
Approximating Orthogonal Matrices with Effective Givens Factorization

Thomas Frerix¹ Joan Bruna²

Abstract

We analyze effective approximation of unitary matrices. In our formulation, a unitary matrix is represented as a product of rotations in two-dimensional subspaces, so-called Givens rotations. Instead of the quadratic dimension dependence when applying a dense matrix, applying such an approximation scales with the number of factors, each of which can be implemented efficiently. Consequently, in settings where an approximation is once computed and then applied many times, such a representation becomes advantageous. Although effective Givens factorization is not possible for generic unitary operators, we show that minimizing a sparsity-inducing objective with a coordinate descent algorithm on the unitary group yields good factorizations for structured matrices. Canonical applications of such a setup are orthogonal basis transforms. We demonstrate numerical results of approximating the graph Fourier transform, which is the matrix obtained when diagonalizing a graph Laplacian.

1. Introduction

Unitary operators are ubiquitous in many areas, from numerical linear algebra to quantum computing and cryptography. Celebrated applications include the QR-decomposition and the diagonalization of symmetric matrices (Golub & Van Der Vorst, 2000). Without any assumptions on the structure of the matrix, applying a unitary transformation in d dimensions requires $\mathcal{O}(d^2)$ operations for the matrix-vector product. In scenarios where a given unitary operator needs to be intensively applied many times, using approximations that trade-off accuracy with a better scaling behavior in the dimension is desirable.

In this paper, we develop a method to compute approxima-

tions of unitary matrices in the form of Givens factorization (Givens, 1958). Givens rotations are localized in a two-dimensional subspace of predefined coordinates. Therefore, computations with Givens sequences scale with the number of factors and the computational cost for applying each factor can be kept low since efficient implementations are possible (Golub & Van Loan, 2012). Our main motivation comes from the success story of the Fast Fourier transform (FFT) (Cooley & Tukey, 1965), which brought down the computational cost of applying a Fourier transform to $\mathcal{O}(d \log(d))$ operations. This reduction led to a revolution in signal processing and was recognized by Sullivan & Dongarra (2000) as one of the most important algorithms of the 20th century. However, this speed-up relies on the fact that the classical Fourier transform is defined over a periodic grid, which provides many symmetries leveraged in the butterfly structure of the FFT.

These symmetries do not carry over to unstructured domains such as graphs and general unitary operators. In fact, using simple covering bounds, we show that generic unitary matrices require $\mathcal{O}(d^2 / \log d)$ Givens factors to be effectively approximated. However, the question of approximating with fewer factors in the presence of structure remains open: given an element $U \in U(d)$, how to produce the best possible N -term sequence of Givens rotations $G_1 \dots G_N$ that minimizes $\|U - \prod_j G_j\|$?

Due to the combinatorial nature of selecting Givens subspaces, this is an NP-hard optimization problem. In this paper, we propose a relaxation based on sparsity-inducing norms over the unitary group. In essence, given a point $U \in U(d)$, we use the gradient flow of a potential function $f : U(d) \rightarrow \mathbb{R}$ to define a path that links U to its nearest signed permutation matrix, the sparsest elements of the group and thus the global minimizers of f . Then, our algorithm tries to approximately follow this path using coordinate descent with the Givens factors acting as generators of the group.

We validate our algorithm on a family of structured orthogonal operators, constructed with a planted random sequence of K Givens factors and demonstrate that effective approximation is possible in the regime $K = \mathcal{O}(d \log d)$. Finally, we apply our algorithm to approximate a graph Fourier transform (GFT), the orthogonal matrix obtained when di-

¹Technical University of Munich ²New York University. Correspondence to: Thomas Frerix <thomas.frerix@tum.de>, Joan Bruna <bruna@cims.nyu.edu>.

agonalizing a graph Laplacian.

For ease of exposition, we restrict our discussion to approximating orthogonal group elements. However, this does not impose a restriction on the outlined approaches, as they equally apply to the complex unitary group as well as the real orthogonal group.

2. Related Work

Givens rotations were introduced by (Givens, 1958) to factorize the unitary matrix that transforms a square matrix into triangular form. The elementary operation of rotating in a two-dimensional subspace led to numerous successful applications in numerical linear algebra (Golub & Van Loan, 2012), in particular, for eigenvalue problems (Golub & Van Der Vorst, 2000). In this context, a Givens sequence factorizes a unitary basis transform, which is an operation of paramount importance to signal processing.

In contrast to signal processing on a Euclidean domain, recently there has been increased interest in signal processing on irregular domains such as graphs (Shuman et al., 2013; Bronstein et al., 2017). In this setting, Magoarou et al. (2018) considered a truncated version of the classical Jacobi algorithm (Jacobi, 1846) to approximate the orthogonal matrix that diagonalizes a graph Laplacian. Other notable strategies to efficiently approximate large matrices with presumed structure include multiresolution analysis (Kondor et al., 2014) and sparsity (Kyg & Sachdeva, 2016).

In quantum computation, approximate representation of unitary operators is a fundamental problem. Here, a unitary operation that performs a computation on a quantum state needs to be represented by or approximated with few elementary single- and two-qubit gates, ideally polynomial in the number of qubits. In the literature of quantum computing, a Givens rotation is commonly referred to as a two-level unitary matrix; a generic n -qubit unitary operator can be factorized in such two-level matrices with $\mathcal{O}(4^n)$ elementary quantum gates (Vartiainen et al., 2004).

An alternative viewpoint on Givens sequences was analyzed by Shalit & Chechik (2014). The authors considered manifold coordinate descent over the orthogonal group as sequentially applying Givens factors. Consequently, the minimizing sequence of this algorithm yields a Givens factorization of the initial orthogonal matrix.

In this work, we analyze information theoretic properties of approximating unitary matrices via Givens factorization. We then propose to minimize a sparsity-inducing objective via manifold coordinate descent in a regime where effective approximation is possible. Subsequently, we apply this approach to approximate the graph Fourier transform and demonstrate that the proposed method can find better

sequences compared to a truncated Jacobi algorithm. This allows to efficiently transform a graph signal into the graph’s approximate Fourier basis, an essential operation in graph signal processing.

3. Givens Factorization and Elimination

Givens matrices represent rotations in a two-dimensional subspace, while leaving all other dimensions invariant (Givens, 1958; Golub & Van Loan, 2012). Such a counter-clockwise rotation in the (i, j) -plane by an angle α can be written as applying $G^T(i, j, \alpha)$, where

$$G(i, j, \alpha) = \begin{bmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & \cos(\alpha) & \cdots & \sin(\alpha) & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & -\sin(\alpha) & \cdots & \cos(\alpha) & \cdots & 0 \\ \vdots & & \vdots & & \vdots & & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{bmatrix} \quad (1)$$

The trigonometric expressions appear in the i -th and j -th rows and columns. Any orthogonal matrix $U \in \mathbb{R}^{d \times d}$ that is a rotation, $U \in SO(d)$, can be decomposed into a product of at most $d(d-1)/2$ Givens rotations. In general, there exist many possible factorizations. If $U \in O(d) \setminus SO(d)$, then it cannot be represented directly by a sequence of Givens rotations. However, a factorization can be obtained up to permutation with a negative sign, e.g., by flipping two columns.

In numerical linear algebra, Givens factors are often used to selectively introduce zero matrix entries by controlling the rotation angle. This leads to a constructive factorization algorithm, which demonstrates a $d(d-1)/2$ -factorization. To this end, we start with the matrix $U \in SO(d)$ and introduce zeros on the lower diagonal column-wise from left to right and bottom to top within every column. This is achieved by choosing the rotation subspace (i, j) and a suitable rotation angle to zero-out the matrix element (i, j) . The elimination order is illustrated for $d = 4$ by

$$\begin{pmatrix} * & * & * & * \\ 3 & * & * & * \\ 2 & 5 & * & * \\ 1 & 4 & 6 & * \end{pmatrix} \quad (2)$$

After $N = d(d-1)/2$ steps, we have $G_N^T \dots G_1^T U = D$, where D is a diagonal matrix with $D_{kk} = -1$ for an even number of values and $D_{kk} = 1$ otherwise. This result can be reduced to the identity by selecting two subspaces with values $D_{ii} = D_{jj} = -1$ and applying a rotation by an angle $\alpha = \pi$. We refer to this algorithm by *structured elimination*.

Apart from this sign ambiguity, we consider factorizations in the broader sense up to signed permutation of the resulting matrix columns. To be explicit, the set of signed

permutation matrices is defined as $\mathcal{P}_d := \{P \in \mathbb{R}^{d \times d} | P_{ij} \in \{-1, 0, 1\}, \sum_i |P_{ij}| = 1 \forall j, \sum_j |P_{ij}| = 1 \forall i\}$. For a matrix $U \in O(d)$, to measure approximation quality, we denote an approximation by $\hat{U}$ and use a symmetrized Frobenius norm criterion up to a signed permutation matrix as follows:

$$\|U - \hat{U}\|_{F, \text{sym}} := \min_{P \in \mathcal{P}_d} \|U - \hat{U}P\|_F. \quad (3)$$

The range of (3) over the orthogonal group is $[0, \sqrt{2d}]$ as the maximum is obtained for the distance between Hadamard¹ matrices $H(d)$ and the identity with $\|H(d) - I\|_{F, \text{sym}} / \sqrt{d} \rightarrow \sqrt{2}$ as $d \rightarrow \infty$. Since $\|A\|_F^2 = \mathbb{E}_{x \sim \mathcal{N}(0, I)} [\|Ax\|_2^2]$, the criterion measures the average approximation quality over random Gaussian vectors when applying $\hat{U}$ instead of U . The motivation for this definition is twofold. First, this definition allows us to discuss Givens factorizations of orthogonal matrices with negative determinant and henceforth we consider factorization over the orthogonal group $O(d)$ rather than the special orthogonal group $SO(d)$. Second, it enlarges the class of possible factorization algorithms to those that cannot distinguish between signed permutation matrices. Observe that since the cost of multiplying by a signed permutation matrix is $\mathcal{O}(d)$ (Knuth, 1998), the computational efficiency arguments in this paper are not affected by the permutation equivalence class as we are discussing approximations in the regime of $\mathcal{O}(d \log(d))$ factors.

4. Information Theoretic Rate of Givens Representation

The elimination algorithm discussed in Section 3 guarantees to factorize any orthogonal matrix in at most $d(d-1)/2$ Givens factors, which corresponds to the dimension of the orthogonal group. Since each Givens factor is parametrized by a single angle, it immediately follows that exact Givens factorization for arbitrary elements $U \in O(d)$ necessarily requires $d(d-1)/2$ factors.

Hence, this leads to the question of approximate factorization: if one tolerates a certain error $\|U - \hat{U}\|_F \leq \epsilon$, is it possible to find approximations $\hat{U} = \prod_{n \leq N} G_n$ with $N = o(d^2)$, ideally with $N = \mathcal{O}(d \log d)$? A covering argument shows that generic orthogonal matrices in d dimensions require at least $\Theta(d^2 / \log(d))$ Givens factors to achieve an ϵ -approximate factorization. We denote by μ the uniform Haar measure on the unitary group, which we normalize for each d , $\mu(U(d)) = 1$. For notational simplicity, we carry out the proof for the operator 2-norm. An analogous argument holds by replacing the operator 2-norm with the Frobenius norm while re-scaling the error by $\sqrt{d}$.

¹A Hadamard matrix is an orthogonal matrix H whose entries satisfy $|H_{i,j}| = 1/\sqrt{d}$ for all i, j .

Lemma 1. Let $\prod_{n \leq N} G_n$ be a product of Givens factors with rotation angles α_n and $\bar{G}_n$ be the respective perturbed factors with rotation angles $\alpha_n + \delta_n$ and perturbations $0 \leq \delta_n \leq \delta$. Then,

$$\left\| \prod_{n \leq N} \bar{G}_n - \prod_{n \leq N} G_n \right\|_F \leq 2N\delta. \quad (4)$$

Proof. For any orthogonal matrices U, U', V, V' , we have

$$\begin{aligned} \|U'V' - UV\|_F &= \|(U + U' - U)V' - UV\|_F \\ &\leq \|U(V' - V)\|_F + \|(U' - U)V'\|_F \\ &= \|V' - V\|_F + \|U' - U\|_F, \end{aligned} \quad (5)$$

by using the fact that the Frobenius norm is invariant to orthogonal matrix multiplication. By iterating this relation, we obtain

$$\left\| \prod_{n \leq N} \bar{G}_n - \prod_{n \leq N} G_n \right\|_F \leq \sum_{n \leq N} \|\bar{G}_n - G_n\|_F. \quad (6)$$

Since $\bar{G}_n$ and G_n rotate in the same subspace,

$$\|\bar{G}_n - G_n\|_F = 2\sqrt{1 - \cos(\delta_n)}. \quad (7)$$

Inequality (4) follows from $\sqrt{1 - \cos(\delta_n)} \leq \delta_n \leq \delta$. $\square$

Theorem 1. Let $\epsilon > 0$. If $N = o(d^2 / \log(d))$, then as $d \rightarrow \infty$,

$$\mu \left(\left\{ U \in U(d) \mid \inf_{G_1 \dots G_N} \|U - \prod_{n \leq N} G_n\|_2 \leq \epsilon \right\} \right) \rightarrow 0.$$

Proof. Consider an ϵ -covering of the unitary group, i.e., a discrete set $\mathcal{X}$ such that $\inf_{X \in \mathcal{X}} \|U - X\|_2 \leq \epsilon$ for all $U \in U(d)$. Since the manifold dimension of the unitary group is $d(d-1)/2$, we need $|\mathcal{X}| = \Theta(\epsilon^{-d(d-1)/2})$ many balls for that cover. Let $N := N(d)$ be the number of available Givens factors for approximation at dimension d , and $\mathcal{A}_N = \{X \in U(d) \mid \inf_{G_1 \dots G_N} \|X - \prod_{n \leq N} G_n\|_2 \leq \epsilon/2\}$ denote the set of unitary operators which can be effectively approximated with N Givens terms. Now, suppose that $\mu(\mathcal{A}_N) \geq c > 0$, i.e., the set of group elements admitting an $\epsilon/2$ -approximation has positive measure. This implies that any ϵ -cover of $\mathcal{A}_N$ must be of size $\Theta(\epsilon^{-d(d-1)/2})$. Let us build such an ϵ -cover.

If we discretize the rotation angle to a value $\delta > 0$, then there are $(d(d-1)/2\delta)$ many different quantized Givens factors, denoted by $\bar{G}_i$, and consequently $(d(d-1)/2\delta)^N$ many different sequences. It follows that if $\delta := \frac{\epsilon}{4N}$, the discrete set $\mathcal{Y} = \{\prod_{n \leq N} \bar{G}_{i_n}\}$ containing all possible sequences of length N of quantized Givens rotations is an

ϵ -cover of $\mathcal{A}_N$. Indeed, by using Lemma 1 and the fact that the operator 2-norm is bounded by the Frobenius norm, we have $\forall X \in \mathcal{A}_N$,

$$\|X - \prod_{n \leq N} \tilde{G}_n\|_2 \leq \|X - \prod_{n \leq N} G_n\|_2 + 2N\delta \leq \frac{\epsilon}{2} + \frac{\epsilon}{2} = \epsilon.$$

Since $|\mathcal{Y}| = \left(\frac{2d(d-1)N}{\epsilon}\right)^N$, it follows that

$$\left(\frac{2d(d-1)N}{\epsilon}\right)^N = \Theta(\epsilon^{-d(d-1)/2}),$$

which implies $N = \mathcal{O}(d^2/\log d)$. $\square$

An immediate consequence of Theorem 1 is that generic effective approximation, i.e., with a number of factors $N = \mathcal{O}(d \log d)$, is information theoretically impossible. However, the situation may be entirely different for structured distributions of unitary operators. For that purpose, we develop an algorithm to obtain effective approximations based on sparsity-inducing norms.

5. Givens Factorization and Coordinate Descent on $O(d)$

In this section, we offer an alternative viewpoint presented by Shalit & Chechik (2014) that interprets Givens factorization as manifold coordinate descent on the orthogonal group over a certain potential energy.

The orthogonal group $O(d)$ is a matrix Lie group with associated Lie algebra $\mathfrak{o}(d) = \text{Skew}(d) = \{X \in \mathbb{R}^{d \times d} | X = -X^T\}$, the set of $d \times d$ skew-symmetric matrices (Hall, 2003). The tangent space at an element U is $T_U O(d) = \{XU | X \in \text{Skew}(d)\}$ and the Riemannian directional derivative of a differentiable function f in the direction $XU \in T_U O(d)$ is given by

$$D_X f(U) = \left. \frac{d}{d\alpha} f(\text{Exp}(\alpha X)U) \right|_{\alpha=0}, \quad (8)$$

where $\text{Exp} : \mathfrak{o}(d) \rightarrow O(d)$ is the matrix exponential. If we choose the basis $\{X_{ij} = e_i e_j^T - e_j e_i^T | 1 \leq i < j \leq d\}$ for the tangent space, then $D_{X_{ij}} f(U)$ represents the directional derivative in such a coordinate direction. A coordinate descent algorithm uses a criterion to choose coordinates (i, j) and a step size (rotation angle) α to iteratively update

$$U^{k+1} = \text{Exp}(-\alpha X_{ij})U^k. \quad (9)$$

A greedy criterion determines the best descent on f by a search over all possible coordinate directions $\{X_{ij}\}_{i < j \leq d}$ with the optimal step size obtained by a line search.

A Givens factor can be interpreted as a coordinate descent step over the orthogonal group. This follows from the relation

$$\text{Exp}(-\alpha X_{ij}) = G^T(i, j, \alpha). \quad (10)$$

In $d = 3$, an explicit example of the correspondence between Lie algebra and Lie group elements is

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -\alpha \\ 0 & \alpha & 0 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{pmatrix}. \quad (11)$$

Suppose we want to minimize a function f over the orthogonal group,

$$\min_{U \in O(d)} f(U). \quad (12)$$

Then minimizing (12) with manifold coordinate descent iterations (9) yields a Givens factorization of the initial point U^0 . A truncated sequence leads to an approximate factorization. From this viewpoint, the quality of a Givens factorization can be controlled by properties of the function f . In the following, we construct an objective function that results in approximate factorization with less than $\mathcal{O}(d^2)$ factors.

6. Sparsity-Inducing Dynamics

To factorize a matrix $U \in O(d)$ one may choose it as an initial value to problem (12) when minimizing a suitable potential function f with manifold coordinate descent. We want to find a factorization up to signed permutation of the matrix columns. As the signed permutation matrices are the sparsest orthogonal matrices, we consider an energy function that quickly enforces sparsity, the element-wise L_1 -norm of a matrix,

$$f(U) := d^{-1} \|U\|_1 = d^{-1} \sum_{i,j=1}^d |U_{ij}|. \quad (13)$$

Although f is convex in $\mathbb{R}^{d^2}$ (since it is a norm), due to the non-convexity of the domain, the problem $\min_{U \in O(d)} f(U)$ is non-convex. The landscape of f characterizes the class of orthogonal matrices that admit effective Givens approximation. It is easy to see that the global minima of f in $O(d)$ consist of signed permutation matrices, with $\min f(U) = 1$, and the global maxima are located at Hadamard matrices, with $\max f(U) = \sqrt{d}$. A more involved question concerning the presence or absence of spurious local minima of f is of interest. The following proposition partially addresses this question by showing that critical points of f are necessarily located at $U \in O(d)$ with some of its entries set to zero.

Proposition 1. Let $x \in \mathbb{R}^{2 \times d}$ and let

$$R(\alpha) := \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \quad (14)$$

be a counter-clockwise rotation in the plane by an angle α . Consider the function $g(\alpha) := \|R(\alpha)x\|_1$. Then, at every local minimum α^* of g there exist indices k, l such that $(R(\alpha^*)x)_{kl} = 0$.

Proof. We show equivalently that any stationary point α^* with $(R(\alpha^*)x)_{kl} \neq 0 \ \forall k, l$ is a local maximum. At any such point the function g is twice continuously differentiable and the second derivative is

$$\left. \frac{\partial^2 g}{\partial \alpha^2} \right|_{\alpha=\alpha^*} = -g(\alpha^*) < 0. \quad (15)$$

Consequently, any stationary point under this assumption must be a local maximum. $\square$

Proposition 1 implies that for a given subspace (i, j) , the best rotation angle can be found by checking all axis transitions for the 2D points $(u_{ik}, u_{jk}), k \in \{1, \dots, d\}$ and selecting the angle that most minimizes the objective among them. It also implies that any local minimum of f must correspond to an orthogonal matrix with at least d zeros placed at specific entries, such that no two rows or columns have the same support. Indeed, Proposition 1 implies that there exists a continuous path $t \mapsto U(t) = G(i, j, \alpha(t))$ with $\alpha(0) = 0$, generated by a Givens rotation of angle $\alpha(t)$, such that $f(U(t))$ is non-increasing at $t = 0$, provided one can find two rows or columns of U with the same support. However, this result does not exclude the possibility that f has spurious local minima at matrices U with the above special sparsity pattern. In fact, we conjecture that the landscape of f does have spurious local minima.

A manifold coordinate descent on the objective function f is explicitly stated in Algorithm 1. The crucial step involves optimizing this objective in the rotation angle α for a given subspace (i, j) , which is a non-convex optimization problem. Nevertheless, the global optimum can be found as stated by Proposition 1. In d dimensions, this step requires d operations. Consequently, due to the squared dimension dependence of the double for-loop, a naive implementation of Algorithm 1 would require $\mathcal{O}(d^3)$ operations. However, applying the selected Givens factor in each step changes only two rows of the matrix; thus, in the subsequent iteration, only those pairs of rows that involve the previously modified ones need to be re-computed. These are $\mathcal{O}(d)$ rows and altogether the runtime of an iteration is $\mathcal{O}(d^2)$.

Algorithm 1 Coordinate descent on the L_1 -criterion

Input: initial value $U^0 \in O(d)$, $f(U) = \|U\|_1$
repeat
 for $i = 1$ **to** d **do**
 for $j = 1$ **to** d **do**
 if α_{ij}^* not up-to-date **then**
 $\alpha_{ij}^* = \operatorname{argmin}_{\alpha} f(G^T(i, j, \alpha)U^k)$
 end if
 end for
 end for
 $i^*, j^* = \operatorname{argmin}_{i,j} f(G^T(i, j, \alpha_{ij}^*)U^k)$
 $U^{k+1} = G^T(i^*, j^*, \alpha_{i^*j^*}^*)U^k$
until $\|U^{k+1} - I\|_{F, \text{sym}} < \varepsilon$ or maxIter is reached

7. Numerical Experiments

7.1. Planted Models

Theorem 1 shows that we cannot expect to find good approximations to Haar-sampled matrices with less than $\mathcal{O}(d^2/\log(d))$ Givens factors. Therefore, we focus on a distribution for which we can control approximability. We use the uniform distribution over the set $\{U \in SO(d) | U = G_1 \cdots G_K, G_k = G(i_k, j_k, \alpha_k)\}$, where each G_k is obtained by first sampling a subspace uniformly at random (with replacement), and then sampling the corresponding angle uniformly from $(0, 2\pi)$. We denote the resulting distribution by the K -planted distribution ν_K . While this distribution may be sparse in the number of Givens factors for $K \ll d(d-1)/2$, this does not imply that the resulting matrices are sparse. In fact, products of Givens matrices become dense quickly. It follows from the Coupon Collector's Lemma that matrices generated with $\Theta(d \log_2(d))$ Givens factors are already dense with high probability. To visualize this effect, Figure 1 shows the L_0 -norm as a function of planted Givens factors.

We compare the following factorization algorithms. A *greedy baseline* iteratively finds the Givens factor that most minimizes the objective (3). The *structured elimination* algorithm described in Section 3 yields a sequence of Givens factors that eliminate matrix entries in the order (2) and is guaranteed to find a perfect factorization with $d(d-1)/2$ factors. Our sparsity-inducing algorithm minimizes the L_1 -criterion (13) via a *manifold coordinate descent* scheme.²

In an initial experiment, we demonstrate the approximation effectiveness of these algorithms; the results are shown in Figure 2. They indicate that minimizing the L_1 -criterion improves over directly minimizing the Frobenius norm (greedy

²An implementation of these algorithms can be found at <https://github.com/tfrerix/givens-factorization>

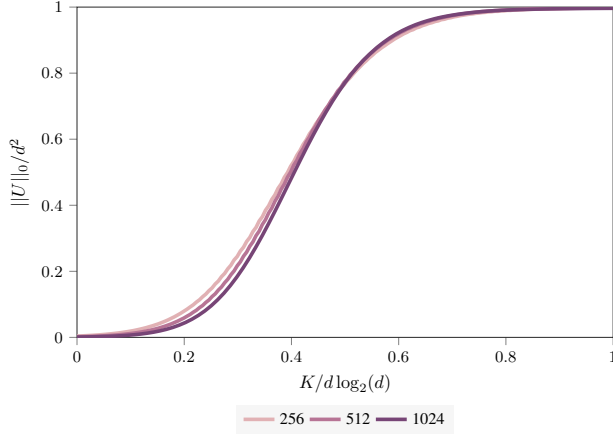


Figure 1. Average sparsity based on 100 samples of matrices drawn from the K -planted distribution over $SO(d)$ for increasing K . Standard deviation is negligible and not shown. Matrices become dense quickly as the number of planted Givens factors grows. In particular, matrices sampled from the $d \log_2(d)$ -planted distribution are already dense.

baseline). Next, we analyze the approximability of samples drawn from the K -planted distribution ν_K as a function of K . To obtain a Givens sequence, we factorize these samples with manifold coordinate descent on the L_1 -objective (13). Along the optimization path, we define $N_\epsilon(U)$ as the number of Givens factors for which the normalized approximation error (3) is smaller than $\epsilon = 0.1$, i.e.,

$$N_\epsilon(U) := \min \left\{ N \left| \frac{\|U - G_1 \dots G_N\|_{F, \text{sym}}}{\sqrt{d}} < \epsilon \right. \right\} \quad (16)$$

We refer to a Givens sequence with such $N_\epsilon(U)$ factors as an ϵ -factorizing sequence of U . In Figure 3, the sample average $N_\epsilon = n^{-1} \sum_{i=1}^n N_\epsilon(U_i)$ for $n = 10$ samples is shown as a function of K . We are interested in the rate at which N_ϵ grows for increasing K . The data in Figure 3 show that for $K = \alpha d \log_2(d)$ and $N_\epsilon = \beta d \log_2(d)$, the ratio β/α is not independent of d . For the shown dimension regime this implies that for $K = \mathcal{O}(d \log(d))$, N_ϵ grows polynomial in d , albeit with small rate for few planted factors. To make this relation more precise, we extract the exponent η of a model $N_\epsilon \sim d^\eta$. Figure 4 shows that the growth is slightly superlinear in the few-factor regime and becomes quadratic towards $K = d \log_2(d)$. Analytically characterizing such growth is left for future work.

That said, our initial results suggest the existence of a computational-to-statistical gap for the recovery (or detection) of sparse planted Givens factors. Indeed, Theorem 1 proves that recovery with $K = \mathcal{O}(d^2/\log d)$ planted factors is information-theoretically possible, whereas our greedy recovery strategy is only effective for

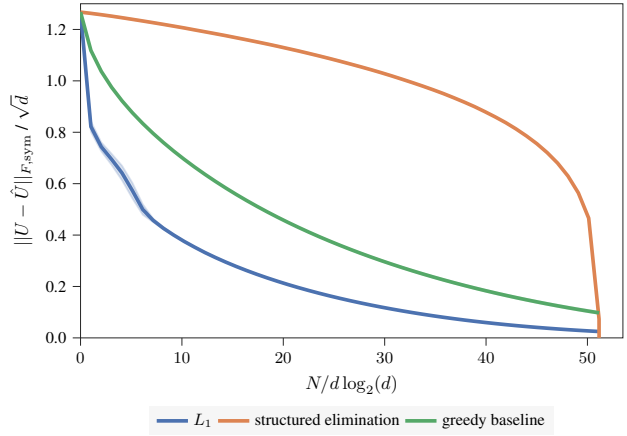


Figure 2. Average Frobenius norm approximation error in $d = 1024$ dimensions when factorizing 10 samples drawn from the $d \log_2(d)$ -planted distribution over $SO(d)$ with $d(d-1)/2$ factors. Shaded area denotes standard deviation.

$K = \mathcal{O}(d \log d)$. The mathematical analysis of our coordinate descent algorithm in the regime where effective approximation is feasible is beyond the scope of the present paper. In particular, proving that $N_\epsilon = \mathcal{O}(d \log d)$ is sufficient when $K \lesssim d \log d$ remains an open question.

7.2. Application: Graph Fourier Transform

The method introduced in this paper is useful in situations where one at first computes an approximation to a unitary operator, which is subsequently applied many times. Hence, the trade-off between initial computation and approximation on the one hand and efficient application on the other hand is in favor of the latter. Canonical examples for this scenario are orthogonal basis transforms. In this paper, we draw motivation from the FFT, which yields a speed-up of applying a Fourier transformation over a regular grid domain from $\mathcal{O}(d^2)$ to $\mathcal{O}(d \log(d))$ time complexity (Cooley & Tukey, 1965). However, these speed-ups do not carry over when the domain is unstructured, such as general graphs. Here, we compute an effective approximation of the graph Fourier transformation (GFT). Consider a simple, undirected graph with degree matrix D and adjacency matrix A . The unnormalized graph Laplacian is defined as $L := D - A$, which is a positive semi-definite, symmetric matrix. The GFT is represented by the orthogonal matrix that diagonalizes L .

A baseline for our method is the Jacobi algorithm (Jacobi, 1846), which diagonalizes a symmetric matrix L by greedily minimizing the off-diagonal squared Frobenius norm,

$$\text{off}(L) := \|L\|_F^2 - \sum_{k=1}^d L_{kk}^2. \quad (17)$$

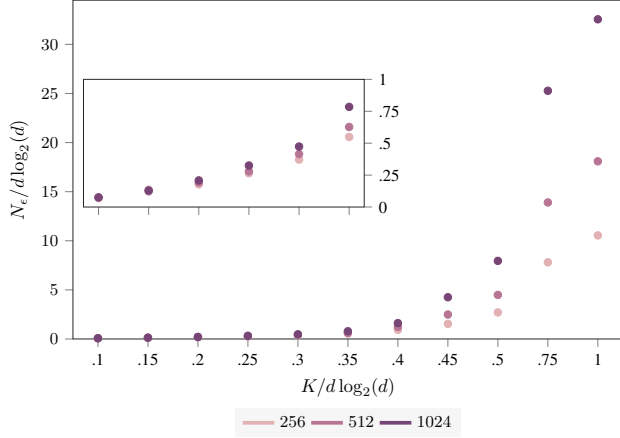


Figure 3. Average number of Givens factors necessary to factorize a K -planted matrix in $d \in \{256, 512, 1024\}$ dimensions up to desired accuracy as a function of K . Here, $\epsilon = 0.1$ is the accuracy as defined in expression (16). Note that the x-axis is shown with unequal spacing to highlight the relevant regime of the data. The inset plot shows a zoom of the first data points.

This is achieved by zeroing-out the largest matrix element in absolute value at every iteration. To this end, a Givens matrix similarity transformation with a suitably chosen rotation subspace and rotation angle is applied. However, the Jacobi algorithm does not guarantee factorization in a finite number of steps; in particular, it may take more than $N = d(d-1)/2$ iterations. In fact, the algorithm converges linearly (Golub & Van Loan, 2012),

$$\text{off}(L^{k+1}) \leq \left(1 - \frac{1}{N}\right) \text{off}(L^k). \quad (18)$$

If the iteration number k is large enough, quadratic convergence was shown by Schönhage (1964). Hence, the method is ineffective for small iteration numbers and in high dimensions. A truncated version of this algorithm was used by Magoarou et al. (2018) to obtain an approximation to the GFT. The objective (17) of the Jacobi method is motivated by approximating the spectrum of the symmetric matrix through the Gershgorin circle theorem (Gershgorin, 1931). However, we argue here that a criterion focused on approximating the eigenbasis of the symmetric matrix directly yields a more effective approximation to this orthogonal basis transformation. We consider the eigendecomposition $L = U\Lambda U^T$ and compute an approximation of the orthogonal matrix U with the algorithms outlined in Section 7.1. We demonstrate this procedure on Barabási-Albert random graphs and several real world graphs.

The Barabási-Albert model starts with n_0 unconnected vertices and iteratively adds vertices to the graph, which are connected to a number m of already existing ones with a probability proportional to the degree of these vertices.

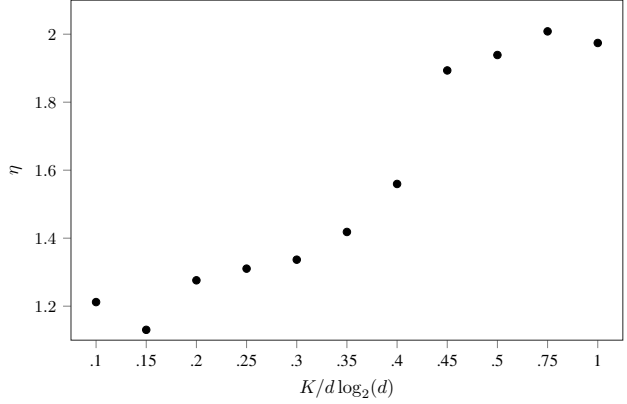


Figure 4. Polynomial growth rate η of the model $N_\epsilon \sim d^\eta$ as a function of the number of planted factors estimated from $d \in \{256, 512, 1024\}$. Note that the x-axis is shown with unequal spacing to highlight the relevant regime of the data.

Table 1. Construction of Barabási-Albert graphs. An n -vertex graph is constructed by choosing $n_0 = m_k$ initial vertices, then adding vertices and connecting them to m_k of already existing ones with a probability proportional to the degree of these vertices. m_k is chosen such that the number of resulting edges is approximately $k \cdot 0.25n(n-1)/2$.

n	64	128	256	512	1024
m_1	54	109	218	437	874
m_2	36	69	136	267	528

This construction is known as preferential attachment and induces a scale-free degree distribution found in real world graphs (Barabási & Albert, 1999). The details of generating these graphs are described in Table 1. We approximate the corresponding graph Laplacians with $n \log_2(n)$ factors leading to the results shown in Figure 5. While our sparsity-inducing algorithm yields better factorizations in most cases, there exist scenarios, where the greedy baseline results in better approximations ($d \in \{512, 1024\}$ for $\sim 0.25n \log_2(n)$ edges). Finally, we demonstrate approximate factorization of the graph Laplacian of various real world graphs listed in Table 2. Our L_1 -algorithm yields the best factorization for the Minnesota, HumanProtein, and Email graphs, while the greedy baseline algorithm is superior for the Facebook graph.

A simple strategy to improve the performance of our L_1 greedy method with mild computational overhead is to perform beam-search, which is beyond the scope of this paper. Overall, it remains an open question to more closely characterize the graphs for which our sparsity-inducing algorithm yields effective approximations of the GFT.

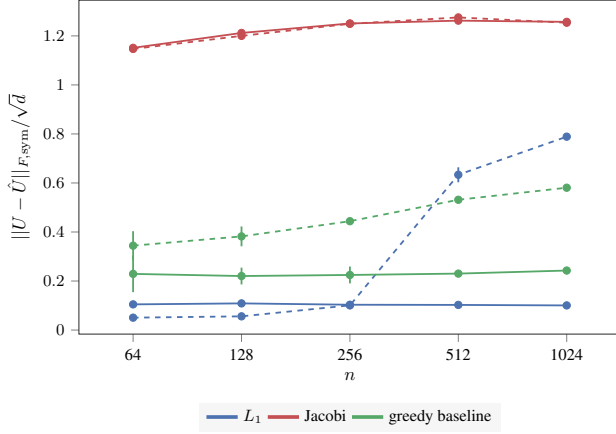


Figure 5. Approximate factorization of the graph Laplacian of n -vertex Barabási-Albert graphs with $n \log_2(n)$ factors. Data points are averages of 10 samples, vertical lines denote standard deviation. The solid (—) lines show factorizations of graphs with $\sim 0.5n(n-1)/2$ edges, while the dashed (---) lines show factorizations of graphs with $\sim 0.25n(n-1)/2$ edges.

Table 2. GFT approximation for real world graphs with n vertices and n_e edges.

	n	n_e
MINNESOTA (Defferrard et al.)	2642	3304
HUMANPROTEIN (Rual et al., 2005)	3133	6726
EMAIL (Guimerà et al., 2003)	1133	5451
FACEBOOK (McAuley & Leskovec, 2012)	2888	2981

8. Discussion

We analyzed the problem of approximating orthogonal matrices with few Givens factors. While a perfect factorization in $\mathcal{O}(d^2)$ is always possible, an approximation with fewer factors is advantageous if the orthogonal matrix is applied many times. We showed that effective Givens factorization of generic orthogonal matrices is impossible and inspected a distribution of planted factors, which allows us to control approximability. Our initial results suggest that sparsity inducing factorization is promising beyond the sparse matrix regime. However, it remains an open problem to further characterize the matrices that admit effective factorization using manifold coordinate descent on an L_1 -criterion.

This work opens up questions we believe are important both from a theoretical and an applied perspective. On the theory side, important problems arising from our analysis are: (i)

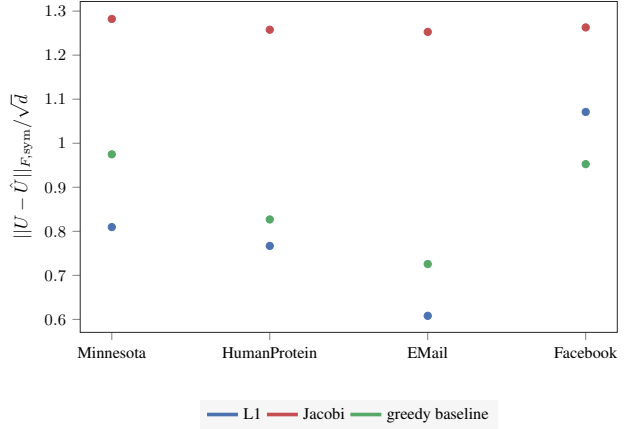


Figure 6. Approximate factorization of the graph Laplacian of various n -vertex real world graphs with $n \log_2(n)$ factors.

a complete description of the landscape of $f(U) = \|U\|_1$ over the orthogonal and unitary groups, (ii) a precise classification of the detection threshold $K(d)$ below which it is possible to discriminate a K -planted sample from a Haar sample in polynomial time, and (iii) a guarantee that the proposed sparse Givens coordinate descent algorithm requires $N = \Theta(d \log d)$ terms for $K \leq Cd \log d$ for some constant $C > 0$. These questions suggest a learning approach whereby our sparsity promoting potential f would be replaced by a classifier f_θ trained to discriminate between K -planted and Haar distributions. From an applied perspective, the method allows to approximately invert a time-varying symmetric linear operator $H(t)$. Similar to the Woodbury formula for low-rank updates of an inverse, one could set up an approximate Givens factorization of the eigenbasis of $H(t_0)$, and update it efficiently at subsequent times. If successful, this could dramatically improve the efficiency of second-order optimization schemes, where $H(t)$ is the Hessian of a loss function.

Acknowledgements

The authors would like to thank Oded Regev for in-depth discussions and early feedback, as well as Kyle Cranmer and Lenka Zbedorova for fruitful discussions on the topic, and Yann LeCun for introducing us to the problem. This work was partially supported by NSF grant RI-IIS 1816753, NSF CAREER CIF 1845360, the Alfred P. Sloan Fellowship and Samsung Electronics.

References

- Barabási, A.-L. and Albert, R. Emergence of Scaling in Random Networks. *Science*, 286(5439):509–512, 1999.
- Bronstein, M. M., Bruna, J., LeCun, Y., Szlam, A., and Vandergheynst, P. Geometric Deep Learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.
- Cooley, J. W. and Tukey, J. W. An algorithm for the machine calculation of complex Fourier series. *Math. Comp.*, 19: 297–301, 1965.
- Defferrard, M., Martin, L., Pena, R., and Perraudin, N. Pygsp: Graph signal processing in python. description: <https://www.cise.ufl.edu/research/sparse/matrices/Gleich/minnesota.html>.
- Gershgorin, S. A. Über die Abgrenzung der Eigenwerte einer Matrix. *Izv. Akad. Nauk. USSR Otd. Fiz.-Mat. Nauk*, (6):749–754, 1931.
- Givens, W. Computation of Plain Unitary Rotations Transforming a General Matrix to Triangular Form. *Journal of the Society for Industrial and Applied Mathematics*, 6(1): 26–50, 1958.
- Golub, G. H. and Van Der Vorst, H. A. Eigenvalue Computation in the 20th Century. *Journal of Computational and Applied Mathematics*, 123(1-2):35–65, 2000.
- Golub, G. H. and Van Loan, C. F. *Matrix computations*. JHU Press, 4th edition, 2012.
- Guimerà, R., Danon, L., Díaz-Guilera, A., Giralt, F., and Arenas, A. Self-similar Community Structure in a Network of Human Interactions. *Phys. Rev. E*, 68(6):065103, 2003. retrieved from <http://konect.uni-koblenz.de/networks/arenas-email>.
- Hall, B. *Lie Groups, Lie Algebras, and Representations*. Graduate Texts in Mathematics. Springer, 2003.
- Jacobi, C. Über ein leichtes Verfahren die in der Theorie der Säcularstörungen vorkommenden Gleichungen numerisch aufzulösen. *Journal für die reine und angewandte Mathematik*, 30:51–94, 1846.
- Knuth, D. E. *The Art of Computer Programming, Volume 3: Sorting and Searching*. 2nd edition, 1998. Ex 5.2-10, p.80.
- Kondor, R., Teneva, N., and Garg, V. Multiresolution matrix factorization. In *International Conference on Machine Learning (ICML 2014)*, pp. 1620–1628, 2014.
- Kyng, R. and Sachdeva, S. Approximate Gaussian Elimination for Laplacians: Fast, Sparse, and Simple. In *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*, pp. 573–582, 2016.
- Magoarou, L. L., Gribonval, R., and Tremblay, N. Approximate Fast Graph Fourier Transforms via Multilayer Sparse Approximations. *IEEE Transactions on Signal and Information Processing over Networks*, 4(2):407–420, 2018.
- McAuley, J. and Leskovec, J. Learning to Discover Social Circles in Ego Networks. In *Advances in Neural Information Processing Systems (NIPS 2012)*, pp. 548–556. 2012. retrieved from <http://konect.uni-koblenz.de/networks/ego-facebook>.
- Rual, J.-F., Venkatesan, K., Hao, T., Hirozane-Kishikawa, T., Dricot, A., Li, N., Berriz, G. F., Gibbons, F. D., Dreze, M., and Ayivi-Guedehoussou, N. Towards a Proteome-scale Map of the Human Protein–Protein Interaction Network. *Nature*, (7062):1173–1178, 2005. retrieved from <http://konect.uni-koblenz.de/networks/maayan-vidal>.
- Schönhage, A. Zur quadratischen Konvergenz des Jacobi-Verfahrens. *Numerische Mathematik*, 6(1):410–412, 1964.
- Shalit, U. and Chechik, G. Coordinate-Descent for Learning Orthogonal Matrices through Givens Rotations. In *Proceedings of the 31st International Conference on Machine Learning (ICML 2014)*, 2014.
- Shuman, D. I., Narang, S. K., Frossard, P., Ortega, A., and Vandergheynst, P. The Emerging Field of Signal Processing on Graphs: Extending High-Dimensional Data Analysis to Networks and other Irregular Domains. *IEEE Signal Processing Magazine*, 30(3):83–98, 2013.
- Sullivan, F. and Dongarra, J. Guest Editors’ Introduction: The Top 10 Algorithms. *Computing in Science and Engineering*, 2:22–23, 2000.
- Vartiainen, J. J., Möttönen, M., and Salomaa, M. M. Efficient Decomposition of Quantum Gates. *Phys. Rev. Lett.*, 92:177902, 2004.