$\mathbf{6.721 \pm 0.346}$	0.9152 ± 0.0271	1.184 ± 0.050
Gaussian triplet	–	7.537 ± 0.300	–	1.900 ± 0.170
SelfBalance	–	44.004 ± 5.870	–	$\mathbf{0.698 \pm 0.112}$
Varying norms				
Metropolis-adjusted logistic	2	6.494 ± 0.462	0.3667 ± 0.0189	0.933 ± 0.059
Metropolis-adjusted logistic	3	9.133 ± 0.709	0.0459 ± 0.0059	0.959 ± 0.144
Metropolis-adjusted logistic	4	11.669 ± 0.681	0.0122 ± 0.0041	0.946 ± 0.089
Unadjusted logistic	2	$\mathbf{6.365 \pm 0.314}$	–	0.777 ± 0.058
Gaussian fixed-point	–	6.465 ± 0.324	0.1530 ± 0.0153	2.664 ± 0.275
Gaussian triplet	–	6.880 ± 0.475	–	2.057 ± 0.348
SelfBalance	–	26.378 ± 3.245	–	$\mathbf{0.706 \pm 0.036}$

C SELF-BALANCING WALK OF ALWEISS, LIU, AND SAWHNEY

For completeness, Algorithm 2 restates Algorithm 1 of Alweiss et al. (2021) using our notation. The walk begins at the origin and sets $c = 30 \log(dT/\delta)$. Before processing each vector, it verifies that both the current infinity-norm discrepancy and the inner product used by the transition rule are at most c. On this event, the probability p_t lies in $[0, 1]$.

Algorithm 2 SELFBALANCE$(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_T, \delta)$ (Alweiss et al., 2021)

Require: $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_T \in \mathbb{R}^d$ with $\|\boldsymbol{v}_t\|_2 \le 1$ and $\delta \in (0, 1)$

$c \leftarrow 30 \log(dT/\delta)$
$\boldsymbol{w}_0 \leftarrow 0$
for $t = 1, \ldots, T$ **do**
  if $|\langle \boldsymbol{w}_{t-1}, \boldsymbol{v}_t \rangle| > c$ or $\|\boldsymbol{w}_{t-1}\|_\infty > c$ **then**
    return FAIL
  $p_t \leftarrow \dfrac{1}{2} - \dfrac{\langle \boldsymbol{w}_{t-1}, \boldsymbol{v}_t \rangle}{2c}$
  Set $\varepsilon_t \leftarrow +1$ with probability p_t, and $\varepsilon_t \leftarrow -1$ otherwise
  $\boldsymbol{w}_t \leftarrow \boldsymbol{w}_{t-1} + \varepsilon_t \boldsymbol{v}_t$
return $(\varepsilon_1, \ldots, \varepsilon_T)$

D UNADJUSTED LOGISTIC WALK

Algorithm 3 uses the same state-dependent proposal as the Metropolis-adjusted logistic walk and accepts every proposed sign. Each update uses only the current state and the incoming vector. We use $\beta = 2$ in the experiments. Gaussian initialization alone does not ensure that the auxiliary state retains a Gaussian marginal for changing input vectors.

Algorithm 3 LOGISTICWALK(β)

Require: $\beta > 1$ and incoming vectors $\boldsymbol{v}_t \in \mathbb{R}^d$ with $\|\boldsymbol{v}_t\|_2 \le 1$

$\mathbf{x}_0 \sim \mathcal{N}(0, \beta^2 \boldsymbol{I}_d/4)$
$\mathbf{w}_0 \leftarrow 0$
for $t = 1, \dots, T$ **do**
  Define $p_t^+(\boldsymbol{x}) \leftarrow \mathrm{sigmoid}\left(-\dfrac{4\langle \boldsymbol{x}, \boldsymbol{v}_t \rangle}{\beta^2 - \|\boldsymbol{v}_t\|_2^2}\right)$
  Set $\delta_t \leftarrow +1$ with probability $p_t^+(\mathbf{x}_{t-1})$, and $\delta_t \leftarrow -1$ otherwise
  $\mathbf{x}_t \leftarrow \mathbf{x}_{t-1} + \delta_t \boldsymbol{v}_t$
  $\mathbf{w}_t \leftarrow \mathbf{x}_t - \mathbf{x}_0$
return $(\delta_1, \dots, \delta_T)$