9. APPENDIX: Stochastic Flows and Geometric Optimization on the Orthogonal Group

9.1. Hyperparameters and Training for CNNs

9.1.1. Supervised Learning

In the Plain-110 task on CIFAR10, we performed grid search across the following parameters and values in the orthogonal setting:

Hyperparameter	Values
learning rate (LR)	{0.05, 0.1, 0.5}
use bias (whether layers use bias)	{False, True}
batch size	{128, 1024, 8196}
maximum epoch length	{100, 300, 900}
scaling on LR for orthogonal integrator	{0.1, 1.0, 10.0}

For the vanilla baselines, we used a momentum optimizer with the same settings found in (Xie et al., 2017; He et al., 2016) (0.9 momentum, 0.1 learning rate, 128 batch size). The learning rate decay schedule occurs when the epoch number is {3/9, 6/9, 8/9} of the maximum epoch length.

We also used a similar hyperparameter sweep for the MLP task on MNIST.

9.1.2. Extra Training Details

For the CIFAR10 results from Figure 5, to understand the required computing resources to train PlainNet-110, we further found that **numerical issues** using the exact integrator could occur when using a naive variant of the matrix exponential. In particular, when the Taylor series truncation $\sum_{k=0}^{T} \frac{1}{k!}\mathbf{X}^k$ for approximating e^X is too short (such as even $T = 100$), PlainNet-110 could not reach $\geq 80\%$ *training* accuracy, showing that achieving acceptable precision on the matrix exponential can require a large amount of truncations. An acceptable truncation length was found at T = 200. Furthermore, library functions (e.g. `tensorflowf.linalg.expm` (Higham, 2009)), albeit using optimized code, are still inherently limited to techniques computing these truncations as well.

For the cluster-based stochastic integrators, we set the cluster size for each parameter matrix $\mathbf{M} \in \mathbb{R}^{d,k}$ to be the rounding-up of $\frac{d}{\log d}$. We found that this was an optimal choice, as sizes such as $O(\log d), O(\sqrt{d})$ did not train properly.

9.2. Orthogonal Optimization for RL - Additional Details

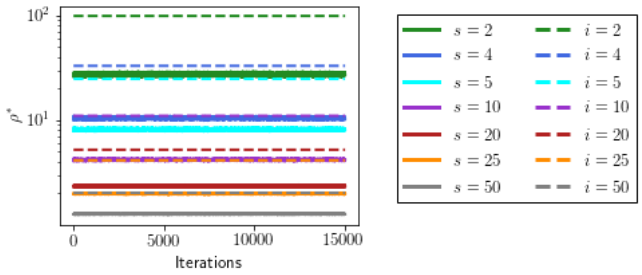
We conducted extensive ablation studies to see whether the assumption that one can take upper bound τ for τ^* (see: Section 3) of the order $O(\frac{s}{d\alpha\beta})\|h(\Omega)\|_1$ for small constants α^{-1}, β^{-1} is valid. In other words, we want to see whether $\frac{1}{\tau^*}$ can be lower-bounded by expressions of the order $\Omega(\frac{d\alpha\beta}{s}\frac{1}{\|h(\Omega)\|_1})$. We took Humanoid environment and the setting as in Section 6.1. Note that by the definition of τ^* we trivially have: $\frac{1}{\tau^*} \geq \rho\frac{1}{\|h(\Omega)\|_1}$, where $\rho = \frac{\|h(\Omega)\|_1}{\gamma_\Omega(s,d,h)}$ and $\gamma_\Omega(s,d,h)$ is the sum of the $\frac{d}{s}\binom{s}{2}$ entries of $h(\Omega)$ with largest absolute values.

In Fig. 6 we plot ρ as a function of the number of iterations of the training procedure. Dotted lines correspond to the values $\frac{d}{s}$. The y-axis uses log-scale.

We tested different sizes $s = 2, 4, 5, 10, 20, 25, 50$ and took the size of the hidden layer to be 200 (thus $d = 200$). We noticed that for a fixed s, values of ρ do not change much over time and can be accurately approximated by constants (in Fig. 6 they look almost line the plots of constant functions $y = \text{const}$, even though we observed small perturbations). Furthermore, they can be accurately approximated by renormalized values $\frac{d}{s}$, where renormalization factor c is such that c^{-1} is a small positive constant. That suggests two things:

- τ^* can be in practice upper-bounded by expressions of the form $O(\frac{s}{d\alpha\beta})\|h(\Omega)\|_1$ for small positive constant α^{-1}, β^{-1} and:
- magnitudes of entries of skew-symmetric matrices in applications from Section 6.1 tend to be very similar.

Of course, as explained in the main body of the paper, those findings enable to further improve speed of our sampling procedures.



(a)

Figure 6. Value of $\rho = \frac{\|h(\Omega)\|_1}{\gamma_\Omega(s,d,h)}$ as a function of the number of iterations of the optimization as in Section 6.1 for Humanoid and for different sizes $s = 2, 4, 5, 10, 20, 25, 50$. Dotted lines correspond to values $\frac{d}{s}$ that approximate (up to the positive multiplicative constant that is not too small) values of ρ. We see that for a fixed s, values of ρ almost do not change over the course of optimization and in fact can be accurately approximated by plots of constant functions. We use the log-scale for y-axis.

9.3. Instability of Deterministic Methods for the Optimization on the Orthogonal Group

To demonstrate numerical problems of the deterministic optimizers/integrators on the orthogonal group $\mathcal{O}(d)$, we considered the following matrix differential equation on $\mathcal{O}(d)$:

$$\dot{\mathbf{X}}(t) = \mathbf{X}(t)(\mathbf{N}\mathbf{X}(t)^\top \mathbf{Q}\mathbf{X} - \mathbf{X}^\top \mathbf{Q}\mathbf{X}(t)\mathbf{N}), \tag{11}$$

where $\mathbf{N} = \mathrm{diag}(n_1, ., , , n_d)$, $\mathbf{Q} = \mathrm{diag}(q_1, ..., q_d)$ for some scalars $n_1, ..., n_d, q_1, ..., q_d \in \mathbb{R}$, and furthermore $n_i \neq n_j$ and $q_i \neq q_j$ for $i \neq j$. We also assume that $\mathbf{X}(0) \in \mathcal{O}(d)$. Matrix $\Omega(t) = \mathbf{N}\mathbf{X}(t)^\top \mathbf{Q}\mathbf{X} - \mathbf{X}^\top \mathbf{Q}\mathbf{X}(t)\mathbf{N}$ is clearly skew-symmetric thus the above differential equation encodes flow evolving on $\mathcal{O}(d)$ (see: Sec. 2).

It can be proven that for all matrices $\mathbf{X} \in \mathcal{O}(d)$, but a set of measure zero the following holds:

$$\mathbf{X}(t) \xrightarrow{t} \mathbf{P}, \tag{12}$$

where $\mathbf{P}$ is a permutation matrix corresponding to the permutation $(r_1, ..., r_d)$ of $(q_1, ..., q_d)$ that maximizes the expression:

$$x_1 n_1 + ... + x_d n_d \tag{13}$$

over all permutations $(x_1, ..., x_d)$ of $(q_1, ..., q_d)$. Since Expression 13 is maximized for the permutation $(x_1, ..., x_d)$ s.t. $x_i < x_j$ iff $n_i < n_j$, we conclude that the flow which is a solution to Eq. 11 can be applied to sort numbers (e.g. one can take $(n_1, ..., n_d) = (1, ..., d)$ to sort in the increasing order). Furthermore, we can use our techniques to conduct integration.

In our experiments we compared our algorithm (using non-intersecting families with $s = 2$) with the deterministic integrator based on exact exponential mapping. We chose $\mathbf{X}(0)$ to be a random orthogonal matrix that we obtained by constructing Gaussian matrix and then conducting Gram-Schmidt orthogonalization and row-renormalization.

	$\eta = 0.00001$	$\eta = 0.00005$	$\eta = 0.0001$	$\eta = 0.00015$	$\eta = 0.001$	$\eta = 0.0015$	$\eta = 0.01$	$\eta = 0.015$	$\eta = 0.1$	$\eta = 0.15$
ϵ: stoch	e-13	2.0e-12	1.5e-12	1.8e-12	1.65e-12	1.2e-12	1.78e-12	1.3e-12	1.45e-12	1.3e-12
inv: stoch	1.0	1.0	1.0	1.0	1.0	0.8	0.67	0.6	0.59	0.58
ϵ: exact	e-14	2.0e-14	1.5e-13	1.8e-13	nan	nan	nan	nan	nan	nan
inv: exact	0.8	0.75	0.72	0.52	0.0	0.0	0.0	0.0	0.0	0.0

Table 2: Comparison of the stochastic integrator with the exact one on the problem of sorting numbers with flows evolving on $\mathcal{O}(d)$. First two rows correspond to the stochastic integrator and last two to the exact one. The error ϵ is defined as: $\epsilon = \|\mathbf{X}_{\mathrm{final}}\mathbf{X}_{\mathrm{final}}^\top - \mathbf{I}_d\|_{\mathcal{F}}$. Value of inv is the fraction of inverse pairs. For large enough step size exact integrator starts to produce numerical errors that accumulate over time and break integration.

We focused on quantifying numerical instabilities of both methods by conducting ablation studies over different values of step size $\eta > 0$. For each method we run $n = 10$ experiments, each with different random sequence $(q_1, ..., q_d)$. We chose: $N = \mathrm{diag}(d, d-1, ..., 1)$ thus the goal was to sort in the decreasing order. To conduct sorting, we run $\frac{50.0}{\eta}$ iterations of both algorithms. Denote by $\mathbf{X}_{\mathrm{final}}$ the matrix obtained by conducting integration. We computed $\|\mathbf{X}_{\mathrm{final}}\mathbf{X}_{\mathrm{final}}^\top - \mathbf{I}_d\|_{\mathcal{F}}$ to measure the deviation from the orthogonal group $\mathrm{O}(d)$. Matrix $\mathbf{X}_{\mathrm{final}}$ was projected back to the permutation group that

was then used to obtain permuted version $(p_1, ..., p_d)$ of the original sequence $(q_1, ..., q_d)$ The quality of the final result was measured in the number of inverses, i.e. pairs (p_i, p_j) such that $i < j$ but $p_i > p_j$. For perfect sorting all the pairs (p_i, p_j) such that $i < j$ are inverses. As we see in Table 2, if step size is too large exact method produces matrices with infinite field values and the algorithm fails.

9.4. The Geometry of the Orthogonal Group & Riemannian Optimization

In this section we provide additional technical terminology that we use in the main body of the paper.

9.4.1. Smooth Curves on Manifolds

Definition 9.1 (smooth curves on $\mathcal{M}$)**.** *A function $\gamma : I \to \mathcal{M}$, where $I \subseteq \mathbb{R}$ is an open interval is a smooth curve on $\mathcal{M}$ passing through $\mathbf{p} \in \mathcal{M}$ if there exists $\phi : \Omega_{\mathbf{p}} \to U_{\mathbf{p}}$ for open subsets $\Omega_p \subseteq \mathbb{R}^d$, $U_{\mathbf{p}} \subseteq \mathcal{M}$ and $\epsilon > 0$ such that the function $\phi^{-1} \circ \gamma : (t - \epsilon, t + \epsilon) \to \mathbb{R}^d$ is smooth.*

Vectors tangent to smooth curves γ on $\mathcal{M}$ passing through fixed point $\mathbf{p} \in \mathcal{M}$ give rise to the linear subspace tangent to $\mathcal{M}$ at $\mathbf{p}$, the *tangent space* $\mathcal{T}_{\mathbf{p}}(\mathcal{M})$ that we define in the main body.

9.4.2. Inner Products

Standard inner products used for $\mathcal{ST}(d, k)$ are: the *Euclidean inner product* defined as: $\langle \mathbf{Z}_1, \mathbf{Z}_2 \rangle_{\mathrm{e}} = \mathrm{tr}(\mathbf{Z}_1^\top \mathbf{Z}_2)$ and the *canonical inner product* given as: $\langle \mathbf{Z}_1, \mathbf{Z}_2 \rangle_c = \mathrm{tr}(Z_1^\top (I - \frac{1}{2}\mathbf{X}\mathbf{X}^\top))Z_2$ for a tangent space in $\mathbf{X} \in \mathcal{ST}(d, k)$.

9.4.3. Representation Theorems for On-Manifold Optimization

We need the following standard representation theorem:

Theorem 9.2 (representation theorem)**.** *If $\langle \cdot \rangle$ is an inner product defined on the vector space $\mathcal{R}$, then for any linear functional $L : \mathcal{R} \to \mathbb{R}$ there exists $\mathbf{R} \in \mathcal{R}$ s.t. $\langle \mathbf{R}, \mathbf{Q} \rangle = L(\mathbf{Q})$ for any $\mathbf{Q} \in \mathcal{R}$.*

To apply the above result for on-manifold optimization, we identify:

- L with the directional derivative operator related to the function F being optimized,
- $\mathcal{R}$ with the tangent space,
- $\mathbf{R}$ with the Riemannian gradient.

9.5. Theoretical Results for Sampling Algorithms

Below we prove all the theoretical results from Section 3.

Lemma 9.3. *We state two useful combinatorial facts and one of their consequences:*

- $|\mathcal{T}_s| = \dfrac{d!}{(s!)^{d/s}(\frac{d}{s})!}$
- *Each edge appears in* $W = \dfrac{(d-2)!}{(\frac{d-s}{s})!(s!)^{\frac{d-s}{s}}(s-2)!}$ *tournaments of* $|\mathcal{T}_s|$
- *Therefore, for* $p \sim \mathcal{U}(\mathcal{T}_s)$, $\dfrac{1}{p_T}\mathbf{M}_{\mathcal{T}_s} = \dfrac{d-1}{s-1}\mathbf{J}_d$

Proof. • To compute $|\mathcal{T}_s|$, we can use the way we sample them: we choose a random permutation and take the s first vertices to be the first connected component, the s next vertices to be the second etc... This way, multiple random permutations will lead to the same tournament. More precisely, exactly $\dfrac{d}{s}!(s!)^{\frac{d}{s}}$ permutations lead to the same tournament.

Therefore $|\mathcal{T}_s| = \dfrac{d!}{(s!)^{d/s}(\frac{d}{s})!}$.

- By symmetry, we know that each edge appears in the same number of tournaments of $\mathcal{T}_s$. Let W be this number. Let N_T be the number of edges in the tournament T. We have that $N_T = \frac{d}{s}\binom{s}{2}$. Therefore $\sum_{T\in\mathcal{T}_s} N_T = |\mathcal{T}_s|\frac{d}{s}\binom{s}{2}$. We also have that $\sum_{T\in\mathcal{T}_s} N_T = W\binom{d}{2}$. Therefore $|\mathcal{T}_s|\frac{d}{s}\binom{s}{2} = W\binom{d}{2}$ That gives:

$$\begin{aligned} W &= \frac{d!\frac{d}{s}\binom{s}{2}}{\binom{d}{2}(\frac{d}{s})!(s!)^{d/s}} \\ &= \frac{(d-2)!s(s-1)}{(\frac{d}{s}-1)!(s!)^{d/s}} \\ &= \frac{(d-2)!}{(\frac{d-s}{s})!(s!)^{\frac{d-s}{s}}(s-2)!} \\ W &= \frac{(d-2)!}{(\frac{d-s}{s})!(s!)^{\frac{d-s}{s}}(s-2)!} \end{aligned}$$

- For $p\sim\mathcal{U}(\mathcal{T}_s)$, $p_T = \frac{1}{|\mathcal{T}_s|}$. Therefore, $\frac{1}{p_T}\mathbf{M}_{\mathcal{T}_s} = \frac{|\mathcal{T}_s|}{W}\mathbf{J}_d = \frac{d-1}{s-1}\mathbf{J}_d$

□

9.5.1. Proof of Lemma 3.2

Below we prove Lemma 3.2 from the main body of the paper.

Proof. Let $\Omega\in\mathrm{Sk}(d)$ be a skew-symmetric matrix . Fix a family $\mathcal{T}$ of subtournaments of $T(\Omega)$. We aim to show that the distribution $\mathcal{P}$ over $\mathcal{T}$ minimizing the variance $\mathrm{Var}(\widehat{\Omega}) = \mathbb{E}[\|\widehat{\Omega}-\Omega\|_{\mathcal{F}}^2]$ among unbiased distributions of the form given by equations 5 and 6, satisfies: $p_T\sim\sqrt{\sum_{e\in E(G_T)} w_e^2}$, where w_e is the weight of edge e.

The constraint on the scalars $\{p_T\}_{T\in\mathcal{T}}$ is simply that the family $\{p_T\}_{T\in\mathcal{T}}$ forms a valid probability distribution. The unbiasedness is guaranteed by the equations 5 and 6.

The variance rewrites:

$$\mathrm{Var}(\widehat{\Omega}) = \mathbb{E}[\|\widehat{\Omega}-\Omega\|_{\mathcal{F}}^2] = \mathbb{E}[\|\widehat{\Omega}\|_{\mathcal{F}}^2] - \|\Omega\|_{\mathcal{F}}^2$$

Then we consider the following functional,

$$\begin{aligned} f(\mathcal{P}) &= \mathbb{E}_{\mathcal{P}}[\|\widehat{\Omega}\|_{\mathcal{F}}^2] \\ &= \sum_{T\in\mathcal{T}} p_T\cdot\left\|\frac{1}{p_T}M_{\mathcal{T}}\odot\Omega[T]\right\|_F^2 \\ &= \sum_{T\in\mathcal{T}}\frac{2}{p_T}\sum_{(i,j)\in E(G_T)} M_{\mathcal{T}}[i,j]^2\cdot\Omega[i,j]^2 \qquad \text{the order } i,j \text{ does not matter} \end{aligned}$$

We minimize the functional f on the convex open domain $\{\mathcal{P}=\{p_T\}_{T\in\mathcal{T}}\in(\mathbb{R}_{>0})^{\mathcal{T}}, \sum_{T\in\mathcal{T}} p_T = 1\}$ on which f is convex. The Lagrangian has the form:

$$L(\mathcal{P},\lambda) = \sum_{T\in\mathcal{T}}\frac{2}{p_T}\sum_{(i,j)\in E(G_T)} M_{\mathcal{T}}[i,j]^2\cdot\Omega[i,j]^2 + 2\lambda\left(\sum_{T\in\mathcal{T}} p_T - 1\right)$$

and the global optimum can be found from equations:

$$\frac{\partial}{\partial p_T} L(\mathcal{P}, \lambda) = -\frac{2}{p_T^2} \sum_{(i,j)\in E(G_T)} M_{\mathcal{T}}[i,j]^2 \cdot \Omega[i,j]^2 + 2\lambda = 0$$

We finally obtain the optimal $\mathcal{P}$:

$$p_T = \frac{\sqrt{\sum_{(i,j)\in E(G_T)} (M_{\mathcal{T}} \odot \Omega)[i,j]^2}}{Z} \tag{14}$$

where $Z = \sum_{T\in\mathcal{T}} p_T$.

We find that the smallest variance is then given by:

$$\text{Var}^* \left(\widehat{\Omega}\right) = 2 \cdot \left(\sum_{T\in\mathcal{T}} \sqrt{\sum_{(i,j)\in E(G_T)} (M_{\mathcal{T}} \odot \Omega)[i,j]^2} \right)^2 - \|\Omega\|_{\mathcal{F}}^2$$

In case of homogeneous families, $M_{\mathcal{T}}$ has identical coefficients and the constant $M_{\mathcal{T}}$ vanishes into the normalization constant Z. □

9.5.2. Proof of Lemma 3.4

Below we prove Lemma 3.4 from the main body of the paper.

Proof. Let A_k be the random variable which is 1 if the kth sample is accepted and 0 otherwise and T_k be the kth sampled tournament. A_k are iid Bernoulli variables of parameter $\frac{\lambda}{|\mathcal{T}_s|}$.

$$\begin{aligned}
\mathbb{P}[A_1 = 1] &= \sum_{T\in\mathcal{T}_s} \mathbb{P}[A_1 = 1 | T_1 = T]\, \mathbb{P}[T_1 = T] \\
&= \frac{1}{|\mathcal{T}_s|} \sum_{T\in\mathcal{T}_s} \mathbb{P}[A_1 = 1 | T_1 = T] \\
&= \frac{1}{|\mathcal{T}_s|} \sum_{T\in\mathcal{T}_s} q_T^h = \frac{\lambda}{|\mathcal{T}_s|} \sum_{T\in\mathcal{T}_s} p_T^h \\
&= \frac{\lambda}{|\mathcal{T}_s|}
\end{aligned}$$

The number of trials before a sample is accepted is $\min\{k | A_k = 1\}$. This random variable follows a Poisson distribution of parameter $\frac{|\mathcal{T}_s|}{\lambda}$. Therefore, the expected number of trials before a sample is accepted is $\frac{|\mathcal{T}_s|}{\lambda}$

□

9.5.3. Proof of Lemma 3.5

Below we prove Lemma 3.5 from the main body of the paper.

Proof. By definition, $p_T^h \sim h(G_T)$. Let call α the proportionality factor. Then

$$\alpha^{-1} = \sum_{T\in\mathcal{T}_s} h(G_T) = \sum_{T\in\mathcal{T}_s} \sum_{e\in E(G_T)} h(w_e)$$

$$
\begin{aligned}
&= \sum_{T\in\mathcal{T}_s}\sum_{i<j} h(w_{(i,j)})\mathbb{1}\left((i,j)\in E(G_T)\right)\\
&= \sum_{i<j} h(w_{(i,j)}) \sum_{T\in\mathcal{T}_s} \mathbb{1}\left((i,j)\in E(G_T)\right)\\
&= W\sum_{i<j} h(w_{(i,j)}) = W\frac{1}{2}\sum_{i,j} h(w_{(i,j)})\\
&= W\frac{1}{2}\sum_{i,j} h(\Omega_{(i,j)}) = \frac{W\|h(\Omega)\|_1}{2}
\end{aligned}
$$

As $W = \sum_{i<j} h(w_{(i,j)})\mathbb{1}\left((i,j)\in E(G_T)\right)$. The computation of W is done in the proof of Lemma 9.3. □

9.5.4. Proof of Theorem 3.6

Below we prove Theorem 3.6 from the main body of the paper.

Proof. The time complexity results is a direct consequence of Lemma 3.4. We just need to prove that Algorithm 1 returns a sample of $\mathcal{P}^h(\mathcal{T}_s)$. We use the random variables A_k and T_k defined in the proof of Lemma 3.4. Let A be the output of Algorithm 1.

Let $T\in\mathcal{T}_s$. We have to check that $\mathbb{P}\left[A=T\right] = p_T^h$. For this, we notice that $\{A=T\} = \cup_{k=1}^{+\infty}\{A_k = 1 \cap T_k = T \cap_{i=1}^{k-1} A_i = 0\}$. These events being disjoints, we have:

$$
\begin{aligned}
\mathbb{P}\left[A=T\right] &= \sum_{k=1}^{+\infty}\mathbb{P}\left[A_k = 1\cap T_k = T\cap_{i=1}^{k-1} A_i = 0\right]\\
&= \sum_{k=1}^{+\infty}\mathbb{P}\left[A_k = 1\cap T_k = T|\cap_{i=1}^{k-1} A_i = 0\right]\mathbb{P}\left[\cap_{i=1}^{k-1} A_i = 0\right]\\
&= \sum_{k=1}^{+\infty}\mathbb{P}\left[A_k = 1\cap T_k = T|\cap_{i=1}^{k-1} A_i = 0\right]\left(1-\frac{\lambda}{|\mathcal{T}_s|}\right)^{k-1}\\
&= \sum_{k=1}^{+\infty}\mathbb{P}\left[A_k = 1|T_k = T\cap_{i=1}^{k-1} A_i = 0\right]\frac{1}{|\mathcal{T}_s|}\left(1-\frac{\lambda}{|\mathcal{T}_s|}\right)^{k-1}\\
&= \sum_{k=1}^{+\infty} q_T^h\frac{1}{|\mathcal{T}_s|}\left(1-\frac{\lambda}{|\mathcal{T}_s|}\right)^{k-1}\\
&= p_T^h\frac{\lambda}{|\mathcal{T}_s|}\sum_{k=0}^{+\infty}\left(1-\frac{\lambda}{|\mathcal{T}_s|}\right)^{k}\\
\mathbb{P}\left[A=T\right] &= p_T^h
\end{aligned}
$$

Therefore Algorithm 1 samples from $\mathcal{P}^h(\mathcal{T}_s)$. □

9.5.5. Estimating $\|h(\Omega)\|_1$

Denote $n = \binom{d}{2}$. Consider matrix $h(\Omega)\in\mathbb{R}^{d\times d}$. We will approximate $\|h(\Omega)\|_1$ as:

$$
X = \sum_{i,j} X_{i,j}, \tag{15}
$$

for $1\leq i<j\leq d$ and where $X_{i,j} = \frac{n}{r}h(\Omega_{i,j})$ with probability $\frac{r}{n}$ and $X_{i,j}=0$ otherwise. Note that $\mathbb{E}[X] = \|h(\Omega)\|_1$ and furthermore the expected number R of nonzero entries $X_{i,j}$ is clearly r. Now it suffices to notice that R is strongly

concentrated around its mean using standard concentration inequalities (such as Azuma's inequality). Furthermore, for any $a > 0$, by Azuma's inequality, we have:

$$\mathbb{P}[X - \mathbb{E}[X] > a] \leq \exp(-\frac{a^2}{2(\frac{n}{r})^2 \sum_{i,j} h^2(\Omega_{i,j})}). \tag{16}$$

The upper bound is clearly smaller than $\exp(-(\frac{\epsilon\alpha\beta r}{3d})^2)$ for $a = \epsilon\|h(\Omega)\|_1$ and $(\alpha\beta, h)$-balanced Ω. That directly leads to the results regarding approximating $\|h(\Omega)\|_1$ by sub-sampling Ω from the main body of the paper.

9.6. Variance Results

Below we present variance results of the estimators of skew-symmetric matrices Ω studied in the main body of the paper.

Lemma 9.4 (Variance of h-regular estimators)**.** *The variance of an estimator $\hat{\Omega}$ following an h-regular distribution over $\mathcal{T}_s$ is*

$$\mathrm{Var}(\hat{\Omega}) = \frac{\|h(\Omega)\|_1}{2W} \sum_{T \in \mathcal{T}_s} \frac{\|\Omega[T]\|_{\mathcal{F}}^2}{h(G_T)} - \|\Omega\|_{\mathcal{F}}^2$$

Proof.

$$\begin{aligned}
\mathrm{Var}(\hat{\Omega}) &= \sum_{T \in \mathcal{T}_s} p_T^h \|\Omega_T\|_{\mathcal{F}}^2 - \|\Omega\|_{\mathcal{F}}^2 \\
&= \sum_{T \in \mathcal{T}_s} \frac{1}{W^2 p_T^h} \|\Omega[T]\|_{\mathcal{F}}^2 - \|\Omega\|_{\mathcal{F}}^2 \\
&= \sum_{T \in \mathcal{T}_s} \frac{\|h(\Omega)\|_1}{2W h(G_T)} \|\Omega[T]\|_{\mathcal{F}}^2 - \|\Omega\|_{\mathcal{F}}^2 \\
&= \frac{\|h(\Omega)\|_1}{2W} \sum_{T \in \mathcal{T}_s} \frac{\|\Omega[T]\|_{\mathcal{F}}^2}{h(G_T)} - \|\Omega\|_{\mathcal{F}}^2
\end{aligned}$$

□

Lemma 9.5. *Let $\hat{\Omega}$ be the h-regular estimator over $\mathcal{T}_s$ where h is the squared function. Then* $\mathrm{Var}(\hat{\Omega}) = \frac{d-s}{s-1}\|\Omega\|_{\mathcal{F}}^2$

Proof. Using lemma 9.4 with h being the squared function gives:

$$\begin{aligned}
&= \frac{\|\Omega\|_{\mathcal{F}}^2}{W} \sum_{T \in \mathcal{T}_s} 1 - \|\Omega\|_{\mathcal{F}}^2 \quad \text{as } 2h(G_T) = \|\Omega[T]\|_{\mathcal{F}}^2 \\
&= \|\Omega\|_{\mathcal{F}}^2 \left(\frac{|\mathcal{T}_s|}{W} - 1 \right) \\
&= \|\Omega\|_{\mathcal{F}}^2 \left(\frac{d-1}{s-1} - 1 \right) \quad \text{as seen in the proof of Lemma 9.3}
\end{aligned}$$

Therefore:

$$\mathrm{Var}(\hat{\Omega}) = \frac{d-s}{s-1}\|\Omega\|_{\mathcal{F}}^2$$

□

Lemma 9.6. *Let $\hat{\Omega}$ be uniformly distributed over $\mathcal{T}_s$. Then* $\mathrm{Var}(\hat{\Omega}) = \frac{d-s}{s-1}\|\Omega\|_{\mathcal{F}}^2$

Proof. Let $\hat{\Omega}$ be uniformly distributed over $\mathcal{T}_s$. We have:

$$\begin{aligned}
\mathrm{Var}(\hat{\Omega}) &= \sum_{T\in\mathcal{T}_s}\frac{1}{|\mathcal{T}_s|}\|\Omega_T\|_{\mathcal{F}}^2 - \|\Omega\|_{\mathcal{F}}^2 \\
&= \frac{1}{|\mathcal{T}_s|}\sum_{T\in\mathcal{T}_s}\frac{(d-1)^2}{(s-1)^2}\|\Omega[T]\|_{\mathcal{F}}^2 - \|\Omega\|_{\mathcal{F}}^2 \\
&= \frac{1}{|\mathcal{T}_s|}\frac{(d-1)^2}{(s-1)^2}\sum_{T\in\mathcal{T}_s}\sum_{(i,j)\in T} 2\Omega_{i,j}^2 - \|\Omega\|_{\mathcal{F}}^2 \\
&= \frac{1}{|\mathcal{T}_s|}\frac{(d-1)^2}{(s-1)^2}\sum_{T\in\mathcal{T}_s}\sum_{i<j} 2\Omega_{i,j}^2\mathbb{1}\left((i,j)\in E(G_T)\right) - \|\Omega\|_{\mathcal{F}}^2 \\
&= \frac{1}{|\mathcal{T}_s|}\frac{(d-1)^2}{(s-1)^2}\sum_{i<j} 2\Omega_{i,j}^2\sum_{T\in\mathcal{T}_s}\mathbb{1}\left((i,j)\in E(G_T)\right) - \|\Omega\|_{\mathcal{F}}^2 \\
&= \frac{W}{|\mathcal{T}_s|}\frac{(d-1)^2}{(s-1)^2}\sum_{i,j}\Omega_{i,j}^2 - \|\Omega\|_{\mathcal{F}}^2 \\
&= \frac{d-s}{s-1}\|\Omega\|_{\mathcal{F}}^2 \quad \text{as } \frac{W}{\mathcal{T}_s} = \frac{s-1}{d-1} \text{ as seen in the proof of Lemma 9.3}
\end{aligned}$$

So $\mathrm{Var}(\hat{\Omega}) = \frac{d-s}{s-1}\|\Omega\|_{\mathcal{F}}^2$ □

9.7. The Combinatorics of Domain-Optimization for Sampling Subtournaments

In this section we provide additional theoretical results regarding variance of certain classes of the proposed estimators of skew-symmetric matrices Ω and establish deep connection with challenging problems in graph theory and combinatorics. We will be interested in particular in shaping the family of tournaments $\mathcal{T}$ on-the-fly to obtain low-variance estimators. Even though we did not need these extensions to obtain the results presented in the main body of the paper, we discuss them in more detail here due to the interesting connections with combinatorial optimization. We will focus here on non-intersecting families $\mathcal{T}$ and $s = 2$. Thus the corresponding undirected graphs are just matchings and they altogether cover all the edges of the base complete undirected weighted graph $G_{T(\Omega)}$.

9.7.1. More on the variance

We will denote the family of all these matchings as $\mathcal{M}$. and start with function $h : \mathbb{R} \rightarrow \mathbb{R}$ given as: $h(x) = |x|$. The following is true:

Lemma 9.7 (variance of matching-based estimators for non-intersecting families and $h(x) = |x|$)**.** *Given a skew-symmetric matrix Ω and the corresponding complete weighted graph $G_{T(\Omega)}$ with the set of edge-weights $\{w_e\}_{e\in E(G_{T(\Omega)})}$, the variance/mean squared error of the unbiased estimator $\widehat{\Omega}$ applying function $h(x) = |x|$ and family of matchings $\mathcal{M}$ satisfies:*

$$\begin{aligned}
&\mathrm{MSE}(\widehat{\Omega}) = \mathrm{Var}(\widehat{\Omega}) \\
&= \mathbb{E}[\|\widehat{\Omega} - \Omega\|_{\mathcal{F}}^2] = K\sum_{e\in E(G_{T(\Omega)})}\frac{w_e^2}{K(e)} - \|\Omega\|_{\mathcal{F}}^2, \qquad (17)
\end{aligned}$$

where $K(e)$ stands for the sum of absolute values of weights of the edges of the matching $m \in \mathcal{M}$ containing e and K for the sum of all the absolute values of all the weights.

Proof. We have the following for $\mathbf{V}_m$ defined as: $\mathbf{V}_m = \sum_{e\in m}\frac{|a_{i,j}|}{K_m}K\mathrm{sgn}(a_{i,j})\mathbf{H}_{i,j}$,where m stands for the matching,

and K_m is the sum of weights of matching m:

$$\mathbb{E}[\|\widehat{\Omega}-\Omega\|_{\mathcal{F}}^2]=\mathbb{E}[\|\widehat{\Omega}\|_{\mathcal{F}}^2]-\|\Omega\|_{\mathcal{F}}^2$$
$$=\sum_{m\in\mathcal{M}}p_m\|\mathbf{V}_m\|_F^2-\|\Omega\|_{\mathcal{F}}^2=\sum_{m\in\mathcal{M}}p_m\sum_{e\in m}\frac{w_e^2}{K_m^2}K^2-\|\Omega\|_{\mathcal{F}}^2=$$
$$K^2\sum_{m\in\mathcal{M}}\frac{K_m}{K}\sum_{e\in m}\frac{w_e^2}{K_m^2}-\|\Omega\|_{\mathcal{F}}^2= \tag{18}$$
$$K\sum_{m\in\mathcal{M}}\frac{1}{K_m}\sum_{e\in m}w_e^2-\|\Omega\|_{\mathcal{F}}^2=K\sum_{e\in E(G_{T(\Omega)})}\frac{w_e^2}{K(e)}-\|\Omega\|_{\mathcal{F}}^2,$$

where p_m is the probability of choosing matching $m\in\mathcal{M}$, i.e. $p(m)=\frac{\sum_{e\in m}|w_e|}{K}=\frac{K_m}{K}$. □

Thus the variance minimization problem reduces to finding a family of matchings $\mathcal{M}$ which minimizes $\sum_{e\in E(G_{T(\Omega)})}\frac{w_e^2}{K(e)}$.

Let us list a couple of observations. First, if every matching is a single edge (that would correspond to conducting **exactly one** multiplication by Givens rotation per iteration of the optimization procedure using an estimator) the variance is the largest. Intuitively speaking, we would like to have in $\mathcal{M}$ lots of heavy-weight matchings. ideally if $\mathcal{M}$ consists of just one matching covering all nonzero-weight edges (the zero-weight edges can be neglected) the variance is the smallest and in fact equals to 0 since then we take entire matrix Ω. There are lots of heuristics that can be used such as taking maximum weight matching (see: (Micali & Vazirani, 1980)) in $G_{T(\Omega)}$ as the first matching, delete it from graph, take the second largest maximum weight matching and continue to construct entire $\mathcal{M}$. Since finding maximum weight matching requires nontrivial computational time such an approach would work best if we reconstruct $\mathcal{M}$ periodically, as opposed to doing it in every single step of the optimization procedure. Interestingly, it can be shown that this algorithm, even though working very well in practice accuracy-wise, does not minimize the variance (one can find counterexamples with graphs as small as of six vertices). The following is true:

Lemma 9.8 (Variance minimization vs. NP-hardness)**.** *Given a weighted and undirected graph G, the problem of finding a partition of the edges into matchings $\mathcal{M}$ which minimizes $\sum_{e\in E(G)}\frac{w_e^2}{K(e)}$ is NP-hard.*

Proof. There is a one-to-one correspondence between partitions of the edges into matchings $\mathcal{M}$ and edge-colorings. Thus, we will reduce to the problem of computing the chromatic index of an arbitrary graph G, which is known to be NP-complete (see (Holyer, 1981)).

Take an arbitrary G and set all its weights w_e equal to 1. Then we claim the optimal objective value of the optimization problem is the chromatic index of G. Indeed,

$$\sum_{e\in E(G)}\frac{w_e^2}{K(e)}=\sum_{e\in E(G)}\frac{1}{K(e)}=$$
$$\sum_{e\in E(G)}\frac{1}{\#\{e'\in m:e\in m\}}=\#\mathcal{M}$$

(where $\#A$ denotes the cardinality of A). Thus the expression which minimizes the sum on the LHS is the smallest possible cardinality of the set $\mathcal{M}$, which is the chromatic index of G, and thus we have completed the reduction. □

The above result shows an intriguing connection between stochastic optimization on the orthogonal group and graph theory. Notice that we know (see: Lemma 3.2) that under assumptions regarding estimator from Lemma 3.2, the optimal variance is achieved if p_m is proportional to the square root of the sum of squares of the weights of all its edges. Thus one can instead use such a distribution $\{p_m\}_{m\in\mathcal{M}}$ instead the one generated by function h. It is an interesting question whether optimizing family of matchings $\mathcal{M}$ (thus we still focus on the case $s=2$) in such a setting can be done in the polynomial time. We leave it to future work.

9.7.2. DISTRIBUTED COMPUTATIONS FOR ON-MANIFOLD OPTIMIZATION

The connection with maximum graph matching problem suggests that one can apply distributed computations to construct on-the-fly families $\mathcal{M}$ used to conduct sampling. Maximum weight matching is one of the most-studied algorithmic problems in graph theory and the literature on fast distributed optimization algorithms constructing approximations of the maximum weight matching is voluminous (see for instance: (Czumaj et al., 2018),(Lattanzi et al., 2011),(Assadi et al., 2019)). Such an approach might be particularly convenient if we want to update $\mathcal{M}$ at every single iteration of the optimization procedure and dimensionality d is very large.

9.7.3. ON-MANIFOLD OPTIMIZATION VS. GRAPH SPARSIFICATION PROBLEM

Finally, we want to talk about the connection with graph sparsification techniques. Instead of partitioning into matchings the original graph G_T, one can instead sparsify G_T first and then conduct partitioning into matchings of the sparsified graph. This strategy can bypass potentially expensive computations of the heavy-weight matchings in the original dense graph by those in its sparser compact representation. That leads to the theory of graph sparsification and graph sketches (Chu et al., 2018) that we leave to future work.

9.8. Theorem 5.1 Proof

Proof. Consider the i-th step of the update rule. Denote $g(\eta) = F(\exp(\eta\widehat{\Omega}_i)\mathbf{X}_i)$. Then by a chain rule we get

$$g'(\eta) = \langle \nabla F(\exp(\eta\widehat{\Omega}_i)\mathbf{X}_i), \exp(\eta\widehat{\Omega}_i)\widehat{\Omega}_i\mathbf{X}_i\rangle_{\mathrm{e}}$$

Next we deduce

$$\begin{aligned}
&|g'(\eta) - g'(0)| = |\langle \nabla F(\exp(\eta\widehat{\Omega}_i)\mathbf{X}_i), \exp(\eta\widehat{\Omega}_i)\widehat{\Omega}_i\mathbf{X}_i\rangle_{\mathrm{e}} - \langle \nabla F(\mathbf{X}_i), \widehat{\Omega}_i\mathbf{X}_i\rangle_{\mathrm{e}}| \\
&= |\langle \nabla F(\exp(\eta\widehat{\Omega}_i)\mathbf{X}_i), \exp(\eta\widehat{\Omega}_i)\widehat{\Omega}_i\mathbf{X}_i\rangle_{\mathrm{e}} - \langle \nabla F(\mathbf{X}_i), \exp(\eta\widehat{\Omega}_i)\widehat{\Omega}_i\mathbf{X}_i\rangle_{\mathrm{e}} + \langle \nabla F(\mathbf{X}_i), \exp(\eta\widehat{\Omega}_i)\widehat{\Omega}_i\mathbf{X}_i\rangle_{\mathrm{e}} \\
&- \langle \nabla F(\mathbf{X}_i), \widehat{\Omega}_i\mathbf{X}_i\rangle_{\mathrm{e}}| \\
&\leq |\langle \nabla F(\exp(\eta\widehat{\Omega}_i)\mathbf{X}_i) - \nabla F(\mathbf{X}_i), \exp(\eta\widehat{\Omega}_i)\widehat{\Omega}_i\mathbf{X}_i\rangle_{\mathrm{e}}| + |\langle \nabla F(\mathbf{X}_i), \exp(\eta\widehat{\Omega}_i)\widehat{\Omega}_i\mathbf{X}_i - \widehat{\Omega}_i\mathbf{X}_i\rangle_{\mathrm{e}}| \\
&\leq \|\nabla F(\exp(\eta\widehat{\Omega}_i)\mathbf{X}_i) - \nabla F(\mathbf{X}_i)\|_{\mathcal{F}} \|\exp(\eta\widehat{\Omega}_i)\widehat{\Omega}_i\mathbf{X}_i\|_{\mathcal{F}} + \|\nabla F(\mathbf{X}_i)\|_{\mathcal{F}} \|\exp(\eta\widehat{\Omega}_i)\widehat{\Omega}_i\mathbf{X}_i - \widehat{\Omega}_i\mathbf{X}_i\|_{\mathcal{F}} \quad (19) \\
&= \|\nabla F(\exp(\eta\widehat{\Omega}_i)\mathbf{X}_i) - \nabla F(\mathbf{X}_i)\|_{\mathcal{F}} \|\widehat{\Omega}_i\|_{\mathcal{F}} + \|\nabla F(\mathbf{X}_i)\|_{\mathcal{F}} \|\exp(\eta\widehat{\Omega}_i)\widehat{\Omega}_i - \widehat{\Omega}_i\|_{\mathcal{F}} \quad (20) \\
&\leq L\|\exp(\eta\widehat{\Omega}_i)\mathbf{X}_i - \mathbf{X}_i\|_{\mathcal{F}} \|\widehat{\Omega}_i\|_{\mathcal{F}} + \|\nabla F(\mathbf{X}_i)\|_{\mathcal{F}} \|\exp(\eta\widehat{\Omega}_i)\widehat{\Omega}_i - \widehat{\Omega}_i\|_{\mathcal{F}} \quad (21) \\
&= L\|\exp(\eta\widehat{\Omega}_i) - \mathbf{I}_d\|_{\mathcal{F}} \|\widehat{\Omega}_i\|_{\mathcal{F}} + \|\nabla F(\mathbf{X}_i)\|_{\mathcal{F}} \|\exp(\eta\widehat{\Omega}_i)\widehat{\Omega}_i - \widehat{\Omega}_i\|_{\mathcal{F}} \quad (22) \\
&\leq L\|\exp(\eta\widehat{\Omega}_i) - \mathbf{I}_d\|_{\mathcal{F}} \|\widehat{\Omega}_i\|_{\mathcal{F}} + \|\nabla F(\mathbf{X}_i)\|_{\mathcal{F}} \|\exp(\eta\widehat{\Omega}_i) - \mathbf{I}_d\|_{\mathcal{F}} \|\widehat{\Omega}_i\|_{\mathcal{F}} \quad (23)
\end{aligned}$$

where a) in transition 19 we use Cauchy-Schwarz inequality, b) in 20, 22 we use invariance of the Frobenius norm under orthogonal mappings, c) in 21 we use 10 and d) in 23 we use sub-multiplicativity of Frobenius norm. We further derive that

$$\begin{aligned}
\|\nabla F(\mathbf{X}_i)\|_{\mathcal{F}} &\leq \|\nabla F(\mathbf{X}_i) - \nabla F(\mathbf{I}_d)\|_{\mathcal{F}} + \|\nabla F(\mathbf{I}_d)\|_{\mathcal{F}} \leq L\|\mathbf{X}_i - \mathbf{I}_d\|_{\mathcal{F}} + \|\nabla F(\mathbf{I}_d)\|_{\mathcal{F}} \\
&\leq L(\|\mathbf{X}_i\|_{\mathcal{F}} + \|\mathbf{I}_d\|_{\mathcal{F}}) + \|\nabla F(\mathbf{I}_d)\|_{\mathcal{F}} = 2L\sqrt{d} + \|\nabla F(\mathbf{I})\|_{\mathcal{F}}
\end{aligned}$$

where we use that $\|\mathbf{X}_i\|_{\mathcal{F}} = \|\mathbf{I}_d\|_{\mathcal{F}} = \sqrt{d}$ due to orthogonality. Now we have

$$|g'(\eta) - g'(0)| \leq \Big((2\sqrt{d}+1)L + \|\nabla F(\mathbf{I}_d)\|_{\mathcal{F}}\Big)\|\widehat{\Omega}_i\|_{\mathcal{F}} \cdot \|\exp(\eta\widehat{\Omega}_i) - \mathbf{I}_d\|_{\mathcal{F}} \quad (24)$$

Next, we employ Theorem 12.9 from (Gallier, 2011) which states that, due to its skew-symmetry, $\widehat{\Omega}_i$ can be decomposed as $\widehat{\Omega}_i = \mathbf{P}\mathbf{E}\mathbf{P}^\top$ where $\mathbf{P} \in \mathcal{O}(d)$ and $\mathbf{E}$ is a block-diagonal matrix of form:

$$\mathbf{E} = \begin{bmatrix} \mathbf{E}_1 & & \\ & \dots & \\ & & \mathbf{E}_p \end{bmatrix}$$

such that each block $\mathbf{E}_j$ is either $\begin{bmatrix}0\end{bmatrix}$ or a two-dimensional matrix of form

$$\mathbf{E}_j = \begin{bmatrix} 0 & -\mu_j \\ \mu_j & 0 \end{bmatrix}$$

for some $\mu_j \in \mathbb{R}$. From this we deduce that

$$\exp(\eta\widehat{\Omega}_i) - \mathbf{I}_d = \mathbf{P}\mathbf{J}\mathbf{P}^\top$$

where $\mathbf{J}$ is block-diagonal matrix of type

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_1 & & \\ & \dots & \\ & & \mathbf{J}_p \end{bmatrix}$$

where for each j $\mathbf{J}_j = \exp(\eta\mathbf{E}_j) - \mathbf{I}$ where $\mathbf{I}$ is either $\mathbf{I}_1$ or $\mathbf{I}_2$. Hence, for each j $\mathbf{J}_j$ is either $\begin{bmatrix}0\end{bmatrix}$ or a two-dimensional matrix of the form

$$\mathbf{J}_j = \begin{bmatrix} \cos(\eta\mu_j) - 1 & -\sin(\eta\mu_j) \\ \sin(\eta\mu_j) & \cos(\eta\mu_j) - 1 \end{bmatrix}$$

Denote by $\mathcal{J}$ the set of indices j from $\{1, \dots p\}$ which correspond to two-dimensional blocks of $\mathbf{E}$ and $\mathbf{J}$. Then

$$\begin{aligned} \|\exp(\eta\widehat{\Omega}_i) - \mathbf{I}_d\|_{\mathcal{F}}^2 = \|\mathbf{P}\mathbf{J}\mathbf{P}^\top\|_{\mathcal{F}}^2 = \|\mathbf{J}\|_{\mathcal{F}}^2 = 2\sum_{j\in\mathcal{J}}\Big(\sin^2(\eta\mu_j) + (\cos(\eta\mu_j) - 1)^2\Big) = 4\sum_{j\in\mathcal{J}}\Big(1 - \cos(\eta\mu_j)\Big) \\ \leq 2\sum_{j\in\mathcal{J}}(\eta\mu_j)^2 = \eta^2\|\mathbf{E}\|_{\mathcal{F}}^2 = \eta^2\|\widehat{\Omega}_i\|_{\mathcal{F}}^2 \end{aligned}$$

where we use the inequality $1 - \cos x \leq \frac{x^2}{2}$. Therefore we can rewrite Equation 24 as

$$|g'(\eta) - g'(0)| \leq \Big((2\sqrt{d}+1)L + \|\nabla F(\mathbf{I}_d)\|_{\mathcal{F}}\Big)\|\widehat{\Omega}_i\|_{\mathcal{F}}^2 \cdot |\eta| \leq L_g \cdot |\eta|$$

where $L_g = \Big((2\sqrt{d}+1)L + \|\nabla F(\mathbf{I}_d)\|_{\mathcal{F}}\Big)\|\widehat{\Omega}_i\|_{\mathcal{F}}^2$. We further deduce:

$$g(\eta) - g(0) - \eta g'(0) = \int_0^\eta \Big(g'(\tau) - g'(0)\Big)d\tau \geq -\int_0^\eta \Big|g'(\tau) - g'(0)\Big|d\tau \geq -\int_0^\eta L_g|\tau|d\tau = -\frac{\eta^2}{2}L_g \tag{25}$$

We unfold g's definition, put $\eta = \eta_i$ and rewrite 25 as follows:

$$\eta_i\langle\nabla F(\mathbf{X}_i), \widehat{\Omega}_i\mathbf{X}_i\rangle_{\mathrm{e}} \leq F(\mathbf{X}_{i+1}) - F(\mathbf{X}_i) + \frac{\eta_i^2}{2}L_g \tag{26}$$

Recall that from $\widehat{\Omega}_i$'s definition we have that $\mathbb{E}\widehat{\Omega}_i = \Omega_i = \nabla_{\mathcal{O}}F(\mathbf{X}_i)\mathbf{X}_i^\top$. By taking expectation w.r.t. random $\widehat{\Omega}_i$ sampling at i's step from both sides of Equation 26 we obtain that

$$\eta_i\langle\nabla F(\mathbf{X}_i), \nabla F_{\mathcal{O}}(\mathbf{X}_i)\rangle_{\mathrm{e}} \leq \mathbb{E}F(\mathbf{X}_{i+1}) - F(\mathbf{X}_i) + \frac{\eta_i^2}{2}\mathbb{E}L_g$$

Since the Riemannian gradient can be expressed as $\nabla F_{\mathcal{O}}(\mathbf{X}_i) = (\nabla F(\mathbf{X}_i)\mathbf{X}_i^\top - \mathbf{X}_i\nabla F(\mathbf{X}_i)^\top)\mathbf{X}_i$, we have that

$$\begin{aligned} \|\nabla F_{\mathcal{O}}(\mathbf{X}_i)\|_{\mathcal{F}}^2 = \|\nabla F(\mathbf{X}_i)\mathbf{X}_i^\top - \mathbf{X}_i\nabla F(\mathbf{X}_i)^\top\|_{\mathcal{F}}^2 = \mathrm{tr}\Big((\nabla F(\mathbf{X}_i)\mathbf{X}_i^\top - \mathbf{X}_i\nabla F(\mathbf{X}_i)^\top)^\top\nabla F(\mathbf{X}_i)\mathbf{X}_i^\top\Big) \\ + \mathrm{tr}\Big((\mathbf{X}_i\nabla F(\mathbf{X}_i)^\top - \nabla F(\mathbf{X}_i)\mathbf{X}_i^\top)^\top\mathbf{X}_i\nabla F(\mathbf{X}_i)^\top\Big) \\ = \mathrm{tr}\Big(\mathbf{X}_i^\top(\nabla F(\mathbf{X}_i)\mathbf{X}_i^\top - \mathbf{X}_i\nabla F(\mathbf{X}_i)^\top)^\top\nabla F(\mathbf{X}_i)\Big) + \mathrm{tr}\Big(\nabla F(\mathbf{X}_i)^\top(\nabla F(\mathbf{X}_i)\mathbf{X}_i^\top - \mathbf{X}_i\nabla F(\mathbf{X}_i)^\top)\mathbf{X}_i\Big) \end{aligned}$$

$$= 2\langle(\nabla F(\mathbf{X}_i)\mathbf{X}_i^\top - \mathbf{X}_i\nabla F(\mathbf{X}_i)^\top)\mathbf{X}_i, \nabla F(\mathbf{X}_i)\rangle_{\mathrm{e}} = 2\langle\nabla F_{\mathcal{O}}(\mathbf{X}_i), \nabla F(\mathbf{X}_i)\rangle_{\mathrm{e}}$$

where we use that $\mathrm{tr}(\mathbf{A}^\top\mathbf{B}) = \mathrm{tr}(\mathbf{B}^\top\mathbf{A})$ and $\mathrm{tr}(\mathbf{AB}) = \mathrm{tr}(\mathbf{BA})$. Hence

$$\eta_i\|\nabla F_{\mathcal{O}}(\mathbf{X}_i)\|_{\mathcal{F}}^2 \leq 2\Big(\mathbb{E}F(\mathbf{X}_{i+1}) - F(\mathbf{X}_i)\Big) + \eta_i^2\mathbb{E}L_g \tag{27}$$

$$\leq 2\Big(\mathbb{E}F(\mathbf{X}_{i+1}) - F(\mathbf{X}_i)\Big) + \eta_i^2\Big((2\sqrt{d}+1)L + \|\nabla F(\mathbf{I}_d)\|_{\mathcal{F}}\Big)\sigma^2 \tag{28}$$

By taking expectation of Equation 28 w.r.t. $\widehat{\Omega}_i$ random sampling at steps $i = \overline{0..T}$ and summing over all these steps one arrives at

$$\sum_{i=0}^{T}\eta_i\mathbb{E}\|\nabla F_{\mathcal{O}}(\mathbf{X}_i)\|_{\mathcal{F}}^2 \leq 2\mathbb{E}\Big(F(\mathbf{X}_{i+1}) - F(\mathbf{X}_0)\Big) + \sigma^2\Big((2\sqrt{d}+1)L + \|\nabla F(\mathbf{I}_d)\|_{\mathcal{F}}\Big)\sum_{i=0}^{T}\eta_i^2$$

$$\leq 2\Big(F^* - F(\mathbf{X}_0)\Big) + \sigma^2\Big((2\sqrt{d}+1)L + \|\nabla F(\mathbf{I}_d)\|_{\mathcal{F}}\Big)\sum_{i=0}^{T}\eta_i^2$$

Finally we use that

$$\Big[\sum_{i=0}^{T}\eta_i\Big]\cdot\min_{i=\overline{0..T}}\mathbb{E}\|\nabla F_{\mathcal{O}}(\mathbf{X}_i)\|_{\mathcal{F}}^2 \leq \sum_{i=0}^{T}\eta_i\mathbb{E}\|\nabla F_{\mathcal{O}}(\mathbf{X}_i)\|_{\mathcal{F}}^2$$

and divide by $\sum_{i=0}^{T}\eta_i$ to conclude the proof. □

9.9. Stochastic Optimization on the Orthogonal Group vs Recent Results on Givens Rotations for ML

There is an interesting relation between algorithms for stochastic optimization on the orthogonal group $\mathcal{O}(d)$ proposed by us and some results results about applying Givens rotations in machine learning.

Givens Neural Networks: In (Choromanski et al., 2019) the authors propose neural network architectures, where matrices of connections are encoded as **trained products** of Givens rotations. They demonstrate that such architectures can be effectively used for neural network based policies in reinforcement learning and furthermore provide the compactification of the parameters that need to be learned. Notice that such matrices of connections correspond to consecutive steps of the matching-based optimizers/integrators proposed by us. This points also to an idea of neural ODEs that are constrained to evolve on compact manifolds (such as an orthogonal group).

Approximating Haar measure: Approximating Haar measure on the orthogonal group $\mathcal{O}(d)$ was recently shown to have various important applications in machine learning, in particular for kernel methods (Choromanski et al., 2018) and in general in the theory of Quasi Monte Carlo sequences (Rowland et al., 2018). Some of the most effective methods conduct approximations through products of random Givens matrices (Choromanski et al., 2018). It turns out that we can think about this problem through the lens of matrix differential equations encoding flows evolving on $\mathcal{O}(d)$. Consider the following DE on the orthogonal group:

$$\dot{\mathbf{X}}(t) = \mathbf{X}(t)\Omega_{\mathrm{rand}}(t) \tag{29}$$

with an initial condition: $\mathbf{X}(0) \in \mathcal{O}(d)$. It turns out that when $\Omega_{\mathrm{rand}}(t)$ is "random enough" (one can take for instance Gaussian skew-symmetric matrices with large enough standard deviations of each entry or random walk skew-symmetric matrices, where each entry of the upper triangular part is an independent long enough random walk on a discrete 1d-lattice $\{0, 1, -1, 2, -2, ...\}$), the above differential equation describes a flow on $\mathcal{O}(d)$ such that for $T \to \infty$ the distribution of $\mathbf{X}(t)$ converges to Haar measure. Equation 29 is also connected to heat kernels on $\mathcal{O}(d)$.

Interestingly, if we use our stochastic matching-based methods for integrating such a flow, we observe that the solution is a product of **random** Givens rotations. Furthermore, these products tend to have the property that vertices/edges corresponding to different Givens rotations do not appear for consecutive elements that often as for the standard method (for instance, every block of Givens rotations corresponding to one step of the integration uses different edges since they correspond to a valid matching). We do believe that such property helps to obtain even stronger mixing properties in comparison to standard

mechanism. Finally, these products of Givens rotations can be seen right now as a special instantiation of a much more general mechanism, since nothing prevents us from using our methods with $s > 2$ rather than $s = 2$ to conduct integration. That provides a convenient way to trade-off accuracy of the estimator versus its speed. We leave detailed analysis of the applications of our methods in that context to future work.