Controlled density transport using Perron Frobenius generators

Jake Buzhardt and Phanindra Tallapragada

Abstract— We consider the problem of the transport of a density of states from an initial state distribution to a desired final state distribution through a dynamical system with actuation. In particular, we consider the case where the control signal is a function of time, but not space; that is, the same actuation is applied at every point in the state space. This is motivated by several problems in fluid mechanics, such as mixing and manipulation of a collection of particles by a global control input such as a uniform magnetic field, as well as by more general control problems where a density function describes an uncertainty distribution or a distribution of agents in a multi-agent system. We formulate this problem using the generators of the Perron-Frobenius operator associated with the drift and control vector fields of the system. By considering finite-dimensional approximations of these operators, the density transport problem can be expressed as a control problem for a bilinear system in a high-dimensional, lifted state. With this system, we frame the density control problem as a problem of driving moments of the density function to the moments of a desired density function, where the moments of the density can be expressed as an output which is linear in the lifted state. This output tracking problem for the lifted bilinear system is then solved using differential dynamic programming, an iterative trajectory optimization scheme.

I. INTRODUCTION

In this paper, we consider the problem of controlled density transport, where given an initial distribution of states specified by a density function, we seek to determine a control sequence to drive this initial distribution to a desired final distribution. We consider the case where a common control signal is applied to the entire distribution of states. This differs from the usual formulation of swarm control and optimal transport problems, where typically each agent can select a control input independently, making the control signal a function of the states and time. This problem of density transport is motivated by problems of manipulation of a large collection of non-interacting agents using a uniform control signal [1]. Such problems have found applications recently in micro-fluidics, where it has been shown that collections of micro-particles can be manipulated through a fluid using a uniform magnetic field for targeted drug delivery [2] or to generate a pumping effect to transport fluid particles [3]. Simultaneously, the transport of density has relevance to the propagation of an uncertainty distribution arising due to uncertainty in the initial state or of a model parameter through an otherwise deterministic control system (see, e.g. [4]–[6]). We formulate and solve this problem using an operator theoretic approach, specifically using the generator of the Perron-Frobenius operator.

The authors are with the Department of Mechanical Engineering, Clemson University, Clemson, SC, USA. jbuzhar@g.clemson.edu, ptallap@clemson.edu

In recent years the operator theoretic approach to dynamical systems has gained significant research attention, both from a dynamical systems perspective [7], [8], as well as in application areas such as control systems [9], [10] and fluid mechanics [11], [12]. A dynamical system can be framed in terms of such an operator either by considering the evolution of observable functions of the state using the Koopman operator or by considering the evolution of densities of states using the Perron-Frobenius operator [13]. The interest in these approaches is primarily due to the fact that these operators allow for a linear, although typically infinite dimensional representation of a nonlinear system. The linearity of these operators is useful from an analytical perspective, as it allows for the use of linear systems techniques such as the analysis of eigenvalues and eigenfunctions, but also from a computational perspective, as in many cases a useful approximation for these operators can be found by considering a finite dimensional approximation in which the operator is represented as a matrix acting on coordinates corresponding to a finite set of a set of dictionary functions [14]–[16].

In applications in control systems, much of the recent work has been on developing methods involving the Koopman operator [9], [10], as the transformation to a space of observable functions can be viewed as a nonlinear change of coordinates which maps the system to a higher dimensional space where the dynamics are (approximately) linear [17]. This makes the numerical approximation of the operator particularly amenable to linear control methods, such as the linear quadratic regulator (LQR) and model predictive control (MPC) [17]–[19]. On the other hand, the Perron-Frobenius operator propagates densities of states forward in time along trajectories of the system, which can have multiple interpretations in the controlled setting. For example, the Perron-Frobenius operator and the Liouville equation, the related PDE formulation, have been used to determine controls for agents in an ensemble or swarm formulation [20]–[22]. It should be noted that such formulations are closely related to optimal transport problems which also involve driving an initial distribution to a desired final distribution (see, e.g., [22]–[24]). Formulations involving the Perron-Frobenius operator have also been used in the context of fluid flows to study the transport of distributions of fluid particles and to detect invariant or almost invariant sets [25], [26].

Our approach involves first obtaining a finite dimensional approximation of the Perron-Frobenius generators associated with the drift and control vector fields of the system, which allow us to represent the density transport dynamics as a

bilinear system in a lifted state. With this system, we frame the density control problem as a problem of driving moments of the density function to the moments of a desired density function, where the moments of the density can be expressed as a output which is linear in the lifted state. This output tracking problem for the lifted bilinear system is then solved using differential dynamic programming (DDP), an iterative trajectory optimization scheme.

II. PRELIMINARIES

Consider first the autonomous dynamical system on a measure space $(\mathbb{X} \subset \mathbb{R}^n, \mathcal{A}, \mu)$ with a σ-algebra $\mathcal{A}$ on $\mathbb{X}$ and μ a measure on $(\mathbb{X}, \mathcal{A})$,

$$\dot{x} = f(x) \tag{1}$$

and denote the associated time-t flow from an initial state x_0 as $\Phi^t(x_0)$, where $x \in \mathbb{X}$ is the state. The Perron-Frobenius operator $\mathcal{P}^t : L^1(\mathbb{X}) \mapsto L^1(\mathbb{X})$ associated with the flow map Φ^t is defined as

$$\int_{\mathbb{A}} \left[\mathcal{P}^t \rho \right](x)\, dx = \int_{(\Phi^t)^{-1}(\mathbb{A})} \rho(x)\, dx \tag{2}$$

for any $\mathbb{A} \in \mathcal{A}$, assuming that the relevant measure μ is absolutely continuous with respect to the Lebesgue measure and can thus be expressed in terms of a density ρ (i.e., $d\mu(x) = \mu(dx) = \rho(x)dx$). It can be shown that the family of these operators $\{\mathcal{P}^t\}_{t \geq 0}$ form a semigroup, (see [13]). The generator of this semigroup is known as the Liouville operator, denoted $\mathcal{L}$, or Perron-Frobenius generator and expresses the deformation of the density ρ under infinitesimal action of the operator $\mathcal{P}^t$ [13], [27]. That is,

$$\frac{d\rho}{dt} = \mathcal{L}\rho = -\nabla_x \cdot (\rho f) \tag{3}$$

Alternatively, the action of the generator can be written in terms of the Perron-Frobenius operator as

$$\mathcal{L}\rho = \lim_{t \to 0} \frac{\mathcal{P}^t \rho - \rho}{t} = \lim_{t \to 0} \left(\frac{\mathcal{P}^t - \mathcal{I}}{t} \right) \rho \tag{4}$$

where $\mathcal{I}$ is the identity operator.

Lemma 1: Suppose the Liouville operator associated with a vector field $f_1 : \mathbb{X} \mapsto \mathbb{R}^n$ is denoted by $\mathcal{L}_1$ and the Liouville operator associated with the vector field $f_2 : \mathbb{X} \mapsto \mathbb{R}^n$ by $\mathcal{L}_2$, then the Liouville operator associated with the vector field $f(x) = f_1(x) + f_2(x)$, is $\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2$.

Proof: The proof is a direct consequence of Eq. 3. Suppose $f(x) = f_1(x) + f_2(x)$. Then $\mathcal{L}\rho = -\nabla_x \cdot (\rho(f_1 + f_2)) = -\nabla_x \cdot (\rho f_1) - \nabla_x \cdot (\rho f_2) = (\mathcal{L}_1 + \mathcal{L}_2)\rho$. ∎

The Koopman operator $\mathcal{K}^t : L^\infty(\mathbb{X}) \mapsto L^\infty(\mathbb{X})$ propagates observable functions forward in time along trajectories of the system and is defined as

$$[\mathcal{K}^t h](x) = [h \circ \Phi^t](x) \tag{5}$$

where $h(x)$ is an observable. The Koopman and Perron-Frobenius operators are adjoint to one another,

$$\int_{\mathbb{X}} [\mathcal{K}^t h](x)\rho(x)dx = \int_{\mathbb{X}} h(x)[\mathcal{P}^t \rho](x)dx \,. \tag{6}$$

III. NUMERICAL APPROXIMATION OF THE PERRON-FROBENIUS OPERATOR AND GENERATOR

Traditionally, the most common way of approximating the Perron-Frobenius operator is by a set-oriented approach known as Ulam's method, in which a domain of interest is discretized into cells and then the operator is computed by simulating a large nubmer of trajectories over a short time and by approximating the operator as the Markov transition matrix which contains the transition probabilities between the cells. It is well known that this can be viewed as a Galerkin projection of the Perron Frobenius operator onto the function space spanned by indicator functions corresponding to the discrete cells [16]. In recent works involving numerical approximation of the Koopman operator, one of the most common approaches is that of extended dynamic mode decomposition (EDMD) [15], in which the operator is computed by solving a least squares problem, which can also be viewed as a Galerkin projection of the operator onto a function space spanned by a predefined set of basis functions [15], [16]. By exploiting the duality of the Perron Frobenius and Koopman operators, it has been shown that methods typically used for one operator can be used to compute the other [16]. Based on this idea, recent works have developed variations of EDMD for the computation of the Perron-Frobenius operator [16], [28], [29]. In this work, we also implement EDMD for the computation of the Perron-Frobenius operator, which we outline below, largely following [16].

We begin by selecting a dictionary $\mathbb{D}$ of k scalar-valued basis functions, $\mathbb{D} = \{\psi_1, \psi_2, \ldots, \psi_k\}$, where $\psi_i : \mathbb{X} \mapsto \mathbb{R}$ for $i = 1, \ldots, k$, and denote by $\mathbb{V}$ the function space spanned by the elements of $\mathbb{D}$. We then collect trajectory data with fixed timestep, Δt, arranged into snapshot matrices as

$$X = \begin{bmatrix} x_1 & , & \cdots & , & x_m \end{bmatrix} \tag{7}$$
$$Y = \begin{bmatrix} x_1^+ & , & \cdots & , & x_m^+ \end{bmatrix} \tag{8}$$

where the subscript $i = 1, \ldots, m$ is a measurement index and $x_i^+ = \Phi^{\Delta t}(x_i)$.

We then approxmiate the observable function h and density ρ in Eq. 6 by their projections onto $\mathbb{V}$, the space spanned by the dictionary $\mathbb{D} = \{\psi_1, \psi_2, \ldots, \psi_k\}$.

$$h(x) \approx \hat{h}^T \Psi(x) \tag{9}$$
$$\rho(x) \approx \Psi^T(x)\hat{\rho} \tag{10}$$

where $\hat{h}, \hat{\rho} \in \mathbb{R}^k$ are column vectors containing the projection coefficients and $\Psi : \mathbb{X} \mapsto \mathbb{R}^k$ is a column-vector valued function where the elements are given by $[\Psi(x)]_i = \psi_i(x)$. Substituting these expansions into Eq. (6), we have

$$\int_{\mathbb{X}} \mathcal{K}^{\Delta t}[\hat{h}^T \Psi]\Psi^T \hat{\rho}\, dx = \int_{\mathbb{X}} \hat{h}^T \Psi \mathcal{P}^{\Delta t}[\Psi^T \hat{\rho}]\, dx \,. \tag{11}$$

Then replacing $[\mathcal{K}^{\Delta t}\Psi](x) = \Psi(x^+)$ and assuming that $\mathcal{P}^{\Delta t}$ can be approximated by a matrix P operating on the coordinates $\hat{\rho}$, and evaluating this on the collected data, we have

$$\Psi(x_i^+)\Psi^T(x_i) = \Psi(x_i)^T \Psi(x_i)P + e_i \tag{12}$$

for $i = 1, \ldots, m$, where e_i is the residual error arising due to the matrix approximation of $\mathcal{P}^{\Delta t}$. This can then be posed as a least-squares problem for the matrix P

$$\min_P \|\Psi_Y \Psi_X^T - \Psi_X \Psi_X^T P\|_2^2 \qquad (13)$$

where $\Psi_X, \Psi_Y \in \mathbb{R}^{k \times m}$ are matrices containing with columns containing Ψ evaluated on the columns of X and Y respectively. The analytical solution of this least squares problem is

$$P = \left(\Psi_X \Psi_X^T\right)^\dagger \Psi_Y \Psi_X^T \qquad (14)$$

where $(\cdot)^\dagger$ is the Moore-Penrose pseudoinverse.

Given this matrix approximation of the operator, P, if the timestep Δt chosen in the data collection is sufficiently small, the corresponding matrix approximation L of the Perron Frobenius generator can be approximated based on the limit definition of the generator in Eq. 4. as

$$L \approx \frac{P - I_k}{\Delta t} \qquad (15)$$

where I_k is the $k \times k$ identity matrix. Just as the matrix operator P approximates the propagation of a density function ρ by advancing the propagation coordinates $\hat{\rho}$ forward for a finite time, the approximation of the generator allows us to approximate the infinitesimal action of the operator $\mathcal{P}^t$ by approximating the time derivative of the projection coordinates

$$\frac{d\hat{\rho}}{dt} = L\hat{\rho}. \qquad (16)$$

A. Extension to controlled systems

In the context of applying the Koopman operator to control systems, several recent works have noted the usefulness of formulating the problem in terms of the Koopman generator, rather than the Koopman operator itself [30]–[34], which typically results in a lifted system that is bilinear in the control and lifted state. This approach allows for a better approximation of the effects of control, especially for systems in control-affine form

$$\dot{x} = f(x) + \sum_{i=1}^{n_c} g_i(x) u_i \qquad (17)$$

as it expresses the effect of the control vector fields g_i in a way that is also dependent on the lifted state. Here we apply a similar approach to the density transport problem, expressed in terms of the Perron-Frobenius generator. As shown in Ref. [32] for the Koopman generator, by the property of the Perron-Frobenius generator given in Lemma 1, if the dynamics are control-affine, then the generators are also control affine, as can be seen by application of Eq. 3. This leads to density transport dynamics of the following form

$$\frac{d}{dt}\rho(x) = (\mathcal{L}_0 \rho)(x) + \sum_{i=1}^{n_c} u_i (\mathcal{B}_i \rho)(x) \qquad (18)$$

where $\mathcal{L}_0$ is the Perron Frobenius generator associated with the vector field $f(x)$ and similarly, the $\mathcal{B}_i$ are the Perron Frobenius generators associated with the control vector fields

$g_i(x)$. Therefore, given the finite dimensional approximation of these generators, we can approximate the density transport dynamics as

$$\frac{d\hat{\rho}}{dt} = L_0 \hat{\rho} + \sum_{i=1}^{n_c} u_i B_i \hat{\rho} \qquad (19)$$

where the matrices L_0 and B_i are the matrix approximations of the operators in Eq. 18.

A similar approximation in terms of the Perron-Frobenius generator has been implemented with control [35], [36] in order to exploit a convex formulation of an optimal control problem in terms of densities.

B. Propagation of moments

In order to make the problem of controlling a density function using a finite dimensional control input well-posed, we formulate the problem as a control problem of a finite number of outputs. In particular, we will describe the density function, in terms of a finite number of its moments. For a scalar x, recall that the ith raw moment, m_i is defined as $m_i = \int_{\mathbb{X}} x^i \rho(x) dx$ and the ith central moment, μ_i about the mean m_1 is $\mu_i = \int_{\mathbb{X}} (x - m_1)^i \rho(x) dx$.

Given a projection of the density as in Eq. 10, the mean is approximated as

$$m_1^i = \int x_i \rho(x) dx = \hat{\rho}^T \int x_i \Psi(x) dx \qquad (20)$$

which simply indicates that the mean of the density can be written as a summation of the means of the dictionary functions, weighted by the projection coefficients. For higher moments, if the central moment is considered, it will be polynomial in the projection coefficients due to its dependence on the mean, whereas the raw moment remain linear in the projection coefficients. For this reason, we choose to work with the raw moments, as the central moments can also be expressed in terms of the raw moments.

In the numerical examples shown later, the dictionary $\mathbb{D}$ consists of Gaussian radial basis functions of the form

$$\psi_l(x) = \exp\left(-\frac{(x - c_l)^T (x - c_l)}{2s^2}\right) \qquad (21)$$

where c_l is the center of the lth basis function, and s is a scaling parameter affecting the spread. For this dictionary, the mean is

$$m_1^i = 2\pi s^2 \hat{\rho}^T c^i \qquad (22)$$

where the superscript i refers to the coordinate index of the state vector, not exponentiation, and c^i is a column vector containing the ith coordinate of the basis function centers. Similarly, the second raw moment can be written as

$$m_2^{ij} = \int x^i x^j \rho(x) dx = \hat{\rho}^T \int x^i x^j \Psi(x) dx \qquad (23)$$

where the last integral reduces to

$$\int x^i x^j \psi_l(x) dx = \begin{cases} 2\pi s^2 (s^2 + (c_l^i)^2) & i = j \\ 2\pi s^2 c_l^i c_l^j & i \neq j \end{cases}$$

for a given basis function $\psi_l(x)$ where, again, superscripts i and j are coordinate indices.
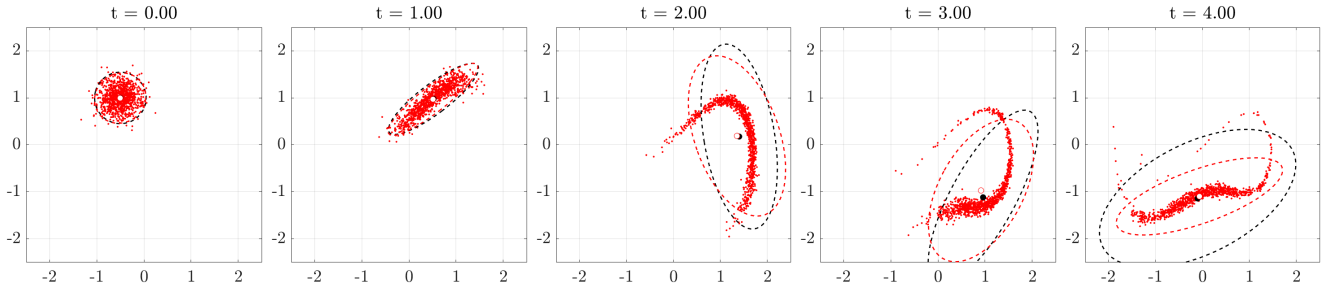
Fig. 1: Moment propagation of proposed method for a Duffing oscillator with sinusoidal forcing. Red points show trajectories from initial conditions sampled from the initial density, $\rho(x(0)) \sim \mathcal{N}([-0.5; 1], 0.05I)$. Red circle and red ellipse show the sample mean and 2σ sample covariance ellipse, respectively. The black circle and black ellipse are the predicted mean and 2σ covariance ellipse.

C. Numerical example

To illustrate the ability of the proposed framework to propagate density functions forward in time, we consider the propagation of an initial density for a forced Duffing oscillator system, given by

$$\frac{d}{dt}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ x_1 - x_1^3 + u \end{pmatrix} \quad (24)$$

where u is the control input. For the purpose of this simulation, we set $u(t) = \sin(4\pi t)$ and the prediction results are shown in Figs. 1 - 2. For the generator calculation, a dictionary of Gaussian radial basis functions, as shown in Eq. 21, is used where the centers lie on an evenly spaced 30×30 grid ranging from -2.5 to 2.5 in x_1 and x_2. The operators are approximated using data collected from short time trajectories with $\Delta t = 0.005$ for a 50×50 grid of initial conditions on the same region. The predicted moment is compared to the sample moment obtained from 1000 trajectories from initial conditions sampled according to the initial density. We see that the moment propagation of the proposed method is good for approximately 3 seconds, which motivates the use of this method in a control formulation, as detailed in the following sections. Also shown for comparison in Fig. 2 is a linear prediction, which is computed by propagating the initial Gaussian through a linearization of Eq. 24, where the linearization is re-computed at each timestep about the predicted mean, as is commonly done in the a priori prediction step of an extended Kalman filter.

IV. CONTROL FORMULATION

We have shown in Sec. III-A and III-B that the problem of steering a density ρ to a desired density can be expressed as an output tracking problem on a lifted, bilinear system given by Eq. 19, where the projection coefficients $\hat{\rho}$ can be interpreted as the lifted state. Then, if the raw moments are taken to be the relevant output, the output, y is linear in the lifted state, $y = C\hat{\rho}$, where the elements of the output matrix C are given by rewriting Eqs. 20, 23 in matrix form.

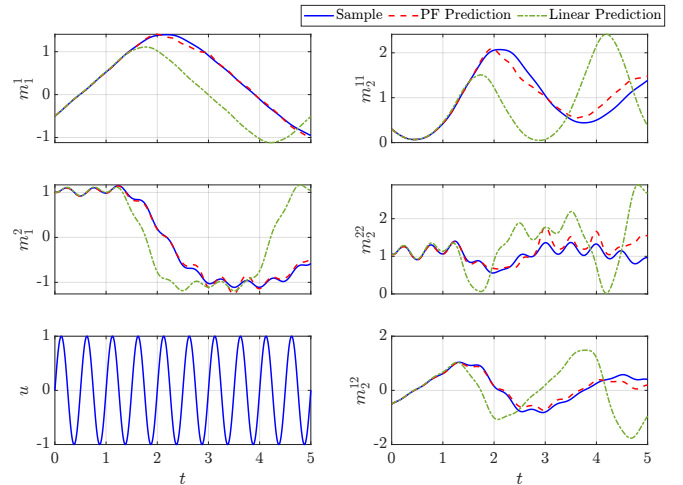For the optimal output tracking problem, we consider a

Fig. 2: Moment propagation for a forced Duffing oscillator. Left, top: First raw moment (mean). Left, bottom: sinusoidal control signal. Right: 2nd raw moment. The proposed method is labelled PF prediction.

discrete time optimal control problem

$$\min_{u_1, u_2, \ldots, u_{H-1}} \sum_{t=1}^{H-1} l(\hat{\rho}_t, u_t) + l_H(\hat{\rho}_H) \quad (25a)$$

$$\text{s.t.} \quad \hat{\rho}_{t+1} = F(\hat{\rho}_t, u_t) \quad (25b)$$

$$y_t = C\hat{\rho}_t \quad (25c)$$

where H is the number of timesteps in the time horizon and Eq. 25b represents the discrete time version of Eq. 19.

In particular, for output tracking, we consider a quadratic cost of the form

$$l(\hat{\rho}_t, u_t) = (y_t - y_t^{\text{ref}})^T S(y_t - y_t^{\text{ref}}) + u_t^T R u_t \quad (26)$$

where S and R are weighting matrices which define the relative penalty on tracking error and control effort, respectively. Since the output y is linear in the lifted state $\hat{\rho}_t$, this cost can be rewritten as a quadratic cost in terms of $\hat{\rho}_t$, with an added linear term.

It is well known that for optimal control problems on bilinear systems with quadratic cost, an effective way of
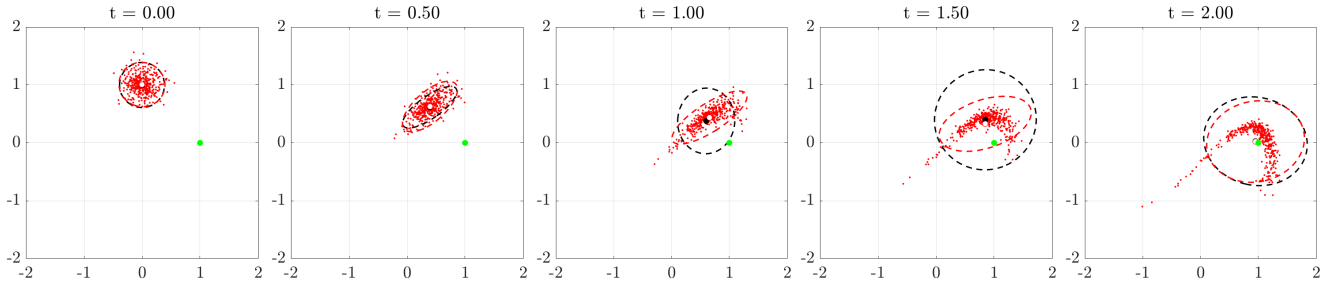
Fig. 3: Controlled transport through the Duffing system from an initial density $\rho(x(0)) \sim \mathcal{N}([0; 1], 0.025I)$ toward the equilibrium point at $(0, 1)$ (green circle). Red circle and red ellipse show the sample mean and 2σ sample covariance ellipse, respectively. The black circle and black ellipse are the predicted mean and 2σ covariance ellipse..

solving the problem is by iteratively linearizing and solving a finite time linear quadratic regulator (LQR) problem about a nominal trajectory, utilizing the Ricatti formulation of that problem [37]. For this reason, we solve the optimal control problem using differential dynamic programming (DDP) [38], [39], which is closely related to the method of iterative LQR. We briefly recount the primary steps of this algorithm below.

DDP computes a locally optimal control around a nominal trajectory by minimizing a quadratic approximation of the value function along this trajectory, and then doing this iteratively about the new trajectories obtained by applying the locally optimal control. First define the value function $V(\hat{\rho}_t, t)$ at time t as,

$$V(\hat{\rho}_t, t) = \min_{u_t}[l(\hat{\rho}_t, u_t) + V(\hat{\rho}_{t+1}, t+1)] \quad (27)$$

which expresses the optimal cost-to-go from $\hat{\rho}_t$, where $V(\hat{\rho}_H, H) = l_f(\hat{\rho}_H)$. Denote by $Q(\delta\hat{\rho}, \delta u)$ the change in the value function due to applying change in control input δu about the nominal trajectory and consider its quadratic approximation

$$Q(\delta\hat{\rho}, \delta u) \approx Q_{\hat{\rho}}\delta\hat{\rho} + Q_u^T\delta u + \delta\hat{\rho}^T Q_{\hat{\rho}u}\delta u \\ + \frac{1}{2}\delta\hat{\rho}^T Q_{\hat{\rho}\hat{\rho}}\delta\hat{\rho} + \frac{1}{2}\delta u^T Q_{uu}\delta u \quad (28)$$

where these derivatives are given by

$$Q_{\hat{\rho}} = l_{\hat{\rho}} + F_{\hat{\rho}}^T V_{\hat{\rho}}'$$
$$Q_u = l_u + F_u^T V_{\hat{\rho}}'$$
$$Q_{\hat{\rho}\hat{\rho}} = l_{\hat{\rho}\hat{\rho}} + F_{\hat{\rho}}^T V_{\hat{\rho}\hat{\rho}}' F_{\hat{\rho}} + V_{\hat{\rho}}' \cdot F_{\hat{\rho}\hat{\rho}}$$
$$Q_{uu} = l_{uu} + F_u^T V_{\hat{\rho}\hat{\rho}}' F_u + V_{\hat{\rho}}' \cdot F_{uu}$$
$$Q_{\hat{\rho}u} = l_{\hat{\rho}u} + F_{\hat{\rho}}^T V_{\hat{\rho}\hat{\rho}}' F_u + V_{\hat{\rho}}' \cdot F_{\hat{\rho}u}$$

where the notation $(\cdot)'$ indicates the next time step. The algorithm proceeds by computing these derivatives by recursing backward in time along the nominal trajectory from the end of the horizon. At each iteration, the control policy is improved by optimizing this quadratic expansion with respect to δu

$$\delta u^* = \arg\min_{\delta u} Q(\delta\hat{\rho}, \delta u) = -Q_{uu}^{-1}(Q_u + Q_{u\hat{\rho}}\delta\hat{\rho}) \quad (29)$$

This can be seen as providing a descent direction in the space of control policies. An updated nominal control is then

computed by a linesearch over a stepsize parameter α to update the policy, that is

$$u_{\text{new}} = u - \alpha Q_{uu}^{-1}Q_u - Q_{uu}^{-1}Q_{u\hat{\rho}}\delta\hat{\rho}$$

and this new control is applied to obtain a new nominal trajectory, and this procedure is iterated until the relative change in cost falls to less than a specified tolerance. For full details of the algorithm, the reader should refer to Refs. [38], [39].

V. VALIDATION AND RESULTS

Here we consider two examples of the density control formulation using Perron Frobenius generators.

A. Example 1: Forced Duffing oscillator

As a first example, we consider the forced Duffing oscillator of Eq. 24, and we use DDP to determine a control sequence to steer a Gaussian initial density $\rho(x(0)) \sim \mathcal{N}([1; 1], 0.025I)$ toward the equilibrium point at $(1, 0)$ over a time horizon of 2s. The generators are approximated using the same data as described in Sec. III-C and the same timestep of 0.005s is used for DDP. In differential dynamic programming, the in-horizon cost weights on the reference error of the first two moments and control effor are all set to unity. The terminal cost on the reference error in the moments is set to 1000, and only the first two moments are considered. The target raw second moment is computed from the desired mean, with the desired variance taken to be zero. The results of this computation are shown in Fig. 3-5.

In Fig. 6, we show a comparison of the performance of the proposed controller, labelled 'PF-DDP' with a standard DDP controller, which computes a control on the Duffing system directly (rather than a lifted state), with the initial condition being the mean. That is, the standard DDP controller acts on the mean as if it were a deterministic initial condition of the system. The comparison shown is the mean and one standard deviation region from trajectories from a set of 500 initial conditions sampled from the initial distribution, with each of the respective controllers applied. We see that the proposed controller moves the mean of the distribution closer to the reference, while maintaining nearly the same variance as the standard DDP controller.
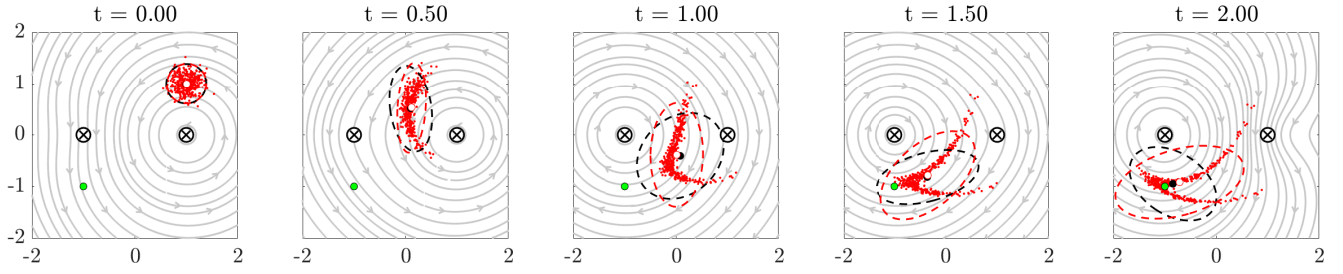
Fig. 4: Controlled transport of fluid particles, driven by two rotlets, or micro-rotors, in a Stokes flow from an initial density, $\rho(x(0)) \sim \mathcal{N}([1;1], 0.025I)$ toward a target mean (green circle) at $(-1, -1)$. The rotors are located at $(-1, 0)$ and $(0, 1)$, as indicated by the black circle-cross. Red circle and red ellipse show the sample mean and 2σ sample covariance ellipse, respectively. The black circle and black ellipse are the predicted mean and 2σ covariance ellipse. Gray streamlines indicate the flow field produced by the rotlets at the instant shown.
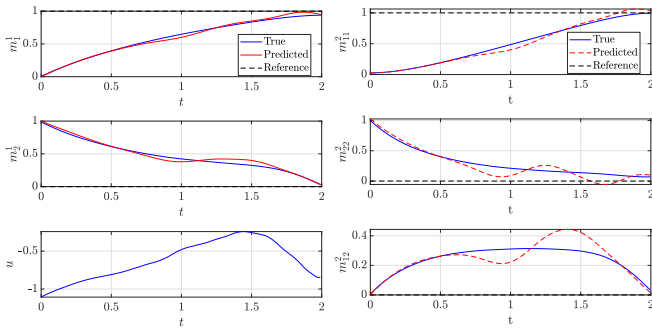

Fig. 5: Control of raw moments for the forced Duffing system. The black line indicates the target (reference). Left, bottom: control from differential dynamic programming for the generator system. Right: 2nd raw moment. Predicted values are from the Perron-Frobenius generator computation. True values are given by the sample moment.

Fig. 7: Streamlines of a rotor driven Stokes flow, where the left rotor torque is $u_1 = -1$ and the right rotor torque is $u_2 = 1$.

this flow is given by

$$\frac{d\mathbf{x}}{dt} = \sum_{i=1}^{N_r} \frac{T_i \times (\mathbf{x} - \bar{\mathbf{x}}_i)}{\|\mathbf{x} - \bar{\mathbf{x}}_i\|^3} \quad (30)$$

where $\bar{\mathbf{x}}_i$ and T_i denote the location and torque of the i-th rotor, respectively. We consider the case where there are two such rotors lying in the x_1-x_2 plane, located at $\bar{\mathbf{x}}_1 = (-1, 0)$ and $\bar{\mathbf{x}}_2 = (1, 0)$ and the controls for the problem are taken to be torques $u_1 = T_1$, $u_2 = T_2$, where the direction of these torques is taken to be normal to the x_1-x_2 plane (in the positive x_3 direction. The streamlines for such a flow are illustrated in Fig. 7 for the case where $u_1 = -1$, $u_2 = 1$.

For this example, the task is to drive a Gaussian initial density $\rho(x(0)) \sim \mathcal{N}([1;1], 0.025I)$ toward a target mean at $(-1, -1)$ over a time horizon of 2s. Again, we consider a timestep of $\Delta t = 0.005$s for both the computation of the generators and for DDP. The in horizon costs weights for the mean error are set to 2, while the weights for the second moment error and control effort are unity. The terminal cost weight on the mean error is 1000 and the terminal cost weight on the second moment error is 500. Results from this computation are shown in Figs. 4,8,9.
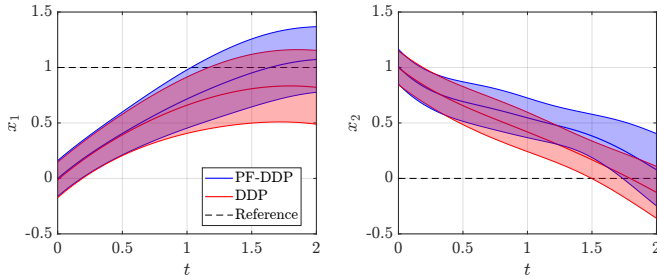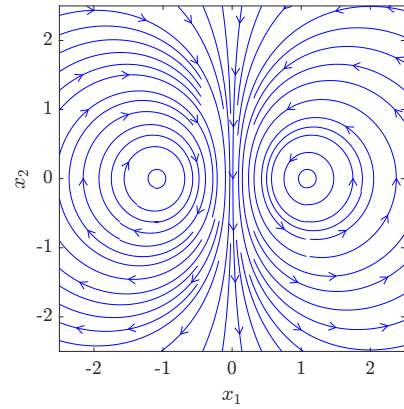

Fig. 6: Comparison of the proposed Perron Frobenius generator DDP with standard DDP for the forced Duffing system. Shaded region indicates one standard deviation

B. Example 2: Rotor-driven Stokes flow

As a second example, we consider the problem of steering a distribution of fluid particles in a Stokes flow, where the flow is produced by two micro-rotors. The micro-rotors are modelled as rotlets, the Stokes flow singularity associated with a point torque in the fluid. For a collection of N_r rotlets,
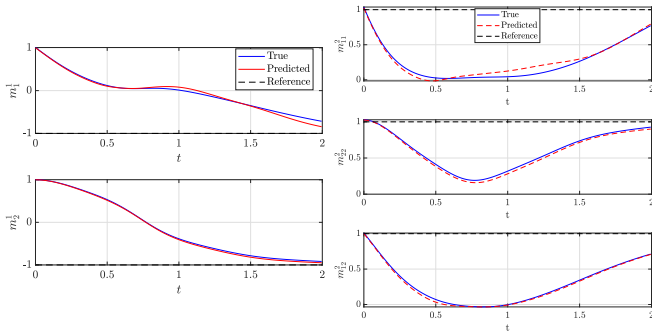
Fig. 8: Control of raw moments for the rotlet-driven Stokes flow. The black line indicates the target (reference). Right: 2nd raw moment. Predicted values are from the Perron-Frobenius generator computation. True values are the sample moment.
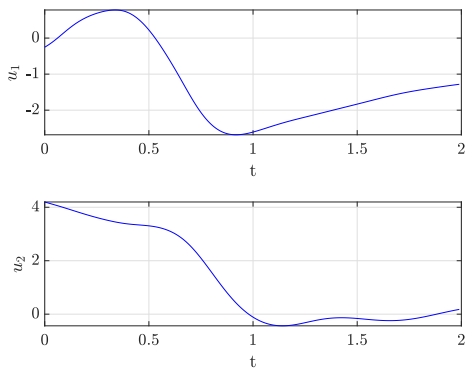

Fig. 9: Control from differential dynamic programming for the generator system applied to the micro-rotor flow.

We see in Fig. 9 that DDP yields a control which drives the mean to the target by first giving a significant counter-clockwise torque on the right rotor to drive the density to a point between the rotors, at which point the left rotor is initiated to drive the flow with a clockwise torque, pulling the density mean near to the target.

This example demonstrates an alternative physical interpretation of the density transport problem, where the density represents a distribution of fluid particles. This also demonstrates the effectiveness of the proposed method on a system which is linear in the controls, but in which the control vector fields are nonlinear.

VI. Conclusion

In this work, we have studied the problem of transporting density functions of states through a controlled dynamical system. This problem formulation has applications both in fluid mechanics and in the control of uncertain systems. Our approach is based on approximations of the Perron-Frobenius operator, whereby we show that approximations of this operator and its generator can be used to model the density transport dynamics as a high-dimensional system which is bilinear in the lifted state and the control. We demonstrated this approach on two examples, a forced Duffing system, in which the density can have the interpretation as an

uncertainty in the initial state and on a rotor driven Stokes flow, in which the density formulation takes on the fluid mechanics interpretation of describing a distribution of fluid particles. Future work in these areas could include extending the proposed control formulation for use in a constrained model predictive control framework for uncertain systems or by studying the fluid transport by more realistic biological microswimmers or artificial microrobots.

References

[1] A. Becker, G. Habibi, J. Werfel, M. Rubenstein, and J. McLurkin, "Massive uniform manipulation: Controlling large populations of simple robots with a common input signal," in *2013 IEEE/RSJ international conference on intelligent robots and systems*, pp. 520–527, IEEE, 2013.

[2] Q. Wang, K. F. Chan, K. Schweizer, X. Du, D. Jin, S. C. H. Yu, B. J. Nelson, and L. Zhang, "Ultrasound doppler-guided real-time navigation of a magnetic microswarm for active endovascular delivery," *Science Advances*, vol. 7, no. 9, p. eabe5914, 2021.

[3] J. K. Hamilton, M. T. Bryan, A. D. Gilbert, F. Y. Ogrin, and T. O. Myers, "A new class of magnetically actuated pumps and valves for microfluidic applications," *Scientific Reports*, vol. 8, no. 1, pp. 1–12, 2018.

[4] A. Mesbah, S. Streif, R. Findeisen, and R. D. Braatz, "Stochastic nonlinear model predictive control with probabilistic constraints," in *2014 American control conference*, pp. 2413–2419, IEEE, 2014.

[5] G. I. Boutselis, Y. Pan, and E. A. Theodorou, "Numerical trajectory optimization for stochastic mechanical systems," *SIAM Journal on scientific computing*, vol. 41, no. 4, pp. A2065–A2087, 2019.

[6] Y. Aoyama, A. D. Saravanos, and E. A. Theodorou, "Receding horizon differential dynamic programming under parametric uncertainty," in *2021 60th IEEE Conference on Decision and Control (CDC)*, pp. 3761–3767, IEEE, 2021.

[7] M. Budišić, R. Mohr, and I. Mezić, "Applied Koopmanism," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 22, no. 4, p. 047510, 2012.

[8] S. L. Brunton, M. Budišić, E. Kaiser, and J. N. Kutz, "Modern Koopman theory for dynamical systems," *SIAM Review*, vol. 64, no. 2, pp. 229–340, 2022.

[9] E. Kaiser, J. N. Kutz, and S. L. Brunton, "Data-driven approximations of dynamical systems operators for control," *The Koopman Operator in Systems and Control: Concepts, Methodologies, and Applications*, pp. 197–234, 2020.

[10] S. E. Otto and C. W. Rowley, "Koopman operators for estimation and control of dynamical systems," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 4, pp. 59–87, 2021.

[11] C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson, "Spectral analysis of nonlinear flows," *Journal of fluid mechanics*, vol. 641, pp. 115–127, 2009.

[12] C. W. Rowley and S. T. Dawson, "Model reduction for flow analysis and control," *Annual Review of Fluid Mechanics*, vol. 49, pp. 387–417, 2017.

[13] A. Lasota and M. C. Mackey, *Chaos, Fractals, and Noise : Stochastic Aspects of Dynamics*. Springer, 1994.

[14] M. Dellnitz and O. Junge, "On the approximation of complicated dynamical behavior," *SIAM Journal on Numerical Analysis*, vol. 36, no. 2, pp. 491–515, 1999.

[15] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A data–driven approximation of the koopman operator: Extending dynamic mode decomposition," *Journal of Nonlinear Science*, vol. 25, pp. 1307–1346, 2015.

[16] S. Klus, P. Koltai, and C. Schütte, "On the numerical approximation of the Perron-Frobenius and Koopman operator," *Journal of Computational Dynamics*, vol. 3, no. 1, pp. 51–79, 2016.

[17] M. Korda and I. Mezić, "Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control," *Automatica*, vol. 93, pp. 149–160, 2018.

[18] J. L. Proctor, S. L. Brunton, and J. N. Kutz, "Generalizing Koopman theory to allow for inputs and control," *SIAM Journal on Applied Dynamical Systems*, vol. 17, no. 1, pp. 909–930, 2018.

[19] M. O. Williams, M. S. Hemati, S. T. Dawson, I. G. Kevrekidis, and C. W. Rowley, "Extending data-driven koopman analysis to actuated systems," *IFAC-PapersOnLine*, vol. 49, no. 18, pp. 704–709, 2016.

[20] R. W. Brockett, "Optimal control of the Liouville equation," *AMS IP Studies in Advanced Mathematics*, vol. 39, p. 23, 2007.

[21] R. W. Brockett, "Notes on the control of the Liouville equation. control of partial differential equations, lecture notes in mathematics," 2012.

[22] K. Elamvazhuthi and P. Grover, "Optimal transport over nonlinear systems via infinitesimal generators on graphs.," *Journal of Computational Dynamics*, vol. 5, 2018.

[23] Y. Chen, T. T. Georgiou, and M. Pavon, "Optimal transport in systems and control," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 4, pp. 89–113, 2021.

[24] A. Halder and R. Bhattacharya, "Geodesic density tracking with applications to data driven modeling," in *2014 American Control Conference*, pp. 616–621, IEEE, 2014.

[25] P. Tallapragada and S. D. Ross, "A set oriented definition of finite-time Lyapunov exponents and coherent sets," *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 5, pp. 1106–1126, 2013.

[26] G. Froyland, N. Santitissadeekorn, and A. Monahan, "Transport in time-dependent dynamical systems: Finite-time coherent sets," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 20, no. 4, p. 043116, 2010.

[27] P. Cvitanovic, R. Artuso, R. Mainieri, G. Tanner, G. Vattay, N. Whelan, and A. Wirzba, "Chaos: classical and quantum," *ChaosBook. org (Niels Bohr Institute, Copenhagen 2005)*, vol. 69, p. 25, 2005.

[28] B. Huang and U. Vaidya, "Data-driven approximation of transfer operators: Naturally structured dynamic mode decomposition," in *2018 Annual American Control Conference (ACC)*, pp. 5659–5664, IEEE, 2018.

[29] D. Goswami, E. Thackray, and D. A. Paley, "Constrained Ulam dynamic mode decomposition: Approximation of the Perron-Frobenius operator for deterministic and stochastic systems," *IEEE control systems letters*, vol. 2, no. 4, pp. 809–814, 2018.

[30] D. Goswami and D. A. Paley, "Global bilinearization and controllability of control-affine nonlinear systems: A koopman spectral approach," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pp. 6107–6112, IEEE, 2017.

[31] S. Klus, F. Nüske, S. Peitz, J.-H. Niemann, C. Clementi, and C. Schütte, "Data-driven approximation of the Koopman generator: Model reduction, system identification, and control," *Physica D: Nonlinear Phenomena*, vol. 406, p. 132416, 2020.

[32] S. Peitz, S. E. Otto, and C. W. Rowley, "Data-driven model predictive control using interpolated koopman generators," *SIAM Journal on Applied Dynamical Systems*, vol. 19, no. 3, pp. 2162–2193, 2020.

[33] D. Bruder, X. Fu, and R. Vasudevan, "Advantages of bilinear koopman realizations for the modeling and control of systems with unknown dynamics," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4369–4376, 2021.

[34] C. Folkestad and J. W. Burdick, "Koopman nmpc: Koopman-based learning and nonlinear model predictive control of control-affine systems," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7350–7356, IEEE, 2021.

[35] B. Huang and U. Vaidya, "A convex approach to data-driven optimal control via Perron–Frobenius and Koopman operators," *IEEE Transactions on Automatic Control*, vol. 67, no. 9, pp. 4778–4785, 2022.

[36] J. Moyalan, H. Choi, Y. Chen, and U. Vaidya, "Data-driven optimal control via linear transfer operators: A convex approach," *Automatica*, vol. 150, p. 110841, 2023.

[37] E. Hofer and B. Tibken, "An iterative method for the finite-time bilinear-quadratic control problem," *Journal of optimization Theory and applications*, vol. 57, pp. 411–427, 1988.

[38] Y. Tassa, T. Erez, and E. Todorov, "Synthesis and stabilization of complex behaviors through online trajectory optimization," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4906–4913, IEEE, 2012.

[39] S. Yakowitz and B. Rutherford, "Computational aspects of discrete-time optimal control," *Applied Mathematics and Computation*, vol. 15, no. 1, pp. 29–45, 1984.