Motion Tomography via Occupation Kernels

Benjamin P. Russo¹, Rushikesh Kamalapurkar², Dongsik Chang³, and Joel A. Rosenfeld⁴

Abstract—The goal of motion tomography is to recover the description of a vector flow field using information about the trajectory of a sensing unit. In this paper, we develop a predictor corrector algorithm designed to recover vector flow fields from trajectory data with the use of occupation kernels developed by Rosenfeld et al.¹, ². Specifically, we use the occupation kernels as an adaptive basis; that is, the trajectories defining our occupation kernels are iteratively updated to improve the estimation on the next stage. We show for a simulated example we have good accuracy in recovering the flow-field using a simple metric. We also apply our algorithm to real world data first presented in [5].

I. INTRODUCTION

Over the past decade, unmanned aircraft and underwater systems have evolved significantly and are on the verge of becoming a ubiquitous part of urban and littoral landscape. To compensate for the lack of access to the global positioning system (GPS), unmanned underwater vehicles (UUVs) and vehicle-induced noise (in the case of UAVs, especially multi-rotor UAVs), creates significant challenges in acquisition and processing of the data generated by on-board sensors. The aforementioned challenges, along with the payload and cost reduction associated with removing flow sensors, motivates the development of estimation techniques that rely only on the effect of the flow field on the motion of UAVs and UUVs, and not on direct measurements of the flow velocities.

Motion tomography refers to the reconstruction of a vector field using its accumulated effects on mobile sensing units as they travel through the field [3], [7]. Motion tomography allows for the use of low cost mobile underwater/air vehicles as sensors to accumulate sufficient data for estimation of vector fields resulting from wind and ocean currents. As a result, military applications such as ocean current mapping for navigation of mine countermeasure UUVs in littoral environments, commercial applications such as wind field mapping for navigation of small package delivery UAVs, and disaster response applications such as wind field mapping for the prediction of flame front propagation and smoke spread, stand to benefit from fast and accurate motion tomography. In this paper we propose an algorithm for motion tomography based on occupation kernels developed in [1], [2].

The developed approach to motion tomography has several advantages over existing techniques such as [6]. The flow field is approximated here using the occupation kernels as basis functions for approximation, whereas [3] requires a piecewise constant description of the flow field or a parameterization with respect to Gaussian RBFs. Moreover, [5] employs a renormalization routine which imposes limitations on the motion of the mobile sensors. The proposed occupation kernel method avoids the renormalization and does not add further restrictions on the motion. Finally, the representation of the flow field with respect to the occupation kernel basis allows for the application of the approximation abilities of RKHS's.

II. TOOLS

A reproducing kernel Hilbert space (RKHS), \( H \), over a set \( X \) is a Hilbert space of real valued functions over the set \( X \) such that for all \( x \in X \) the evaluation functional \( E_x g := g(x) \) is bounded. As such, the Riesz representation theorem guarantees, for all \( x \in X \), the existence of a function \( k_x \in H \) such that \( \langle g, k_x \rangle_H = g(x) \), where \( \langle \cdot, \cdot \rangle_H \) is the inner product for \( H \) [8, Chapter 1]. The function \( k_x \) is called the reproducing kernel function at \( x \), and the function \( K(x,y) = \langle k_y, k_x \rangle_H \) is called the kernel function corresponding to \( H \).

Definition 1. Let \( X \subset \mathbb{R}^n \) be compact, \( H \) be a RKHS of continuous functions over \( X \), and \( \gamma : [0,T] \rightarrow X \) be a bounded measurable trajectory. The functional \( g \mapsto \int_0^T g(\gamma(\tau))d\tau \) is bounded, and may be represented as \( \int_0^T g(\gamma(\tau))d\tau = \langle g, \Gamma_\gamma \rangle_H \), for some \( \Gamma_\gamma \in H \) by the Riesz representation theorem. The function \( \Gamma_\gamma \) is called the occupation kernel corresponding to \( \gamma \) in \( H \).

The value of an inner product against an occupation kernel in a RKHS can be approximated by leveraging quadrature techniques for integration. The numerical experiments described below utilize Simpson’s rule when computing inner products involving occupation kernels and their associated Gram matrices. We present Theorem 2 to validate the convergence of numerical evaluation of the inner product using Simpson’s Rule. Moreover, the occupation kernels themselves can be expressed as an integral against the kernel

---

¹This research was supported, in part, by the Air Force Office of Scientific Research under award number FA9550-20-1-0127 and the National Science Foundation under award number 2027999. Any opinions, findings, or recommendations in this article are those of the author(s), and do not necessarily reflect the views of the sponsoring agencies.

¹Farmingdale State College, email: russobp@farmingdale.edu.
²Oklahoma State University, email: rushikesh.kamalapurkar@okstate.edu.
³University of Michigan, email: dongsik@umich.edu.
⁴University of South Florida, email: joelr@usf.edu.
function in a RKHS as demonstrated in Proposition 1. Theorem 2 and Proposition 1 originally appear in [2], but are reproduced below for completeness.

**Proposition 1.** Let $H$ be a RKHS over a compact set $X$ consisting of continuous functions and let $\gamma : [0, T] \rightarrow X$ be a continuous trajectory as in Definition 2. The occupation kernel corresponding to $\gamma$ in $H$, $\Gamma_\gamma$, may be expressed as

$$\Gamma_\gamma(x) = \int_0^T K(x, \gamma(t))dt.$$  

**Proof.** Note that $\Gamma_\gamma(x) = \langle \Gamma_\gamma, K(\cdot, x) \rangle_H$, by the reproducing property of $K$. Consequently,

$$\Gamma_\gamma(x) = \langle \Gamma_\gamma, K(\cdot, x) \rangle_H = \langle K(\cdot, x), \Gamma_\gamma \rangle_H = \int_0^T K(\gamma(t), x)dt = \int_0^T K(x, \gamma(t))dt,$$

which establishes the result.

Given a kernel function for the RKHS and a trajectory $\gamma$, the occupation kernel corresponding to $\gamma$ in $H$ can be numerically computed by leveraging Proposition 1. Quadrature methods can be demonstrated to result in not only pointwise convergence but also norm convergence of the numerical estimate of the occupation kernel to the actual occupation kernel in the RKHS, which is a strictly stronger result.

**Theorem 2.** Under the hypothesis of Proposition 1 let $t_0 = 0 < t_1 < t_2 < \ldots < t_F = T$ (with $F$ even and $t_i$ evenly spaced), suppose that $\gamma$ is a fourth order continuously differentiable trajectory and $H$ is composed of fourth order continuously differentiable functions. Set $h$ to satisfy $t_i = t_0 + ih$, and consider

$$\hat{\Gamma}_\gamma(x) := \frac{h}{3} \left( K(x, \gamma(t_0)) + 4 \sum_{i=1}^{n-1} K(x, \gamma(t_{2i-1})) + 2 \sum_{i=1}^{n-1} K(x, \gamma(t_{2i})) + K(x, \gamma(t_F)) \right).$$  

The norm distance is bounded as $\|\Gamma_\gamma - \hat{\Gamma}_\gamma\|_H = O(h^4)$.

**Proof.** Consider $\|\Gamma_\gamma - \hat{\Gamma}_\gamma\|_H = \|\Gamma_\gamma, \Gamma_\gamma\|_H + \langle \Gamma_\gamma, \hat{\Gamma}_\gamma \rangle_H - 2\langle \Gamma_\gamma, \hat{\Gamma}_\gamma \rangle_H$. The term $\langle \Gamma_\gamma, \hat{\Gamma}_\gamma \rangle_H$ is an implementation of the two-dimensional Simpson’s rule (cf. 2) while $\|\Gamma_\gamma, \Gamma_\gamma\|_H$ is the double integral $\int_0^T \int_0^T K(\gamma(t), \gamma(\tau))dt\,d\tau$. Thus,

$$\langle \Gamma_\gamma, \hat{\Gamma}_\gamma \rangle_H = \langle \Gamma_\gamma, \Gamma_\gamma \rangle_H + O(h^4).$$

Similarly, $\langle \Gamma_\gamma, \hat{\Gamma}_\gamma \rangle_H$ integrates in one variable while implementing Simpson’s rule in the other. Consequently,

$$\langle \Gamma_\gamma, \hat{\Gamma}_\gamma \rangle_H = \langle \Gamma_\gamma, \Gamma_\gamma \rangle_H + O(h^4).$$

The conclusion of the theorem follows.

## III. Problem Setup and Developed Algorithm

Let $r : [0, T] \rightarrow \mathbb{R}^2$ represent a continuous trajectory for a mobile sensor attempting to travel in a straight line in the direction $\theta$ from its starting point, but subject to an unknown flow field, $F : R \subset \mathbb{R}^2 \rightarrow \mathbb{R}^2$, where $R$ is a compact subset of the plane. Let

$$\dot{r} = s \begin{pmatrix} \cos(\theta) \\ \sin(\theta) \end{pmatrix}^T + F(r),$$

for a positive constant $s$, represent the true dynamics induced by the flow field. If the sensor traveled unperturbed by a flowfield the dynamics would simply be $\dot{r} = s \begin{pmatrix} \cos(\theta) \\ \sin(\theta) \end{pmatrix}^T$. We will assume $F : R \rightarrow \mathbb{R}^2$ is locally Lipschitz in order to assure uniqueness of the solutions [10]. As the flow field is unknown, the anticipated dynamics are

**Algorithm 1:** Iterative Algorithm

**Define** $N$ as the number of iterates

**Input** Samples $r_i(T)$ \quad $i \in \{1, \ldots, M\}$

**Begin**

Generate via a numerical method $\tilde{r}_{i,0} : [0, 1] \rightarrow R$, the unique solution to

$$\dot{\tilde{p}} = s_i \begin{pmatrix} \cos(\theta_i) \\ \sin(\theta_i) \end{pmatrix}^T, \quad p(0) = p_i$$

for $i \in \{1, \ldots, M\}$

Set $D_{i,0} = r_i(T) - \tilde{r}_{i,0}(T)$ \quad $i \in \{1, \ldots, M\}$

Set $\tilde{F}_{-1} = 0$

**For** $n$ in \{$0, \ldots, N$\} **do**

**Input** $\tilde{F}_n = \sum_{i=1}^M w_{i,n} \Gamma_{\tilde{r}_{i,n}}$

**Generate** via a numerical method $\tilde{r}_{i,n+1} : [0, 1] \rightarrow R$ the unique solution to

$$\dot{\hat{p}} = s_i \begin{pmatrix} \cos(\theta_i) \\ \sin(\theta_i) \end{pmatrix}^T + \tilde{F}_n(p), \quad p(0) = p_i$$

for $i \in \{1, \ldots, M\}$

Set $D_{i,n+1} = r_i(T) - \tilde{r}_{i,n+1}(T)$ \quad $i \in \{1, \ldots, M\}$

**Compute** $\tilde{F}_{n+1} = \sum_{i=1}^M w_{i,n+1} \Gamma_{\tilde{r}_{i,n+1}}$ by solving

$$\begin{pmatrix} \langle \Gamma_{\tilde{r}_{i,n+1}}, \Gamma_{\tilde{r}_{j,n+1}} \rangle_{M,M} \\ \vdots \\ \langle \Gamma_{\tilde{r}_{i,n+1}}, \Gamma_{\tilde{r}_{M,n+1}} \rangle_{M,M} \end{pmatrix} \begin{pmatrix} w_{1,n+1} \\ \vdots \\ w_{M,n+1} \end{pmatrix} = \begin{pmatrix} D_{1,n+1} + \langle \tilde{F}_n, \Gamma_{\tilde{r}_{1,n+1}} \rangle_{M,M} \\ \vdots \\ D_{M,n+1} + \langle \tilde{F}_n, \Gamma_{\tilde{r}_{M,n+1}} \rangle_{M,M} \end{pmatrix}$$

for all $i \in \{1, \ldots, M\}$

**Output** $\tilde{F}_{n+1} = \sum_{i=1}^M w_{i,n+1} \Gamma_{\tilde{r}_{i,n+1}}$
\[ \dot{r}(T) \text{, is given as} \]
\[ D = r(T) - \dot{r}(T) = \int_0^T (\dot{r}(t) - \dot{r}(t)) \, dt \]
\[ = \int_0^T F(r(t)) dt = \{ F, \Gamma_r \}_H. \]

That is, the difference between the final locations of the mobile sensor describes the integral of the flow field along the trajectory \( r \), and this integral provides a type of tomographic sample of \( F \). As the trajectory \( r \) is treated as unknown, an approximation of \( F \) using the sample generated by \( \Gamma_r \) is difficult to assess directly. This motivates the developed iterative algorithm to determine the flow field, \( F \), as well as the true trajectories, \( r \).

Let \( \{ s_i \}_{i=1}^M, \{ \theta_i \}_{i=1}^M \) be a collection of speeds and angles used to generate a collection of anticipated trajectories, i.e. trajectories governed by the dynamics \( \dot{r} = s_i (\cos(\theta_i) \sin(\theta_i)) \). Moreover, let \( \{ p_i \}_{i=1}^M \subset \mathbb{R}^2 \) represent the starting point of the trajectories. For each iteration of Algorithm 1, the distance between the endpoints of the true trajectory and the predicted trajectory is calculated. This value is then used to alter our flowfield by projecting the updated data onto the new collection of occupation kernels and then in turn a new predicted trajectory is simulated via a method like fourth order Runge-Kutta.

Significantly, after the initial data collection period, no further experiments are necessary to approximate the flow field. Specifically, Algorithm 1 only needs to produce new simulations of the approximate trajectories which ultimately converge to the true trajectories from the initial experiment.

IV. CONVERGENCE OF ALGORITHM

Future work will be dedicated to providing sufficient conditions for Algorithm 1 to converge via the contraction mapping theorem. However, in contrast to Theorem 3 our numerical experiments suggest that there are indeed a variety of conditions under which our algorithm converges.

Numerical stability of the developed algorithm relies on the invertability of the occupation kernel Grammian \( G_\Gamma \), which, for a set of trajectories \( r_i : [0, T] \to \mathbb{R}^2 \), with \( 1 \leq i \leq N \), is defined as
\[
G_\Gamma := \left( (\Gamma_{r_j}, \Gamma_{r_k}) \right)_{j,k=1}^N \]
\[ = \int_0^T \int_0^T (K(r_j(t), r_k(\tau))) dt \, d\tau. \]

The next theorem will be a trajectory variant of a theorem found in Wendland [11, Theorem 12.3]. In [11, Theorem 12.3], a bound for the minimum eigenvalue of the gramian is obtained by expressing the kernel function \( \Phi(x - y) \) in terms of its Fourier transform and comparing with an intermediate radial function \( \Psi_M \) constructed using characteristic functions for balls of radius \( M \). The comparison is given in terms of the separation distance between centers of the kernel functions. In order to apply the results from [11, Theorem X1] to analyze the stability of the interpolation process in Algorithm 1, we define the following separation distance:

**Definition 2.** The trajectory separation distance is given by
\[
q_{X,T} := \frac{1}{2} \min_{t, r \in [0,T]} \| \gamma_j(t) - \gamma_k(\tau) \|_2
\]

This is the maximum radius such that all “tubes” centered on the trajectories are disjoint. By defining the trajectory separation distance as above the inequalities found in [11, Theorem 12.3] remain valid since the points in the image of each trajectory are separated by at least \( q_{X,T} \).

**Theorem 3.** With \( \Phi(x - y) = K(x, y) = \exp(-\mu \| x - y \|^2_2) \) the minimal eigenvalue of the Occupation kernel Grammian is bounded by
\[
\lambda_{\min}(G_\Gamma) \geq \frac{C_2 \exp(-M_2/(q_{X,T} \mu)) T^2}{2\mu q_{X,T}^2},
\]
with \( M_2 = 12 \sqrt{\frac{T}{\mu}} \) and \( C_2 = \frac{M^2}{16} \).

The proof of this theorem will also be presented in future work mentioned above, but it is the natural extension of [11, Theorem 12.3] to trajectories.

**V. NUMERICAL EXPERIMENTS**

For our first experiment, Figure 2, we generated a slope field \( F(x) \) using a linear combination of Gaussian kernels and generated a set of random points and angles to serve as our anticipated dynamics using unit speed. We then performed 10 iterations of our algorithm.

\[
F(x) = 1 \frac{1}{8} \left( \frac{f_1(x)}{f_2(x)} \right)
\]

\[
f_1(x) = 5 \exp \left( -2 \| x - \left[ \begin{array}{c} 25 \\ 25 \end{array} \right] \|_2^2 \right) - 2 \exp \left( -2 \| x - \left[ \begin{array}{c} 25 \\ 75 \end{array} \right] \|_2^2 \right) + 2 \exp \left( -2 \| x - \left[ \begin{array}{c} 75 \\ 25 \end{array} \right] \|_2^2 \right)
\]

\[f_2(x) = 3 \exp \left( -2 \| x - \left[ \begin{array}{c} 25 \\ 25 \end{array} \right] \|_2^2 \right) + \exp \left( -2 \| x - \left[ \begin{array}{c} 25 \\ 75 \end{array} \right] \|_2^2 \right) - 3 \exp \left( -2 \| x - \left[ \begin{array}{c} 75 \\ 25 \end{array} \right] \|_2^2 \right).\]

Quantitatively, given a sample of vectors from two vector fields \( V(x, y) \) and \( W(x, y) \), we can define the max error (relative to \( V \)) as
\[
\text{Max Error} = \max_i \| V(x_i, y_i) - W(x_i, y_i) \| / \| V(x_i, y_i) \|
\]
and the mean error (relative to \( V \)) as
\[
\text{Mean Error} = \text{Mean} \{ \| V(x_i, y_i) - W(x_i, y_i) \| / \| V(x_i, y_i) \| \}
\]

Let \( V(x, y) \) represent the true vector field and let \( W(x, y) \) denote the estimated vector field. For the 10 iterations performed in Experiment 1, we summarize the accuracy of our algorithm via the defined metric and display this in Table 1.
TABLE I: Max and Mean Errors for the 10 iterations of Algorithm 1.

|          | 10 iterations |
|----------|---------------|
| Max Error| 0.25304       |
| Mean Error| 0.025608     |

Fig. 1: True vector field and trajectories for Experiment 1.

Fig. 2: Results of Experiment 1. The true trajectories are calculated via RK4 over the time frame [0, 1]. Using Gaussian RBFs with a kernel width of 1, we performed 10 iterations of Algorithm 1.

Fig. 3: Initial trajectories

Fig. 4: Approximated field and trajectories after 5 iterations.

Fig. 5: Approximate field and trajectories with 10 iterations.

For our second experiment, Figures 3, 4, 5, and 6, we used the Gliderpalooza 2013 data first presented in [3]. The data was for 31 sequential trajectory segments and contained the initial positions, the true final positions, dead-reckoned final positions, speeds and dead-reckoning times. For their experiments, they used averaged speeds and an average dead-reckoning time of 3.5 hours.

Algorithm [1] is designed to work on multiple trajectories necessitating some prior calculations. The data was scaled by a factor of $10^{-4}$, then the provided initial positions and dead-reckoned final positions were used to calculate the...
initial directions along with calculated speeds using their averaged dead-reckoning time of 3.5 hours. The sequential trajectory data was broken into 31 separate trajectories. Using exponential kernels, we performed 5, 10, and 20 iterations of Algorithm 1 using a kernel width of $\mu = 1/170$ for the 5 iteration run and a kernel width of $\mu = 1/10,000$ for the 10 and 20 iteration runs. The kernel widths for each of these runs were selected to produce well conditioned Gram matrices.

VI. DISCUSSION

This approach to motion tomography has several advantages over that of [3]. In the context of the ocean current estimation problem, the flow field is approximated using the occupation kernels as basis functions for approximation, whereas [3] requires a piece-wise constant description of the flow field or a parameterization with respect to Gaussian RBFs. Moreover, [3] renormalizes the integral by multiplying and dividing by $||\dot{\mathbf{r}}(t)||_2$ to artificially convert the integrals to line integrals. The renormalization may ultimately produce divide by zero errors that leads to stronger assumptions on the dynamical systems. The occupation kernel method avoids the renormalization and does not add further restrictions on the dynamics. Finally, the representation of the flow field with respect to the occupation kernel basis allows for the application of the approximation powers of RKHSs. It should also be noted that although the numerical experiments presented in this paper apply the developed technique to $\mathbb{R}^2$ valued functions, there is no inherit reason to limit to $\mathbb{R}^2$. That is, this technique would apply to $\mathbb{R}^d$ valued functions.

REFERENCES

[1] J. A. Rosenfeld, R. Kamalapurkar, L. Gruss, and T. T. Johnson, “Dynamic mode decomposition for continuous time systems with the liouville operator,” arXiv preprint arXiv:1910.03977, 2019.

[2] J. A. Rosenfeld, B. Russo, R. Kamalapurkar, and T. T. Johnson, “The occupation kernel method for nonlinear system identification,” arXiv preprint arXiv:1909.11792, 2019.

[3] D. Chang, W. Wu, C. R. Edwards, and F. Zhang, “Motion tomography: Mapping flow fields using autonomous underwater vehicles,” The International Journal of Robotics Research, vol. 36, no. 3, pp. 320–336, 2017.

[4] J. Petrich, C. A. Woolsey, and D. J. Stihewell, “Planar flow model identification for improved navigation of small AUVs,” Ocean Eng., vol. 36, no. 1, pp. 119–131, 2009.

[5] A. S. Aweiss, B. D. Owens, J. Rios, J. R. Homola, and C. P. Mohlenbrink, “Unmanned aircraft systems (UAS) traffic management (UTM) national campaign II,” in AIAA Inf. Syst.-AIAA Infotech@Aerosp., 2018, p. 1727.

[6] V. Stepanyan and K. S. Krishnakumar, “Estimation, navigation and control of multi-rotor drones in an urban wind field,” in AIAA Inf. Syst.-AIAA Infotech@Aerosp., 2017, p. 0670.

[7] W. Wu, D. Chang, and F. Zhang, “Glider ct: Reconstructing flow fields from predicted motion of underwater gliders,” in Proceedings of the eighth ACM international conference on underwater networks and systems, 2013, pp. 1–8.

[8] V. I. Paulsen and M. Ragupathi, An introduction to the theory of reproducing kernel Hilbert spaces, ser. Cambridge Studies in Advanced Mathematics. Cambridge University Press, Cambridge, 2016, vol. 152, pp. x+182, ISBN: 978-1-107-10409-9.

[9] R. Burden, R. Burden, and J. Faires, Numerical Analysis, ser. Numerical Analysis v. 1. Brooks/Cole, 2001, ISBN: 9780534382162.

[10] R. Thompson and W. Walter, Ordinary Differential Equations, ser. Graduate Texts in Mathematics. Springer New York, 2013, ISBN: 9781461206019.

[11] H. Wendland, Scattered data approximation. Cambridge university press, 2004, vol. 17.