Directional Statistics on Permutations

Sergey M. Plis[†] **Stephen McCracken**[*] **Terran Lane**[*] **Vince D. Calhoun**[†]
[†]The Mind Research Network
s.m.plis@gmail.com
vcalhoun@mrn.org
[*]Computer Science Dept.
The University of New Mexico
{samcc,terran}@cs.unm.edu

Abstract

Distributions over permutations arise in applications ranging from multi-object tracking to ranking. The difficulty in dealing with these distributions is caused by the size of their domain, which is factorial in the number of entities ($n!$). The direct definition of a multinomial distribution over the permutation space is impractical for all but a very small n. In this work we propose an embedding of all $n!$ permutations for a given n in a surface of a hypersphere defined in $\mathbb{R}^{(n-1)^2}$. As a result, we acquire the ability to define continuous distributions over a hypersphere with all the benefits of directional statistics. We provide polynomial time projections between the continuous hypersphere representation and the $n!$-element permutation space. The framework provides a way to use continuous directional probability densities and the methods developed thereof for establishing densities over permutations. As a demonstration of the benefits of the framework we derive an inference procedure for a state-space model over permutations. We demonstrate the approach with applications and comparisons to existing models.

1 Introduction

Since the inception of the field of computer science, there has been a strong dichotomy between optimization in continuous spaces (such as $\mathbb{R}^d$) and combinatorial spaces (such as the space of permutations on n objects). While there are computationally hard problems in both kinds of spaces, combinatorial spaces are far more often the villain. Bayesian inference in the space of permutations, for example, is an important, yet frustratingly difficult problem (Kondor et al., 2007).

We feel that a key factor at the heart of this dichotomy is that combinatorial spaces are far more *unstructured* than the familiar continuous spaces. Unlike raw combinatorial sets, continuous spaces (e.g., Euclidean d-space) typically come equipped with a topology, continuity, compact subsets, a metric, an inner product, and so on (Munkres, 1975). On these are built the entire infrastructure of analysis, including the derivative (Kreyszig, 1978), and thence to most optimization techniques and representations such as the Fourier basis. Combinatorial spaces, on the other hand, have been burdened with fewer assumptions, but endowed with fewer advantages.

One strategy for working with combinatorial spaces is to embed them into continuous spaces, thus imposing a structure, and work there with powerful analytic tools. This trick has proven to be powerful in, for example, continuous relaxations of integer programming problems (Gomory, 1958; Jaakkola et al., 2010).

In this paper, we demonstrate the power of the embedding approach by developing a fast, accurate approach to Bayesian inference over permutations. Arising in tasks such as object tracking (Kondor et al., 2007) or ranking (Meila et al., 2007), this problem is challenging because of the factorially-large number of parameters in an exact representation of a general probability distribution. Although metric-based methods on permutations are well-known in machine learning (Meila et al., 2007; Fligner and Verducci, 1986), those approaches stop short of identifying and fully exploiting an explicit embedding into a continuous space.

Prior approaches have worked by approximating a general probability distribution with a restricted set of basis functions, performing inference in the Fourier domain, and using accompanying transformations to project back to permutation space (Kondor et al., 2007; Huang et al., 2009), or by defining a metric

Appearing in Proceedings of the 14^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2011, Fort Lauderdale, FL, USA. Volume 15 of JMLR: W&CP 15.

over discrete permutations (Meila et al., 2007) – an approach currently limited to ranking, or maintaining and updating an identity management matrix in the information form (Schumitsch et al., 2005).

The hierarchical structure of our main contributions:

- **Theoretical observations**: we demonstrate an embedding of the $n!$ permutation set onto the surface of a hypersphere $\mathbb{S}^d$ centered at the origin in $\mathbb{R}^{d+1}$ with $d = (n-1)^2 - 1$.
  - *Observations*: we propose a hypersphere embedding of permutations.
  - *Practical results*: we develop polynomial time transformations between the discrete $n!$ permutation space and its continuous hypersphere representation.
- **Practical use**: we demonstrate a bridge between directional statistics (Mardia and Jupp, 2000) and permutation sets that leads to efficient inference.
  - *Observations*: we propose the von Mises-Fisher density over permutations.
  - *Practical results*: we develop efficient inference over permutations in a state-space model.
    - We employ analytical product and marginalization operations.
    - We show efficient transformation of partially observed permutations onto the surface of the hypersphere $\mathbb{S}^d$.

2 Embedding permutations

We will use several representations of permutations, including the $n \times n$ permutation matrix representation $\mathbf{P}$, which is a square bistochastic matrix[1] with entries $\mathbf{P}_{(i,j)} \in \{0, 1\}$. However, in contrast to the usual intent behind this representation, we do not use it as a permutation operator anywhere in the paper. Instead, it serves as an easy-to-interpret guide and a convenient way to establish some required properties. Therefore, in the rest of the paper we will interpret it merely as a vector in $\mathbb{R}^{n^2}$. To avoid notation clutter we treat all the matrices further in the paper as vectors in $\mathbb{R}^{n^2}$, omitting the special vector stacking operation symbols (such as $vec\,(\cdot)$) unless specified otherwise.

2.1 Representation

In this section we will show how a permutation set with $n!$ elements can be embedded onto the surface of a $(n-1)^2$ dimensional hypersphere.

[1] A tuple subscript $\mathbf{P}_{(\bullet,\bullet)}$ indicates an element of the matrix $\mathbf{P}$; a bare index $\mathbf{P}_\bullet$ indicates one of a set of matrices.

Our representation takes advantage of the geometry of the Birkhoff polytope and in part relies on the Birkhoff-von Neumann theorem (Bapat, 1997), which we state here without proof.

Theorem 1. *All $n \times n$ permutation matrices in $\mathbb{R}^{n^2}$ are extreme points of a convex $(n-1)^2$ dimensional polytope, which is the convex hull of all bistochastic matrices.*

Next, we formulate a lemma that the rest of the section is based on:

Lemma 1. *Extreme points of the Birkhoff polytope are located on the surface of a radius $\sqrt{n-1}$ hypersphere clustered around the center of mass of all $n!$ permutations.*

Proof. To show that the statement is valid we first compute the center of mass and then show that each permutation is located at an equal distance from this center. The center of mass for all the permutations on n objects is defined in $\mathbb{R}^{n^2}$ as $c_M = \frac{1}{n!}\sum_{k=1}^{n!} \mathbf{P}_k$.

We observe that the number of permutation matrices for which $\mathbf{P}_{(1,1)} = 1$ is $(n-1)!$, which follows from the effective removal of the first row and column of an $n \times n$ matrix caused by the assignment. Thus, $\sum \mathbf{P}_{(1,1)} = (n-1)!$ which, following the same reasoning, is true for any $\mathbf{P}_{(i,j)}$ and leads to

$$c_M = \frac{1}{n!}(n-1)!\mathbb{1} = \frac{1}{n}\mathbb{1} \tag{1}$$

Observing that $\|\mathbb{1} - \mathbf{P}\|_2 = \sqrt{n^2 - n}$ for any $\mathbf{P}$, we compute the radius of the sphere:

$$r_s = \left\| \frac{1}{n}\mathbb{1} - \mathbf{P} \right\|_2 = \sqrt{n-1} \tag{2}$$

to see that all permutations are equidistant from the center of mass c_M. □

To show that the hypersphere of Lemma 1 is embedded into a space of lower dimension than $\mathbb{R}^{n^2}$ we observe the following. With respect to the original formulation of permutations in $\mathbb{R}^{n^2}$, all of the permutations are located on the intersection of a hypersphere centered at the origin with $\sqrt{n}$ radius and a hypersphere of Lemma 1. This intersection is still a hypersphere only with dimension lowered by one. The following lemma provides a result needed to transform permutations into the lower dimensional space of Theorem 1.

Lemma 2. *The $(n-1)^2$-dimensional affine subspace of $\mathbb{R}^{n^2}$, that contains all permutations $\mathbf{P}_k$ as well as the sphere of Lemma 1, is formed by an intersection of $2n-1$ hyperplanes with highly structured and easily constructable normals.*

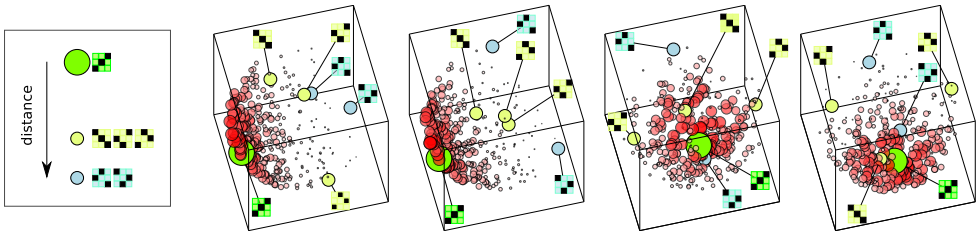


Figure 1: Axis-parallel projections of $\mathbb{S}^3$ with permutation points embedded into it. All 6 permutations are denoted by stencils of their matrix forms and colored circles. Details are covered in the text.

Proof. Let us denote by $\mathbf{W}_{(i,\mathbf{1})}$ an $n \times n$ matrix with all elements except a single i^{th} row of ones set to zero and likewise $\mathbf{W}_{(\mathbf{1},i)}$ for columns. Observe that:

$$vec\left(\mathbf{W}_{(i,\mathbf{1})}\right)^T vec\left(\mathbf{P}\right) = 1 \quad vec\left(\mathbf{W}_{(\mathbf{1},i)}\right)^T vec\left(\mathbf{P}\right) = 1$$

for any permutation matrix[2]. It follows, that all permutations are located at an intersection of $2n$ hyperplanes defined by their normals: $\mathbf{W}_{(\mathbf{1},i)}$ and $\mathbf{W}_{(i,\mathbf{1})}$, with $i \in \{1 \dots n\}$, and having bias of 1. This set is, however, not independent, because any $\mathbf{W}_{(i,\mathbf{1})}$ can be expressed by a linear combination of the other $2n-1$ vectors by setting weights of $\mathbf{W}_{(j\neq i,\mathbf{1})}$ to -1 and weights of $\mathbf{W}_{(\mathbf{1},i)}$ to 1 for $i, j \in \{1 \dots n\}$. This leads to $2n-1$ hyperplanes whose intersection forms the space in which the hypersphere containing the Birkhoff-polytope is located. Thus, the dimension of the transformed space is $n^2 - 2n + 1 = (n-1)^2$. □

All permutation matrices on n objects belong to the surface of a radius $\sqrt{n-1}$ hypersphere, $\mathbb{S}^d$, in $\mathbb{R}^{(n-1)^2}$ as established by Lemmas 1 and 2. Properties of rotation (Meyer, 2000) provide for equal transformation of all of the points on the sphere (not just permutations). We do not rigorously show here, but assume that by inherent symmetry in the structure of permutation matrices they are distributed evenly across the surface of $\mathbb{S}^d$.

Unfortunately the first interesting permutation (3 objects) already lives in a difficult-to-display 4-dimensional space. Nevertheless, it is instructive to see how the points that denote permutations are organized on the surface of $\mathbb{S}^d$. Figure 1 provides axis-parallel projections of $\mathbb{S}^3$ that contain all 6 permutations of 3 objects[3]. Permutations are identified by stencils of their matrix representations. An arbitrary reference permutation is shown as the largest circle, while the 5 other permutations are shown as smaller circles. Circles of the same color denote permutations equidistant from the reference (geodesic distance in the 4-dimensional space on $\mathbb{S}^3$). Additionally, to emphasize the surface of $\mathbb{S}^3$, a number of points were sampled around the reference and shown as unlabeled points with size and color intensity inversely proportional to their geodesic distance from the reference.

[2]In fact, for any bistochastic matrix, by Theorem 1

[3]Interactive version of the figure is available in the supplemental code

2.2 Transformations

The representation of the previous section allows us to define and manipulate probability density functions on $\mathbb{S}^d$ using approaches of continuous mathematics and only then transform quantities of interest back to the discrete $n!$ permutation space. This is useful when there is a way to efficiently transform elements of one space to the other. Next we show how this can be achieved in polynomial time.

The key components posing difficulties are discrete vs. continuous space, and the requirement of $\mathbb{S}^d$ to be origin-centered (required for Section 3). The former poses a considerably more challenging problem than the latter and absence of both would reduce the required transformations to a simple change of basis between $\mathbb{R}^{n^2}$ and $\mathbb{R}^{(n-1)^2}$. We develop the transformations in the proof to the following lemma.

Lemma 3. *There exist polynomial time transformations between the discrete $n!$ permutation space and the surface of the origin-centered $(n-1)^2$ dimensional hypersphere of radius $\sqrt{n-1}$.*

Proof. The transformation **from a permutation space to** $\mathbb{S}^d$ requires only a short sequence of linear operations as it is made clear by lemmas of Section 2.1:

1. Shift the permutation matrix $\mathbf{P}$ by $\frac{1}{n}\mathbb{1}$ to put the center of mass at the origin.
2. Change the basis by projecting into the $\mathbb{R}^{(n-1)^2}$ subspace orthogonal to $\mathbf{W}_{(\mathbf{1},i)}$ and $\mathbf{W}_{(i,\mathbf{1})}$.

Since there are $(n-1)^2$ basis vectors of length n^2, the projection operation takes $O(n^4)$. Note that the basis can be obtained by the QR factorization, which

is $O(n^6)$ in this case, but needs to be computed only once for a given n.

Transforming an arbitrary point **from $\mathbb{S}^d$ to the permutation space** is more challenging. Now we have to linearly transform the point from $\mathbb{S}^d$ to $\mathbb{R}^{n^2}$ and then among $n!$ possibilities find a permutation, that is the closest, in L_2 sense, to a given point. (The points closest in L_2 sense will also be the closest with respect to the geodesic distance on the hypersphere. This is true because a hypersphere is a closed convex manifold of a constant curvature.) The transformation is easily done by inverting the order of operations for going from $\mathbb{R}^{n^2}$ to $\mathbb{S}^d$, which amounts to $O(n^4)$ operations. Let us show how to efficiently find a permutation matrix closest to a transformed point.

Given an arbitrary point $\mathbf{T}^{\mathbb{S}}$ in $\mathbb{R}^{n^2}$, which corresponds to a point on $\mathbb{S}^d$, as indicated by the superscript, we introduce a matrix $\mathbf{D}$ where

$$\mathbf{D}_{(i,j)} = (\mathbf{T}^{\mathbb{S}}_{(i,j)} - 1)^2 \tag{3}$$

Finding the permutation $\mathbf{P}^{\mathbb{S}}$ closest to $\mathbf{T}^{\mathbb{S}}$ amounts to finding $\mathbf{P}^{\mathbb{S}}$ that minimizes $\sum_{ij} \mathbf{D}_{(i,j)}\mathbf{P}_{(i,j)}$. This is the same as matching every column and each row to a single counterpart so that the sum of matching weights (elements of $\mathbf{D}$) is minimal. In this case, $\mathbf{D}$ is an $n \times n$ edge-weight matrix for a $2n$ node bipartite graph with n elements per partition. This is the familiar minimum weighted bipartite matching problem (West, 2001). This observation allows us to apply a minimum weighted bipartite matching algorithm (West, 2001) and obtain a permutation $\mathbf{P}^{\mathbb{S}}$ closest to $\mathbf{T}^{\mathbb{S}}$. The result of the minimization is a permutation matrix, that automatically provides us with the closest permutation. The running time of the fastest general algorithms for solving this problem is $O(n^2 \log n + n^2 e)$, where e is the number of edges in the bipartite graph. Since the number of edges in our case can be n^2, the running time effectively becomes $O(n^4)$. □

2.3 Imposed structure

Although there may be many ways to embed permutations in a continuous space[4], only a few of them are useful. An acceptable embedding has the property that functions that are smooth[5] over the embedding domain also support a useful notion of smoothness in the permutation domain. In the Birkhoff polytope, two neighboring vertices differ by a single transposition (a relationship preserved by our embedding). Smooth functions over the hypersphere are smooth with respect to the transposition distance in the discrete permutation domain. Such smoothness is also at the core of the spectral approach to modeling densities over permutations; see (Kondor et al., 2007, Section 2.2) and (Huang et al., 2009, 2008).

Coupling the probability representations to the transformation operations bridges the gap between the discrete, combinatorial space of permutations and the continuous, low-dimensional hypersphere. This allows us to lift the large body of results developed for directional statistics (Mardia and Jupp, 2000) directly to permutation inference.

3 Directional statistics

A number of probability density functions on $\mathbb{S}^d$ have been developed in the field of directional statistics (Mardia and Jupp, 2000). A detailed account is given for the interested reader in (Mardia and Jupp, 2000, Chapter 9). The directional statistics framework allows us to define quite general classes of density functions over permutations. However, we choose to start with a distribution that directly supports neighbor distribution of permutations on the Birkhoff polytope. In the rest of the paper, we use one of the basic models to demonstrate the usefulness of our representation and the model as well.

3.1 von Mises-Fisher distribution

This is a m-variate von Mises-Fisher[6] (vMF) distribution of a m-dimensional vector $\boldsymbol{x}$, where $\|\boldsymbol{\mu}\| = 1$, $\kappa \geq 0$ and $m \geq 2$:

$$\text{vMF}(\boldsymbol{x}; \boldsymbol{\mu}, \kappa) = Z_m(\kappa)\, e^{\kappa \boldsymbol{\mu}^T \boldsymbol{x}} \tag{4}$$

$$Z_m(\kappa) = \frac{\kappa^{m/2-1}}{(2\pi)^{m/2} \boldsymbol{I}_{m/2-1}(\kappa)}, \tag{5}$$

where $\boldsymbol{I}_r(\cdot)$ is the r^{th} order modified Bessel function of the first kind and κ is called the concentration parameter. Examples of the distribution on $\mathbb{S}^2$ for several random values of κ and μ are shown in Figure 2.

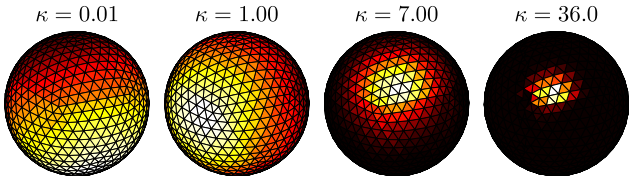


Figure 2: The von Mises-Fisher density function on $\mathbb{S}^2$

In terms of a pdf on permutations the vMF establishes a distance-based model, where distances are geodesic on $\mathbb{S}^d$. The advantage of the formulation in a continuous space is the ability to apply a range of operations

[4] For instance, order $n!$ permutations on the real line.

[5] Instrumental property for taking advantage of continuous methods.

[6] Sometimes also called the Langevin distribution.

on the pdf and still end up with the result on $\mathbb{S}^d$. This advantage is realized in the inference procedures which we establish next.

3.2 Efficient inference in a state space model

The results presented above establish a framework in which it is possible to define and manage, in reasonable time, probability densities over permutations. An important application of this framework is in probabilistic data association (PDA) (Rasmussen and Hager, 2001). In PDA we are interested in maintaining links between objects and tracks under noisy tracking conditions. Following Kondor et al. (2007), we ignore the underlying position estimation problem and focus instead on identity management, which boils down to tracking a hidden permutation (identity assignment) given noisy observed assignments.

We will perform identity management using a recursive Bayesian filtering approach that is analogous to the traditional multivariate Kalman filter, but uses vMF distributions in place of Gaussian distributions. We model uncertainty about hidden states $\boldsymbol{x}_t$ and observed states $\boldsymbol{y}_t$ using continuous probability distributions defined on the $\mathbb{S}^d$ embedding space. The model has two main parts:

1. A transition model $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$, which describes the stochastic evolution of the hidden permutation
2. An observation model $p(\boldsymbol{y}_t|\boldsymbol{x}_t)$, where $\boldsymbol{y}_t$ is the noisy observation of the hidden permutation

Upon receiving a new observation $\boldsymbol{y}_t$, we can compute the posterior $p(\boldsymbol{x}_t|\boldsymbol{y}_t)$ via a two-step update. First, use the transition model to estimate the next hidden state by marginalizing over the old hidden state:

$$p(\boldsymbol{x}_t|\boldsymbol{y}_{t-1}) = \int p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})p(\boldsymbol{x}_{t-1}|\boldsymbol{y}_{t-1})d\boldsymbol{x}_{t-1} \quad (6)$$

Second, use the new observation to update the estimate through the observation model:

$$p(\boldsymbol{x}_t|\boldsymbol{y}_t) \propto p(\boldsymbol{y}_t|\boldsymbol{x}_t)p(\boldsymbol{x}_t|\boldsymbol{y}_{t-1}) \quad (7)$$

Ideally, we would like to use probability distributions for the transition, observation, and posterior models that allow us to perform the update steps efficiently in closed form, as in the Kalman filter. vMF distributions come close to satisfying this criterion. The multiplication in (7) can be computed analytically; while the marginalization in (6) can be computed with reasonable accuracy and speed by approximating the vMFs with angular Gaussians, convolving analytically, and projecting back to vMF space (Mardia and Jupp, 2000). Furthermore, the inference steps operate only on $\mathbb{S}^d$ representations of permutations, avoiding unnecessary transformation overhead.

Therefore we adopt vMF distributions for the transition, observation, and posterior models, parametrized respectively as:

$$\begin{aligned} p(\mathbf{x}_t|\mathbf{x}_{t-1}) &:= \text{vMF}(\mathbf{x}_t; \mathbf{x}_{t-1}, \kappa_{trn}) \\ p(\mathbf{y}_t|\mathbf{x}_t) &:= \text{vMF}(\mathbf{y}_t; \mathbf{x}_t, \kappa_{obs}) \\ p(\mathbf{x}_t|\mathbf{y}_t) &:= \text{vMF}(\mathbf{x}_t; \boldsymbol{\mu}_t, \kappa_t) \end{aligned}$$

The update steps then proceed as follows. The marginalization step (6) produces a new vMF distribution $p(\mathbf{x}_t|\mathbf{y}_{t-1}) := \text{vMF}(\mathbf{x}_t; \boldsymbol{\mu}', \kappa')$, parametrized as:

$$\boldsymbol{\mu}' = \frac{\boldsymbol{x}_{t-1} + \boldsymbol{\mu}_{t-1}}{\|\boldsymbol{x}_{t-1} + \boldsymbol{\mu}_{t-1}\|} \quad (8)$$

$$\kappa' = A_d^{-1}(A_d(\kappa_{t-1})A_d(\kappa_{trn})) \quad (9)$$

$$A_d(\kappa) = \frac{\boldsymbol{I}_{d/2}(\kappa)}{\boldsymbol{I}_{d/2-1}(\kappa)} \quad (10)$$

where the ratio of modified Bessel functions in $A_d(\kappa)$ can be computed accurately and efficiently using the Lentz method, which is based on continued fractions (Lentz, 1976).

Then the multiplication step (7) simply produces a vMF posterior distribution parametrized as:

$$\boldsymbol{\mu}_t = \frac{1}{\kappa_t}(\kappa_{obs}\boldsymbol{y}_t + \kappa'\boldsymbol{\mu}') \quad \kappa_t = \|\kappa_{obs}\boldsymbol{y}_t + \kappa'\boldsymbol{\mu}'\|. \quad (11)$$

3.3 Partial observations

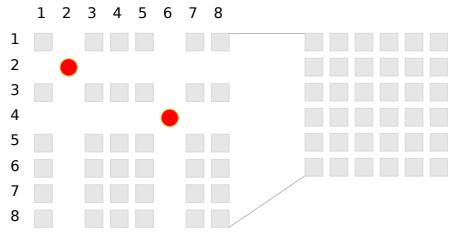


Figure 3: An example of a fallback to a lower dimensional permutation space after a partial observation.

The tracking algorithm presented above presumes that a full (noisy) assignment is included in each observation. However, in realistic tracking problems, this is unnecessarily restrictive. We would like to update the model even if only a subset of tracks can be observed.

Partial observations are conceptually straightforward in the permutation matrix representation. When a partial observation of o objects becomes available, the dimension of the unknown part of the observed permutation matrix $\mathbf{Y}$ is reduced from n^2 to $(n-o)^2$. The

mechanism of this is shown in Figure 3, where circles indicate two observed objects and squares indicate the unknown parts of $\mathbf{Y}$. In this representation, $\mathbf{Y}$ can be separated cleanly into an observed part $\mathbf{Y}_*$ and an unobserved part $\mathbf{Y}_?$, as $\mathbf{Y} = \mathbf{Y}_* + \mathbf{Y}_?$.

Projecting a partial observation into the $\mathbb{S}^d$ embedding space, the observed and unobserved vectors will no longer be separable elementwise. However, we can still identify them as separate components, $\boldsymbol{y}_*$ and $\boldsymbol{y}_?$. Carrying out this transformation, denoting the orthogonal part of the basis in $\mathbb{R}^{n^2}$ that represents the $\mathbb{R}^{(n-1)^2}$ subspace by an $n^2 \times (n-1)^2$ matrix $\mathbf{Q}$, we get:

$$\boldsymbol{y} = \mathbf{Q}^T vec\left((\mathbf{Y}_* + \mathbf{Y}_?) - \frac{\mathbf{1}}{\mathbf{n}}\mathbb{1}\right) = \boldsymbol{y}_* + \boldsymbol{y}_? \qquad (12)$$

Here $\boldsymbol{y}$ ranges over $\mathbb{S}^d$, while in general $\boldsymbol{y}_*$ and $\boldsymbol{y}_?$ are vectors in $\mathbb{R}^{(n-1)^2}$ that fall inside the hypersphere.

To apply the observation model in (7) to a new partial observation, we first need to marginalize out the unobserved portion, producing a new vMF observation model $p(\mathbf{y}_{t,*}|\mathbf{x}_t)$ that we can apply to the observed portion. Performing the marginalization and suppressing t subscripts, we see that the unobserved portion factors out, effectively becoming part of the normalization constant

$$\frac{1}{Z}\int_{\boldsymbol{y}_?} e^{\kappa_{obs}(\boldsymbol{y}_* + \boldsymbol{y}_?)^T \boldsymbol{x}} d\boldsymbol{y}_? =$$
$$\frac{1}{Z}\left(\int_{\boldsymbol{y}_?} e^{\kappa_{obs}\boldsymbol{y}_?^T \boldsymbol{x}} d\boldsymbol{y}_?\right) e^{\kappa_{obs}\boldsymbol{y}_*^T \boldsymbol{x}} \qquad (13)$$

where Z is the original normalization constant.

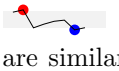
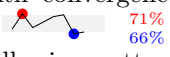
Some details make computing the integral in (13) not totally trivial: $\boldsymbol{x}$, $\boldsymbol{y}_*$, and $\boldsymbol{y}_?$ are of different length; and although $\boldsymbol{x}$ is fixed, $\boldsymbol{y}_*$ and $\boldsymbol{y}_?$ are not allowed to take any possible angle in $\mathbb{R}^{(n-1)^2}$. We omit the details of the derivation dealing with these difficulties and just state the parameters of the resulting vMF likelihood function:

$$\boldsymbol{\mu} = \frac{\boldsymbol{y}_*}{\|\boldsymbol{y}_*\|_2}, \qquad \kappa = \|\kappa_{obs}\boldsymbol{y}_*\|_2. \qquad (14)$$

Thus, in the case of vMF we can execute a recursive Bayesian filter using only analytical computation even when only partial observations are available. This makes the state space model applicable in a much wider range of scenarios than our initial model presented in Section 3.2.

4 Experiments

To demonstrate the correctness of our approach, we show inference of a fixed hidden permutation from its noisy partial observations. Figure 4 shows results of this inference on datasets of 25 and 50 objects. In these synthetic data experiments, we randomly chose a true (hidden) permutation, $\mathbf{P}_{\text{true}}$. We controlled both observation noise ($\nu \in \{0.1, 0.2, ..., 0.9\}$) and the fraction of objects missing from observations ($m \in \{0\%, 20\%, 40\%, 60\%\}$). Noisy observations were drawn from vMF ($\mathbf{P}_{\text{true}}$,κ_ν), where κ_ν was chosen to achieve the fraction ν of incorrectly observed object identities. The final observation, $\mathbf{P}_m$, was generated by hiding m percent of entries from the noisy observation matrix, chosen uniformly at random without replacement. The error in this section is the ratio of incorrectly identified objects to the total number of objects. Figure 4 shows that our representation of the $n!$ discrete permutation space is functional and the approach can gracefully handle large numbers of objects, partial observations and observation noise.

We use the above setup to report run time in seconds on a 2.2Hz PC[7]. For each $n \in \{10, 20 \ldots 80\}$, we estimate 100 random fixed permutations to convergence. The average number of iterations until convergence  and observation errors  are similar across values of n. The following pattern

n	10	20	30	40	50	60	70	80
μ	0.02	0.06	0.35	1.24	2.92	6.21	9.84	17.2

holds for the mean running times μ (seconds) to reach the convergence.

The above simulation was generated with the noise model used by the inference and did not have a temporal component, although it was applied to a really large state space. Next we show experiments on a tracking dataset with a non-vMF transition model. We use a dataset of planar locations of aircraft within a 30 mile diameter of John F. Kennedy airport of New York. The data, in streaming format, is available at `http://www4.passur.com/jfk.html`. The complexity of the plane routes and frequent crossings of tracks in the planar projection make this an interesting dataset for identity tracking. Identity tracking results on this dataset, in the context of the symmetric semigroup approach to permutation inference, were previously reported in Kondor et al. (2007). Replicating the task reported in Kondor et al. (2007), we show results on tracking datasets of 6 and 10 flights, dropping the 15-flight dataset (but see a 41-object dataset below).

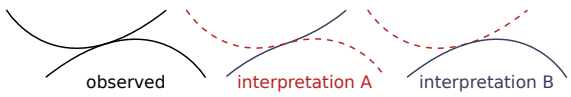
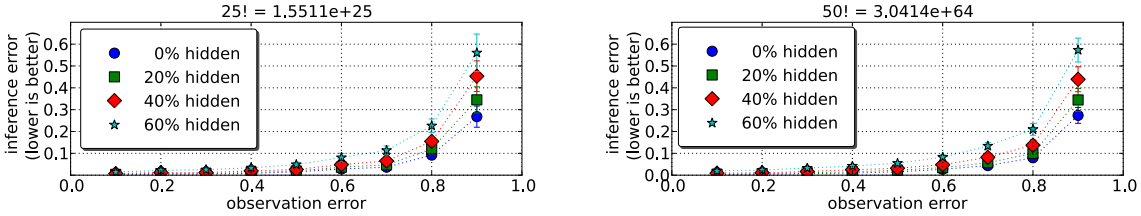


Figure 6: Example errors in track interpretation.

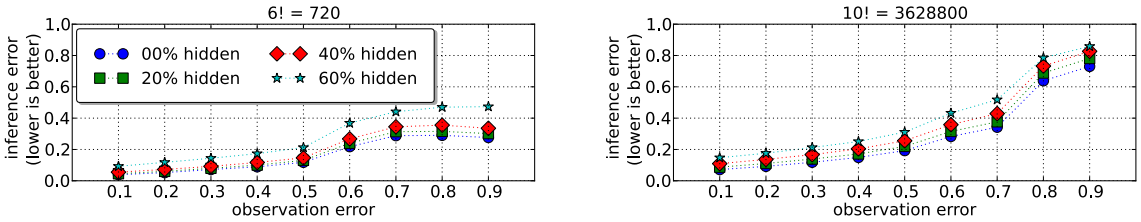
The dataset comes prelabeled, and both the correct

[7] Experiments are based on the supplemental MATLAB code.



(a) 25 objects (b) 50 objects

Figure 4: Average error of a random hidden permutation inference from 100 (partial) noisy observations on 25 and 50 objects simulated datasets. Runs were repeated 10 times with a different true permutation each.



(a) 6 flights (b) 10 flights

Figure 5: Tracking error on the air traffic control dataset for 6 and 10 flights as a function of observation noise shown as the fraction of incorrectly reported identities. Separate plots show error for partial observations when a fraction of object identities is unobserved. The legend is shared between subplots.

tracks and the identities of the flights following them are known. In completely real settings, the tracks provided to the identity manager are not the true tracks, but their interpretations (see Figure 6). To model that, uncertainty is introduced by randomly swapping identities of flights i and j at their respective locations $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ with probability $p_{swap}exp(-\|\boldsymbol{x}_j(t) - \boldsymbol{x}_i(t)\|^2/(2s^2))$, where $p_{swap} = 0.1$ and $s = 0.1$ are strength and scale parameters, respectively. We also use this model for observation noise later in the section.

We then generated observation and hidden identity noise in the same way as for the prior experiment. Figure 5 shows results of applying our identity tracking method to the air traffic control dataset for various levels of observation noise and amount of missing identity observations. It is difficult to compare the performance to the method of Kondor et al. (2007) applied to the same dataset, since it is not clear how observation noise levels correspond to each other. However, error values reported in Kondor et al. (2007) were 0.12 to 0.17 on the 6 flights dataset and 0.2 to 0.32 on the 10 flights dataset. This is comparable to what we get with our approach for observation error below 50%, even when 60% of the flight identities are unobserved. Results of the application of our state space model to this dataset indicate robustness of the model to the choice of the transition model, which was different from the generative model of our tracking inference engine.

To further test our embedding in the tracking task, we have used the flight dataset for 10 flights from the above and compared our approach to the information-form data association (IFDA) filter of Schumitsch et al. (2005). Note that IFDA is specifically built for the PDA problem and does not represent distributions on permutations, as our approach does, avoiding any overhead. The speeds of approaches are comparable since the bottleneck in both of them is the graph matching algorithm. We have used vMF noise model to compare how both models perform under it, as well as proximity noise model when IFDA unlike vMF was provided with additional location information and exact generation parameters. Results of the comparison for various noise levels and ratios of unobserved identities are shown in Figure 7.

Due to the unmanageable size of the factorial space in identity tracking problems, even the powerful methods based on Fourier representation of permutations do not report results on more than 11 (Huang et al., 2009) or 15 (Kondor et al., 2007) simultaneously tracked objects. The results of Figure 4 show that our approach can handle large numbers of objects while operating on $n!$ objects consistently, and Figure 7 demonstrates comparable or better accuracy on the air traffic control dataset. Next we show comparison results with IFDA (which can manage the size of the dataset) on 41 objects from a visual surveillance dataset available from `http://vspets.visualsurveillance.org/`. Figure 8 shows an example of the underlying data and results of the identity tracking. The problem is similar to the above air traffic control. Our approach handles the situation and

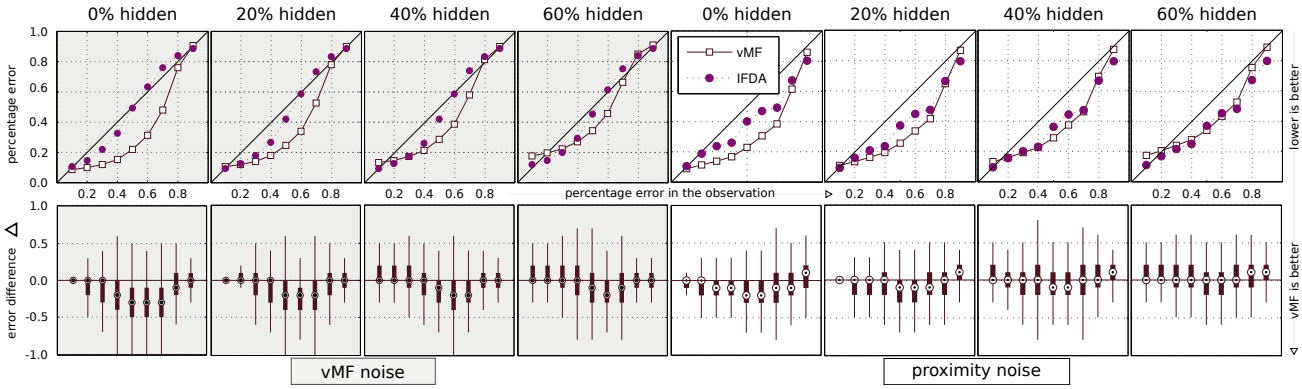
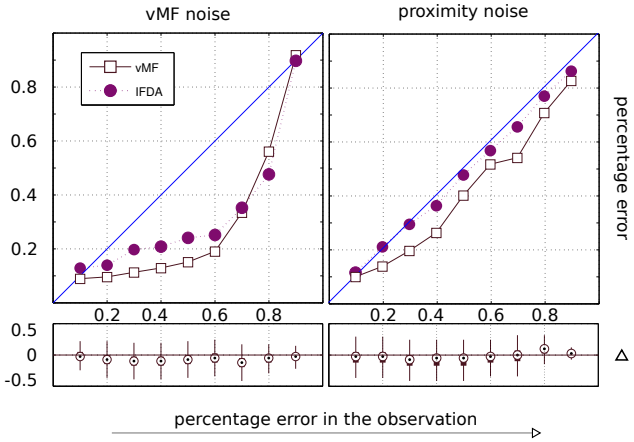


Figure 7: Comparing accuracy of our vMF approach and IFDA on the air traffic control dataset for the case of 10 flights: top row – mean errors, bottom row – boxplot of the pointwise difference between vMF and IFDA errors (Δ). vMF noise satisfies the assumptions of our model, while in the proximity noise case not only the model follows IFDA assumptions but all parameters were set to their true values for IFDA's benefit.

(a) a frame from the tracking task



(b) tracking identities of 41 players

Figure 8: Tracking error comparison of vMF and IFDA approaches on a soccer visual surveillance dataset for 41 players as a function of observation noise level and for 2 kinds of noise the same as in Figure 7. In proximity noise case IFDA is provided with all exact parameters. Δ is vMF errors minus IFDA errors as before.

produces reasonable results with acceptable error rate – almost always better than IFDA and sometimes comparable.

5 Conclusions

The main result of this work is embedding permutations into a continuous manifold, thus lifting a body of results from the directional statistics field (Mardia and Jupp, 2000) to the fields of ranking, identity tracking, and others where permutations play an essential role. Among many potential applications of this embedding we have chosen probabilistic identity tracking as an example and were able to set up a state-space model with efficient recursive Bayesian filter that produced results comparable with the state-of-the-art techniques very efficiently, even on very large datasets that pose difficulties for existing methods. Our model was as fast and often more accurate than IFDA (Schumitsch et al., 2005), which was specifically designed for tracking, while operating with probability densities defined on the $n!$ space, similarly to less efficient but expressive methods (Kondor et al., 2007; Huang et al., 2009). There remains much to be done in this direction. However, our model has already efficiently produced results of a reasonable accuracy. This is promising and encourages further development of more complicated probability distributions for permutations: further exploration of the exponential family already developed in the field (Mardia and Jupp, 2000) as well as developing more complex representations using spherical harmonics.

Acknowledgments

This work was supported by NIH/NIBIB grant #2R01 EB000840-06 and NSF IIS grant #0705681. We thank Risi Kondor for providing the air traffic control dataset, Diane Oyen for proofreading, and our reviewers for helping us to improve the paper.

References

R. Kondor, A. Howard, and T. Jebara. Multi-object tracking with representations of the symmetric group. In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics*, March 2007.

J. R. Munkres. *Topology: A First Course*. Prentice Hall, 1975.

E. Kreyszig. *Introductory Functional Analysis with Applications*. John Wiley & Sons, 1978.

R. E. Gomory. Outline of an algorithm for integer solutions to linear programs. *Bulletin of the American Mathematical Society*, 64(5):275–278, 1958. doi: 10.1090/S0002-9904-1958-10224-4.

T. Jaakkola, D. Sontag, A. Globerson, and M. Meila. Learning bayesian network structure using LP relaxations. In Y. W. Teh and M. Titterington, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, 2010.

M. Meila, K. Phadnis, A. Patterson, and J. Bilmes. Consensus ranking under the exponential model. In *Proceedings of the 23rd Annual Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 285–294, Corvallis, Oregon, 2007. AUAI Press.

M. A. Fligner and J. S. Verducci. Distance based ranking models. *Journal of the Royal Statistical Society. Series B (Methodological)*, 48(3):359–369, 1986.

J. Huang, C. Guestrin, and L. Guibas. Fourier theoretic probabilistic inference over permutations. *Journal of Machine Learning (JMLR)*, 10:997–1070, May 2009.

B. Schumitsch, S. Thrun, G. Bradski, and K. Olukotun. The information-form data association filter. In *Proceedings of Conference on Neural Information Processing Systems (NIPS)*, Cambridge, MA, 2005. MIT Press.

K. V. Mardia and P. E. Jupp. *Directional Statistics*. John Wiley & Sons, 2 edition, 2000. ISBN 0-471-95333-4.

T. E. S. Raghavan R. B. Bapat. *Nonnegative Matrices and Applications*. Encyclopedia of mathematics and its applications. Cambridge University Press, 1997.

C. D. Meyer. *Matrix analysis and applied linear algebra*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2000. ISBN 0-89871-454-0.

D. B. West. *Introduction to graph theory*. Prentice Hall, 2001.

J. Huang, C. Guestrin, and L. Guibas. Efficient inference for distributions on permutations. *Advances in Neural Information Processing Systems*, 20:697–704, 2008.

C. Rasmussen and G. D. Hager. Probabilistic data association methods for tracking complex visual objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):560–576, 2001.

W. J. Lentz. Generating Bessel functions in Mie scattering calculations using continued fractions. *Applied Optics*, 15:668–671, 1976.