A Continuation Method for Large-Scale Modeling and Control: from ODEs to PDE, a Round Trip

Denis Nikitin, Carlos Canudas de Wit, Paolo Frasca

A Continuation Method for Large-Scale Modeling and Control: from ODEs to PDE, a Round Trip

Denis Nikitin, Carlos Canudas-de-Wit and Paolo Frasca

***Abstract*—In this paper we present a continuation method which transforms spatially distributed ODE systems into continuous PDE. We show that this continuation can be performed both for linear and nonlinear systems, including multidimensional, space- and time-varying systems. When applied to a large-scale network, the continuation provides a PDE describing the evolution of a continuous state approximation that respects the spatial structure of the original ODE. Our method is illustrated by multiple examples including transport equations, Kuramoto equations and heat diffusion equations. As a main example, we perform the continuation of a Newtonian system of interacting particles and obtain the Euler equations for compressible fluids, thereby providing an original solution to Hilbert's 6th problem. Finally, we leverage our derivation of Euler equations to solve a control problem multiagent systems, by designing a nonlinear control algorithm for robot formation based on its continuous approximation.**

***Index Terms*—Control of Large-Scale Networks, Multiagent Systems, PDE**

I. Introduction

MOST of the systems we encounter in real life consist of such a large number of particles that the direct analysis of their interaction is impossible. In such cases, simplified continuous models are used, which aggregate the behavior of large sets of particles and replace them with a continuous representation. In general, discrete and continuous system descriptions share many common properties. For instance, a common theory for discrete and continuous boundary problems was developed in [1] and properties of continuous wave-type oscillatory systems were derived as limit case of discrete systems in [2]. However, even if the discretization procedure transforming PDEs to ODEs is a widely known and widely used method, the inverse problem of transforming an ODE system into PDE is more rarely studied. Our work focuses on this particular problem, with the aim of filling this gap and providing a counterpart to the discretization procedure. This "continuation" procedure can be useful because PDEs provide a much more compact way of describing the system, which can be more convenient than the original ODE system.

The idea of replacing a large system with some compact and simplified representation is widely used, especially for the ODE systems describing large-scale networks. Probably the most known approaches of this type are *model reduction* techniques that transform a network into a smaller one while preserving the properties and the dynamics [3], [4], [5], [6]. Apart model reduction techniques, large-scale networks are studied by *mean field* methods in case of the *all-to-all* interaction topology. In this situation the effect of the network on each node is the same, therefore it is enough to use an equation for a single agent together with parameters of a state of the whole network, see [7] for a review with application to Kuramoto networks. The idea of mean field can be further extended to track not only a single agent's state, but the whole probability distribution over all agents' states in the network. This method is called *population density* or *probability density* approach [8] and it can be used for example to model large biological neural networks [9]. Large-scale systems can be also simplified by studying the approximations to their probability densities, represented by *moments*. E.g., in [10], [11] a moment-based approach was taken to control crowds dynamics. Different applications and issues of the method of moments are covered in [12]. Also *shape parameters* can be used to simplify the model and describe a shape of the solution as in [13].

Mean field and population density approaches are suitable in the case when the interaction topology between nodes is all-to-all. In other cases, the continuous representation of a system requires more sophisticated tools. A recently emerged theory of *graphons* studies graph limits, i.e. structural properties that the graph possesses if the number of nodes tends to infinity while preserving interaction topology. Using graphons it is possible to describe any dense graph as a linear operator on a continuous space [14]. This method was further used to control large-scale linear networks [15] and to study sensitivity of epidemic networks [16]. However, the resulting operator is non-local and requires the original network to have dense connections: for example, in [17] a dense network of Kuramoto oscillators was studied using a continuous representation with integral coupling operator.

In this work we are interested in systems that are spatially distributed and have a position-dependent interaction, such as urban traffic, power networks, formations of autonomous robots, etc. By applying population density method or graphon theory to such system we would end up with a continuous model which loses the spatial structure of the problem. The key idea of our approach is to replace the original spatially distributed ODE system by a continuous PDE whose state and space variables preserve the state and space variables of the original system. We first develop the method for linear spatially invariant ODEs which transforms them into PDEs with the help of finite differences. We name this method as a *continuation*, since it is exactly opposite to the discretization procedure. We prove that the continuation converges to the original system both in the sense of spectrum and in the sense of their trajectories. Using the formalism of computational graphs [18], we then extend the method to nonlinear systems and further to space-dependent systems and systems with boundaries.

D. Nikitin, C. Canudas-de-Wit and P. Frasca are with Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP, GIPSA-lab, 38000 Grenoble, France.

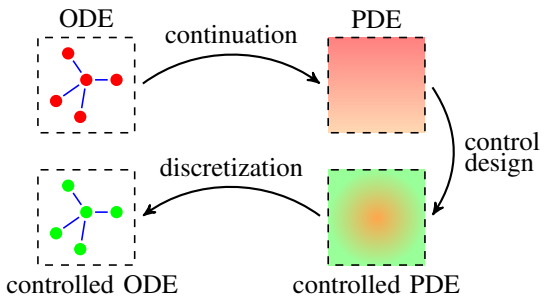

Figure 1. Proposed framework for control design based on the continuation method and a continuous representation of the system.

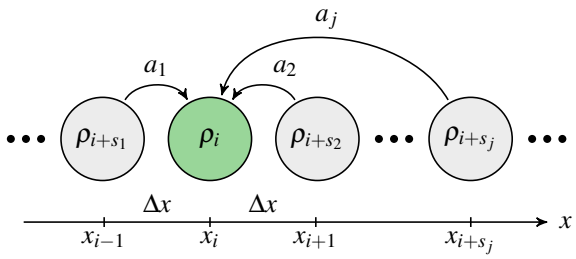

Figure 2. System of nodes aligned in 1D line with dynamics given by (1) with $s_1 = -1$ and $s_2 = 1$.

The idea of substituting finite differences with partial derivatives was already used in several particular applications. In [19] authors derived an LWR traffic PDE model [20] from an equation for a single car, while [21] derived a PDE model for the controlled platooning system. Consensus lattice networks were transformed into PDEs for analysis purposes in [22]. Contrary to these works, we aim at developing a general transformation method, studying its properties, and applying it to different kinds of systems.

The advantage of our proposed continuation method is that it allows to write a PDE which describes the same physical system as the original ODE network. This description can be very helpful for both analysis and control purposes. Indeed, one can use the obtained PDE to design a continuous control which, being discretized back, results in a control law for the original ODE system: Fig. 1 illustrates this design framework.

The continuation method has multiple potential applications. Several of them are being undertaken in the scope of ERC Scale-FreeBack project: in particular, for analysis of linear networks and laser chain stabilization [23], for derivation and control of a general multi-directional 2D urban traffic PDE model [24], [25], and for synchronization analysis for arrays of nonlinear spin-torque oscillators [26].

In this paper, we tackle Hilbert's 6th problem, questioning how one can rigorously transform a system of interacting particles into the Euler PDE. We provide our treatment of this problem using continuation, deriving the Euler PDE from a system of Newton's laws for the case of long-range interaction forces. Further, we show how the method can be applied to multiagent control, providing a simple control algorithm to stabilize a robotic formation along the desired trajectory, performing a maneuver of passing through a window. The control is derived on a level of a PDE representation and then it is discretized to be implemented on every agent in accordance with the scheme in Fig. 1.

We start in Section II by defining a continuation for linear ODEs, discussing questions of accuracy, convergence and choice of the particular model. Section III continues to nonlinear models, utilizing the computational graph formalism. In Section IV the method is extended to much broader class of systems, including multidimensional or space- and time-varying systems and also discussing boundary conditions. Section V is devoted to a derivation of the Euler PDE from a system of interacting particles. Finally, Section VI applies the method to control a robotic formation.

II. Method for linear systems

The simplest class of systems for which the transformation of ODE into PDE can be performed is given by linear ODE systems corresponding to the dynamics of states of nodes, which are aligned on the 1D line in space and depend only on some fixed set of their neighbours. Let the node $i \in \mathbb{Z}$ have a state $\rho_i(t) \in \mathbb{R}$ and a fixed geographical position $x_i \in \mathbb{R}$ such that for every i the distance between two consecutive nodes in space is constant, $x_{i+1} - x_i = \Delta x$ (the assumption of Δx being constant will be relaxed later on). The number of nodes is assumed to be infinite. Then the systems of our interest take the form

$$\dot{\rho}_i(t) = \sum_{j=1}^{N} a_j \rho_{i+s_j}(t), \tag{1}$$

where $\dot{\rho}_i(t)$ denotes time derivative. That is $\dot{\rho}_i(t)$ linearly depends only on N neighbouring nodes shifted by $s_j \in \mathbb{Z}$ for $j \in \{1,...,N\}$, and $a_j \in \mathbb{R}$ are the system gains, see Fig. 2. This type of systems belongs to the class of linear *spatially invariant systems* [27], which is a natural class for distributed control. In the future we will omit writing the dependence on t whenever this is the only argument of ρ.

A. Motivating Example: Transport Equation

We start by considering the most simple ODE system of class (1) which has spatial dependence:

$$\dot{\rho}_i = \frac{1}{\Delta x}(\rho_{i+1} - \rho_i). \tag{2}$$

Comparing with (1), here $N = 2$, $a_1 = 1/\Delta x$, $a_2 = -1/\Delta x$, $s_1 = 1$ and $s_2 = 0$. This equation describes a transport of some quantity along the line, and is usually referred as a Transport ODE. Equation (2) often comes as a result of a discretization process applied to another equation,

$$\frac{\partial \rho}{\partial t}(t,x) = \frac{\partial \rho}{\partial x}(t,x). \tag{3}$$

This equation belongs to a class of PDEs, which is usually thought to be more difficult class of equations to study than ODEs. However, equation (3) describes a perfect transport of information with finite propagation speed along the line, which can be studied much more easily in PDE form than in ODE, as it perfectly conserves the form of a solution, performing

only a shift along the line as time increases. We will refer to this equation as a Transport PDE.

Equation (2) can be obtained from (3) by the discretization process, which has been a well-established mathematical tool. Nevertheless, up to now there was no strict procedure describing a general process which could render equation (3) from (2). In the next subsections we explore more how the discretization procedure is defined for linear systems and how it should be inverted to obtain a continuation process.

B. Discretization

The discretization of PDEs is usually performed by a finite difference method, where the partial derivatives are approximated by finite differences. For example, in the case of Transport ODE $\partial\rho/\partial x \approx (\rho_{i+1}-\rho_i)/\Delta x$. This approximation is valid when Δx is small. Indeed, assuming that the solution to PDE is given by a smooth function $\rho(x)$ and using Taylor series, we can write

$$\rho_{i+1} = \rho(x_{i+1}) = \rho(x_i) + \frac{\partial\rho}{\partial x}\Delta x + \frac{\partial^2\rho}{\partial x^2}\frac{\Delta x^2}{2} + ..., \quad (4)$$

where all partial derivatives are calculated in x_i. Thus, subtracting ρ_i and dividing by Δx, we get

$$\frac{\partial\rho}{\partial x} = \left[\frac{1}{\Delta x}(\rho_{i+1}-\rho_i)\right] - \frac{\partial^2\rho}{\partial x^2}\frac{\Delta x}{2} - ..., \quad (5)$$

which means that the residual belongs to the class $O(\Delta x)$ of all functions which go to zero at least as fast as Δx.

Accuracy can be further increased by taking different *stencil* points where the function is sampled. For example, writing Taylor series for ρ_{i-1} and subtracting it from (4), we get

$$\frac{\partial\rho}{\partial x} = \left[\frac{1}{2\Delta x}(\rho_{i+1}-\rho_{i-1})\right] - \frac{\partial^3\rho}{\partial x^3}\frac{\Delta x^2}{6} + ... \quad (6)$$

Thus, using stencil points $\{i-1, i+1\}$ to approximate the first-order derivative in the point i the obtained residual belongs to the class $O(\Delta x^2)$, which means that this discretization of the Transport PDE is accurate to the second order.

In general, if one wants to approximate a linear combination of derivatives up to order $N-1$ with coefficients $\hat{c}_k$ in point i using N stencil points $\{i+s_1, i+s_2, ..., i+s_N\}$ in form

$$\sum_{k=0}^{N-1}\hat{c}_k\frac{\partial^k\rho}{\partial x^k} \approx \sum_{j=1}^{N}\hat{a}_j\rho_{i+s_j} \quad (7)$$

where coefficients $\hat{a}_j$ are unknown, one can define $S \in \mathbb{R}^{N\times N}$ with elements $S_{jk} = s_j^k$ and $c \in \mathbb{R}^N$ with elements $c_k = \hat{c}_k\, k!/\Delta x^k$ and solve a linear system

$$\hat{a} = S^{-1}\, c. \quad (8)$$

Equation (8) can be trivially obtained by writing Taylor series for all points $\rho_{i+s_1}...\rho_{i+s_N}$ and summing them in a linear combination as in (7).

C. Continuation

Essentially the same process can be applied to the equation (1) to get the PDE version. For every term in the summation in (1) we can write

$$\rho_{i+s_j} = \rho(x_{i+s_j}) = \rho(x_i) + \frac{\partial\rho}{\partial x}\Delta x s_j + \frac{\partial^2\rho}{\partial x^2}\frac{\Delta x^2 s_j^2}{2} + ... \quad (9)$$

Thus, assume we state the problem of finding the PDE approximation of (1) in form

$$\sum_{j=1}^{N} a_j\rho_{i+s_j} \approx \sum_{k=0}^{d} c_k\frac{\Delta x^k}{k!}\frac{\partial^k\rho}{\partial x^k}, \quad (10)$$

where d is the highest order of derivative (*order of continuation*) we want to use. Then the vector of unknown coefficients c can be found by substitution of (9) in (10):

$$c_k = \sum_{j=1}^{N} a_j s_j^k \quad \forall k \in \{0, .., d\}. \quad (11)$$

Once (11) is solved, we write the PDE approximation to (1):

$$\frac{\partial\rho}{\partial t}(t,x) = \sum_{k=0}^{d} c_k\frac{\Delta x^k}{k!}\frac{\partial^k\rho}{\partial x^k}(t,x). \quad (12)$$

As an example, applying (11) to the Transport ODE (2) renders coefficients $c_0 = 0$ and $c_k = 1/\Delta x$ for all $k > 0$, and choosing $d = 1$ we obtain the Transport PDE (3).

Procedures (8) and (11) look very similar from the algebraic point of view, however they are qualitatively different in the way how the problem is formulated and how we should interpret their results. The discretization procedure tries to find the best approximation to a *continuous* function ρ and its derivatives. What is most important, the discretization step Δx is usually an adjustable parameter which can be set by a system engineer *arbitrarily small* to satisfy the desired performance. Instead, when the original system is given by the ODE, the nodes have fixed locations, thus Δx is a *true constant* representing properties of an underlying physical system and it cannot be changed by an engineer.

D. Convergence of continuation

It is clear that the higher order of continuation is taken, the better the original ODE operator (1) is approximated by the PDE (12). It is possible to study the convergence properties by shifting the problem to the frequency domain using the Fourier transform. In the following we first perform a frequency analysis, then derive a bound on solutions' deviation, and finally discuss stability properties of the PDE approximation.

*1) **Spectral analysis**:* For simplicity and without loss of generality assume in this section $\Delta x = 1$. Let us define $a(x)$ as

$$a(x) = \sum_{j=1}^{N} a_j\delta(x+s_j), \quad (13)$$

where $\delta(x)$ is the Dirac delta function. Further, assume that the state $\rho_i(t)$ of (1) was sampled from some integrable function $\rho_i(t) := \rho(t, x_i)$. Then, equation (1) is equivalent to the following system with convolution

$$\frac{\partial\rho}{\partial t}(t,x) = (a \star \rho(t,\cdot))(x). \quad (14)$$

Use now the Fourier transform, defined as

$$\mathscr{F}\{f\}(\omega) = \int_{-\infty}^{\infty} f(x)e^{-ix\omega}dx \tag{15}$$

for any integrable function $f(x)$ and for any frequency $\omega \in \mathbb{R}$. It is known that the Fourier image of a convolution is a multiplication. Therefore the system (14) is just

$$\begin{aligned} \frac{\partial \mathscr{F}\{\rho\}}{\partial t}(t,\omega) &= \mathscr{F}\{a\}(\omega)\mathscr{F}\{\rho\}(t,\omega), \\ \mathscr{F}\{a\}(\omega) &= \sum_{j=1}^{N} a_j e^{is_j\omega}, \end{aligned} \tag{16}$$

where $\mathscr{F}\{a\}(\omega)$ was found by direct calculation of Fourier transform. We can interpret (16) in sense of operator spectrum.

Definition 1. A spectrum of a bounded linear operator T is the set of points $\lambda \in \mathbb{C}$ for which $T - \lambda I$ do not have inverse that is a bounded linear operator, i.e. it is not bijective.

We then introduce a bounded operator T over integrable functions such that $T\rho = (a \star \rho(t,\cdot))(x)$ is a right-hand side of the original ODE system (14). Finding its spectrum is equivalent to finding λ such that for some $v(x)$ there is no solution to $(a\star\rho(t,\cdot))(x) - \lambda\rho(t,x) = v(x)$. Taking Fourier transform one arrives at $(\mathscr{F}\{a\}(\omega) - \lambda)\mathscr{F}\{\rho\}(t,\omega) = \mathscr{F}\{v\}(\omega)$, which clearly has no solution for $\mathscr{F}\{v\}(\omega) \neq 0$ if and only if $\lambda = \mathscr{F}\{a\}(\omega)$ for some ω. Therefore we just showed that the spectrum of T is parametrized by the image of $\mathscr{F}\{a\}(\omega)$. In fact, this result is well-known, since the system (1) on an infinite line belongs to the class of Laurent systems, whose spectrum is known to be (16), see [28].

Let us calculate the spectrum of the right-hand side of the continualized system (12). Denote the state of (12) as $\rho^c(t,x)$. If $\rho^c(t,x)$ is sufficiently smooth and its derivatives are integrable, we can recover their Fourier images by

$$\mathscr{F}\left\{\frac{\partial^k \rho^c}{\partial x^k}\right\}(t,\omega) = (i\omega)^k \mathscr{F}\{\rho^c\}(t,\omega).$$

Therefore (12) is read in frequency domain as

$$\begin{aligned} \frac{\partial \mathscr{F}\{\rho^c\}}{\partial t}(t,\omega) &= \mathscr{F}\{c\}(\omega)\mathscr{F}\{\rho^c\}(t,\omega), \\ \mathscr{F}\{c\}(\omega) &= \sum_{k=0}^{d} c_k \frac{1}{k!}(i\omega)^k. \end{aligned} \tag{17}$$

Substituting (11), we can rewrite $\mathscr{F}\{c\}(\omega)$ in (17) as

$$\mathscr{F}\{c\}(\omega) = \sum_{j=1}^{N} a_j \sum_{k=0}^{d} \frac{(is_j\omega)^k}{k!}. \tag{18}$$

Now, comparing (18) with (16), it is clear that (18) uses the first $d+1$ terms of the Taylor expansion of the exponential in (16). Since the exponential function is analytic on the whole complex plane, we have just proven the following result:

Theorem 1. *The spectrum of the PDE operator* (12) *converges to the spectrum of the original ODE operator* (1) *pointwise as* $d \to \infty$.

*2) **Convergence of solutions**:* Define now a Discrete-Time Fourier Transform (or DTFT, although taken along the coordinate axis) for an infinite sequence f_n for $n \in \mathbb{Z}$ as

$$\mathscr{D}\{f\}(\omega) = \sum_{n=-\infty}^{+\infty} f_n e^{-in\omega}, \quad \omega \in [-\pi,\pi]. \tag{19}$$

We can use Theorem 1 to prove that the sampled trajectory of the PDE converges to the solution of the ODE as d increases:

Theorem 2. *Let* $\tilde{\rho}_i(t) := \rho_i(t) - \rho^c(t,x_i)$ *be a deviation between the original and the continualized systems' solutions at the nodes' positions. Assume that at initial time*

$$\begin{aligned} \mathscr{D}\{\rho\}(0,\omega) &= \mathscr{F}\{\rho^c\}(0,\omega) \quad \forall|\omega| \leqslant \pi, \\ 0 &= \mathscr{F}\{\rho^c\}(0,\omega) \quad \forall|\omega| > \pi, \end{aligned} \tag{20}$$

which defines the initial state of the PDE with respect to the original ODE. Then for $\forall t \geqslant 0$

$$\begin{gathered} \|\tilde{\rho}(t)\|_{l^2} \leqslant e^{\operatorname{Re}\lambda_{max} t}\left(e^{\gamma_d t} - 1\right)\|\rho(0)\|_{l^2}, \quad \textit{where} \\ \operatorname{Re}\lambda_{max} = \max_{|\omega|\leqslant\pi} \operatorname{Re}\mathscr{F}\{a\}(\omega), \quad \gamma_d = \sum_{j=1}^{N} |a_j| \frac{|\pi s_j|^{d+1}}{(d+1)!} e^{|\pi s_j|}. \end{gathered} \tag{21}$$

In particular, $\|\tilde{\rho}(t)\|_{l^2} \to 0$ *as* $d \to \infty$.

Proof. First of all, by definition (19) of DTFT it is clear that $\mathscr{D}\{a\}(\omega) \equiv \mathscr{F}\{a\}(\omega)$, where the former is taken for the sequence a_i in (1) and the latter is taken for the function $a(x)$ in (13). Since the right-hand side of (1) represents a convolution of $\rho_i(t)$ with the sequence a_i, we can use this equality and write the evolution of the DTFT image of ρ_i as

$$\frac{\partial \mathscr{D}\{\rho\}}{\partial t}(t,\omega) = \mathscr{F}\{a\}(\omega)\mathscr{D}\{\rho\}(t,\omega). \tag{22}$$

Further it is easy to show (see e.g. [29]) that the sampling of $\rho^c(x,t)$ induces *periodization* on its Fourier image:

$$\mathscr{D}\{\rho^c(t,x_i)\}(\omega) = \sum_{n=-\infty}^{+\infty} \mathscr{F}\{\rho^c(t,x)\}(\omega + 2\pi n).$$

By (20) and by (17) $\mathscr{F}\{\rho^c\}(t,\omega) = 0$ for $|\omega| > \pi, t \geqslant 0$, which means that $\mathscr{D}\{\rho^c(t,x_i)\}(\omega) \equiv \mathscr{F}\{\rho^c(t,x)\}(\omega)$ for $|\omega| \leqslant \pi$.

Therefore, the DTFT of $\tilde{\rho}_i(t)$ is

$$\mathscr{D}\{\tilde{\rho}\}(t,\omega) := \mathscr{D}\{\rho\}(t,\omega) - \mathscr{F}\{\rho^c\}(t,\omega) \quad \forall\omega \in [-\pi,\pi].$$

Let us now use Parseval's identity for DTFT, see [28]:

$$\|\tilde{\rho}(t)\|_{l^2}^2 = \sum_{i=-\infty}^{+\infty} |\tilde{\rho}_i|^2(t) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \left|\mathscr{D}\{\tilde{\rho}\}(t,\omega)\right|^2 d\omega. \tag{23}$$

The integral is taken over the bounded interval of frequencies since the transformed sequence is discrete.

One can now notice that (17) and (22) are just scalar linear time-invariant ODEs for each ω, thus it is possible to write their explicit solutions as

$$\begin{aligned} \mathscr{D}\{\rho\}(t,\omega) &= e^{\mathscr{F}\{a\}(\omega)t}\mathscr{D}\{\rho\}(0,\omega), \\ \mathscr{F}\{\rho^c\}(t,\omega) &= e^{\mathscr{F}\{c\}(\omega)t}\mathscr{F}\{\rho^c\}(0,\omega). \end{aligned}$$

Using the condition (20) on initial conditions $\mathscr{D}\{\rho\}(0,\omega) = \mathscr{F}\{\rho^c\}(0,\omega)\ \forall|\omega| \leqslant \pi$ we write the Fourier image of $\tilde{\rho}_i(t)$:

$$\begin{aligned}\mathscr{D}\{\tilde{\rho}\}(t,\omega) &= \left(e^{\mathscr{F}\{a\}(\omega)t} - e^{\mathscr{F}\{c\}(\omega)t}\right)\mathscr{D}\{\rho\}(0,\omega) = \\ &= e^{\mathscr{F}\{a\}(\omega)t}\left(1 - e^{(\mathscr{F}\{c\}(\omega)-\mathscr{F}\{a\}(\omega))t}\right)\mathscr{D}\{\rho\}(0,\omega)\end{aligned} \tag{24}$$

which holds $\forall|\omega| \leqslant \pi$. Inserting (24) in (23) and using Hölder's inequality we get

$$\begin{aligned}\|\tilde{\rho}(t)\|_{l^2}^2 \leqslant \frac{1}{2\pi}\int_{-\pi}^{\pi}\left|\mathscr{D}\{\rho\}(0,\omega)\right|^2 d\omega \times \\ \times \max_{|\omega|\leqslant\pi}\left|e^{\mathscr{F}\{a\}(\omega)t}\right|^2 \times \max_{|\omega|\leqslant\pi}\left|1 - e^{(\mathscr{F}\{c\}(\omega)-\mathscr{F}\{a\}(\omega))t}\right|^2.\end{aligned} \tag{25}$$

The first term is $\|\rho(0)\|_{l^2}^2$. Further, $\max_{|\omega|\leqslant\pi}\left|e^{\mathscr{F}\{a\}(\omega)t}\right|^2 = e^{2\operatorname{Re}\lambda_{max}t}$. Thus we will concentrate on the third multiplier.

Let $z = u + iv$ be any complex number. Then by [30]-(3.8.23) we can write $|e^z - 1|^2 \leqslant \left(e^{|z|} - 1\right)^2$. This bound increases with respect to $|z|$, therefore

$$\max_{|\omega|\leqslant\pi}\left|1 - e^{(\mathscr{F}\{c\}(\omega)-\mathscr{F}\{a\}(\omega))t}\right|^2 \leqslant \left(e^{\gamma t} - 1\right)^2$$

for any $\gamma \geqslant \max_{|\omega|\leqslant\pi}|\mathscr{F}\{c\}(\omega) - \mathscr{F}\{a\}(\omega)|$. We can find the lowest bound on γ denoted as γ_d using the definitions of $\mathscr{F}\{a\}(\omega)$ and $\mathscr{F}\{c\}(\omega)$ in (16) and (18). Namely,

$$\begin{aligned}|\mathscr{F}\{c\}(\omega) - \mathscr{F}\{a\}(\omega)| = \left|\sum_{j=1}^{N} a_j \sum_{k=d+1}^{+\infty}\frac{(is_j\omega)^k}{k!}\right| \leqslant \\ \leqslant \sum_{j=1}^{N}|a_j|\sum_{k=d+1}^{+\infty}\frac{|s_j\omega|^k}{k!} \leqslant \sum_{j=1}^{N}|a_j|\frac{|s_j\omega|^{d+1}}{(d+1)!}\sum_{k=0}^{+\infty}\frac{|s_j\omega|^k}{k!},\end{aligned} \tag{26}$$

and the last summation is just $e^{|s_j\omega|}$. Finally, since (26) increases with $|\omega|$, we can substitute the maximal value $|\omega| = \pi$ and thus obtain γ_d as in (21). Finally the bound (21) is recovered by taking square root of (25).

The final statement of the theorem can be proven if one notices that $\gamma_d \to 0$ as $d \to \infty$, which leads to $(e^{\gamma_d t} - 1) \to 0$ as $d \to \infty$ for any fixed $t \geqslant 0$. □

Remark 1. Condition (20) means that the continuous system should be initialized with the low-frequency continuation of the original ODE initial state. Note that this can always be done since (20) uniquely determines the Fourier image of $\rho^c(0,x)$. For example an initial state $\rho_0 = 1$ and $\rho_i = 0$ for $i \neq 0$ results in $\mathscr{D}\{\rho\}(\omega) \equiv 1$ which by (20) sets $\rho^c(0,x) = \operatorname{sinc}(\pi x)$. Moreover, bound (21) at $t = 0$ ensures that $\tilde{\rho}_i(0) \equiv 0$, therefore the continuation coincides with the ODE initial state.

*3) **Stability analysis**:* We can now turn to the discussion of stability of the PDE. Due to the simple nature of scalar equations (16) and (17) we can say that the system (16) is *stable* if and only if $\operatorname{Re}\mathscr{F}\{a\}(\omega) \leqslant 0\ \forall\omega \in \mathbb{R}$, otherwise it is *unstable*. Note, this definition would not hold in a more general case of vector-valued states ρ_i, but in (1) a scalar node state is assumed. A simple corollary of Theorem 2 can be derived:

Corollary 1. *If* $\operatorname{Re}\mathscr{F}\{a\}(\omega) \leqslant \operatorname{Re}\lambda_{max} < 0\ \forall\omega \in \mathbb{R}$, *then* $\|\tilde{\rho}(t)\|_{l^2} \to 0$ *as* $t \to \infty$ *for all high enough* d.

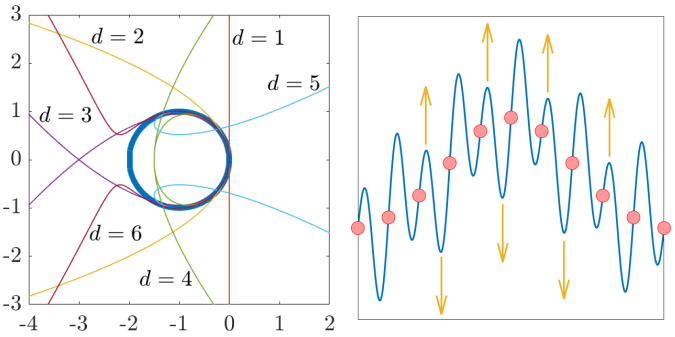

Figure 3. **Left:** spectrum for the Transport ODE (27) $e^{i\omega} - 1$ (blue circle) together with spectrums of the continuations up to the order 6, including (28). As d increases, spectrums converge to the blue circle, however for some orders (such as 4 or 5) they can become unstable. **Right:** Schematic picture of an artificial instability for high order d. Although the continuation (blue) coincides with the original solution (red) at the nodes' positions, high-frequency components can be unstable.

Proof. Indeed, for high enough d we have $\gamma_d < -\operatorname{Re}\lambda_{max}$, which means that (21) is bounded by an exponential $e^{(\operatorname{Re}\lambda_{max}+\gamma_d)t} \to 0$ as $t \to \infty$. □

Note that although Theorem 2 states the convergence of sampled trajectories, in Theorem 1 the convergence of spectrums is not uniform. Moreover, the spectrum (16) is an image of the unit circle and thus is a compact set, while the spectrum (18) for any d is a polynomial and thus unbounded. This can lead to an undesirable effect which we call an *artificial instability*, meaning that the tails of the image of the polynomial (18) happen to lie in the positive complex half-plane, as in the left panel of Fig. 3 for $d = 4$ or 5. Essentially this means that the PDE becomes unstable on high frequencies, see the right panel of Fig. 3. We can though induce several corollaries from Theorems 1 and 2 which can help in understanding stability properties of the obtained PDE.

Corollary 2. *If the original ODE* (1) *is unstable, there exists* $D \geqslant 0$ *such that for all* $d \geqslant D$ *the continualized system* (12) *will also be unstable.*

Proof. Since the original system is unstable, there exists ω_0 such that $\operatorname{Re}\mathscr{F}\{a\}(\omega_0) > 0$. Now, by Theorem 1 there exists $D \geqslant 0$ such that for all $d \geqslant D$ $\operatorname{Re}\mathscr{F}\{c\}(\omega_0) > 0$. □

Corollary 3. *PDE* (12) *with an odd order of continuation* d *has the same stability properties as a PDE with the order of continuation* $d - 1$.

Proof. All odd terms in the spectrum (18) are purely imaginary and thus have no impact on the stability. □

Corollary 4. *Artificial instability is introduced when the last even term in the PDE* (12) *has* $c_k > 0$ *if* $k = 4m$ *or* $c_k < 0$ *if* $k = 4m + 2$ *for some* $m \in \mathbb{Z}^+$.

Proof. Artificial instability comes if the term of the polynomial (18) with the highest even power is positive, which leads to a positive real part of the spectrum on high frequencies. Positivity of the highest even term is equivalent to the statement of the corollary since $i^{4m} = 1$ and $i^{4m+2} = -1$ for $m \in \mathbb{Z}^+$. □

We show the spectrum convergence on the Transport ODE

$$\dot{\rho}_i = \rho_{i+1} - \rho_i. \tag{27}$$

With $\Delta x = 1$, the continuation of (27) is:

$$\frac{\partial \rho}{\partial t}(t,x) = \sum_{k=1}^{d} \frac{1}{k!} \frac{\partial^k \rho}{\partial x^k}(t,x), \tag{28}$$

Spectrum of (27) equals $e^{i\omega} - 1$ by (16), which is depicted as a blue circle in the left panel of Fig. 3 together with the spectrums of the continuations up to the order $d = 6$. It is clear that as the order increases, the approximations become better.

The original Transport ODE is stable. Moreover, it has an intrinsic diffusion in it, which can be captured by the continuation of the second order. However, the continuations of orders 4 and 5 are unstable. It happens because of an artificial instability as described in Corollary 4, since $c_4 = 1 > 0$. In general all stable continuations of the Transport ODE are given by the orders $\{1,2,3,...,4m+2,4m+3,...\}$ for all $m \in \mathbb{Z}^+$.

Theorems 1 and 2 say that increasing order of continuation leads to the more correct capture of the behavior of the original ODE. From the practical point of view, however, low-order PDEs capture low frequency effects very well, while high orders can cause artificial instability. Moreover, lack of tools for control and analysis of high-order PDEs makes impractical their derivation. Therefore it usually makes sense to stick to the orders $d = 1$ or $d = 2$, which are chosen in [19], [23], [25] and which will be used in examples throughout this paper.

III. Method for nonlinear systems

Finite differences give us a complete tool for linear systems, but for nonlinear systems they should be applied in composition with nonlinearities. Using an additional concept of computational graph it is possible to elaborate the case of general nonlinear ODE systems.

As in the previous case we assume without loss of generality that the nodes are equally spaced along the 1D line, a node i having a state ρ_i and a position x_i. Then the general nonlinear ODE with space dependence takes form of

$$\dot{\rho}_i = F(\rho_{i+s_1}, \rho_{i+s_2}, ..., \rho_{i+s_N}). \tag{29}$$

We further assume that the function F is continuous.

A. Computational graph

In 1957 Kolmogorov [31] showed that every multidimensional continuous function can be written as a composition of functions of one variable and additions. This work laid the basis for the neural networks function approximation.

Here we will use this idea and assume that the function F is given in the form of computational graph (see [18] for example). This is a directed acyclic graph, every node of which represents a one-dimensional function, applied to a weighted sum of inputs coming to this node. We assume that the leaves of this graph are the states of the system ρ_{i+s_j} and the root node computes the resulting value of F.

As an example of the computational graph consider

$$\dot{\rho}_i = \sin(\rho_{i+1} - \rho_i) - \sin(\rho_i - \rho_{i-1}) \tag{30}$$

which is a system of Kuramoto oscillators coupled on a ring. The computational graph for (30) is presented in Fig. 4.

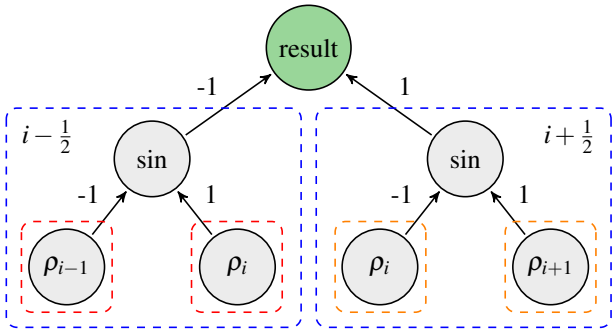

Figure 4. Computation graph for the system (30). Similar subgraphs are outlined by dashed rectangles of the same color. Possible choices of sinus subgraph' positions are written in the corners of blue rectangles.

B. Similar subgraphs and their positions

Now let us introduce an original notion of *similar subgraphs*. Subgraph is a computational graph which computes subexpression of the original computational graph. Every node in a computational graph serves as the root of a subgraph computing expression defined in this node. The leaf nodes are also the subgraphs "computing" themselves.

Definition 2. We call two subgraphs *similar* if

1) they serve as an input to the same node,
2) they differ only in the positions of the leaf nodes, and this difference can be represented by a single shift.

This is an equivalence relation, therefore we can speak about equivalence classes which we call sets of similar subgraphs.

For example, in Fig. 4 there are three sets of similar subgraphs:

1) ρ_{i-1} and ρ_i for the left sinus node,
2) ρ_i and ρ_{i+1} for the right sinus node,
3) $\sin(\rho_i - \rho_{i-1})$ and $\sin(\rho_{i+1} - \rho_i)$ for the root node, because they differ by a single shift which equals 1.

Finally we will define a *position* of a subgraph:

Definition 3. Position of a subgraph is defined as a coordinate in space where the expression of this subgraph is calculated.

The leaf nodes by definition are the states of the system, thus their positions are uniquely specified. For example for the leaf node ρ_{i+1} in Fig. 4 we say that its position is $i+1$.

The root node by definition has a position i, since it is exactly the position of the left-hand side term in (29). We will define the positions of other subgraphs as the average of their leaves' positions. Note that in general there is some freedom in the definition of the subgraphs' positions, with the only constraint that similar subgraphs should differ by a single shift, but we will omit this for simplicity.

Since the position of a subgraph represents a position on the line, it is natural to have non-integer position values, although the leaf nodes and the root have only integer positions. As an example, defining the position as an average, in Fig. 4 the node $\sin(\rho_{i+1} - \rho_i)$ has its position $i + 1/2$.

C. Continuation to a nonlinear PDE

When system (29) is expressed in a form of computational graph with similar subgraphs being found and their positions being defined, one can perform a continuation procedure described in section II-C to obtain a PDE.

Continuation should be performed recursively, starting from the leaves. Each set of similar subgraphs by definition is used in their common ancestor node as a linear combination of equivalent elements shifted by some distance. Continuation of this linear combination by (10) replaces a set of similar subgraphs by a weighted sum of partial derivatives of subexpressions, calculated at the position of the ancestor node.

Let $\Delta x = x_{i+1} - x_i$ be a distance between two neighbouring nodes. Elaborating example (30) and using $d = 1$ for each set of subgraphs, we perform the continuation in three steps:

$$\begin{aligned}
&1)\ \sin(\rho_{i+1} - \rho_i) \quad \to \quad \sin\left(\Delta x \frac{\partial \rho}{\partial x}(x_{i+1/2})\right),\\
&2)\ \sin(\rho_i - \rho_{i-1}) \quad \to \quad \sin\left(\Delta x \frac{\partial \rho}{\partial x}(x_{i-1/2})\right),\\
&3)\ \sin_{i+1/2} - \sin_{i-1/2} \quad \to \quad \Delta x \frac{\partial}{\partial x} \sin.
\end{aligned}$$

which finally gives a nonlinear PDE representation of (30):

$$\frac{\partial \rho}{\partial t}(t,x) = \Delta x \frac{\partial}{\partial x} \sin\left(\Delta x \frac{\partial \rho}{\partial x}(t,x)\right). \tag{31}$$

To obtain higher-order PDE approximations it makes sense to specify the desired order of the equation d and then get rid of all the terms which consist of composition of derivatives of combined order higher than d.

IV. Extensions

Until now we discussed systems with nodes which were uniformly placed on the infinite 1D line and which had common space-independent dynamics. The method can be extended to include more classes of systems.

- Periodic spatially invariant systems [27] can be tackled by choosing different index spaces. We can assume that the positions $x \in \mathbb{S}$ belong to the unit circle and indices $i \in \mathbb{Z} \setminus n\mathbb{Z}$ form a ring of integers modulo n, where n is the number of states of the original ODE. Since any function on $\mathbb{S}$ can be mapped to a periodic function on $\mathbb{R}$, the analysis in Sections II and III remain the same.
- Multidimensional systems can be accounted for by assuming that a position of a node ρ_i is described by $x_i \in \mathbb{R}^n$ and taking multidimensional Taylor expansion at (9).
- Time dependence can be introduced into system gains both for ODEs and for PDEs with continuation being performed independently for all t. This allows to use the method for time-varying systems and switching networks.

In the following subsections we will explore how the method can be extended to include systems with space dependence or nonuniform placing and systems with boundaries. Further we introduce a concept of PDE with index derivatives which can be applied to systems whose states coincide with the positions in space, for example particle systems. Finally, all kinds of systems are covered by the general continuation algorithm presented in the end of this section.

A. Space-dependent and non-uniform systems

Let us now look at the linear system (1) with one important difference: the system gains a_j, the shifts s_j and the number of neighbours N become space-dependent:

$$\dot{\rho}_i = \sum_{j=1}^{N_i} a_{ij} \rho_{i+s_{ij}}. \tag{32}$$

Notice that equation (32) describes in fact any linear system.

Now one can perform a continuation (11) at every point x_i up to the order d and obtain a PDE (12) with space dependent gains c_{ik}. This means that we know the gains c_{ik} at the points with coordinates x_i, which can be seen as a sampling of some function $c_k(x)$ at points x_i.

Non-uniform placing of nodes can be tackled in the same way. Indeed, assuming distance $x_{i+1} - x_i$ can be arbitrary, continuation can be performed by defining $c_{ik} = \sum_{j=1}^{N_i} a_{ij}(x_{i+s_{ij}} - x_i)^k$ instead of (11).

We can now perform either an interpolation or an approximation based on this sampling. In the first case we seek for $c_k(x)$ such that $c_k(x_i) = c_{ik}$, while in the second case it is enough to satisfy this relation approximately. In either case, the resulting continuation of (32) is given by

$$\frac{\partial \rho}{\partial t}(t,x) = \sum_{k=1}^{d} c_k(x) \frac{1}{k!} \frac{\partial^k \rho}{\partial x^k}(t,x). \tag{33}$$

For nonlinear systems the continuation can be performed if computational graphs for every node compute the same dynamics. We can formalize it with the following property:

Definition 4. We say that two computational graphs *have the same structure* if

1) their root nodes compute the same expression,
2) any child subgraph of the root node of the first graph *has the same structure* with some child subgraph of the root node of the second graph and vice versa.

This definition, formulated through recursion, essentially means that the order of nonlinearities which is hidden in two computational graphs should coincide, see Fig. 5.

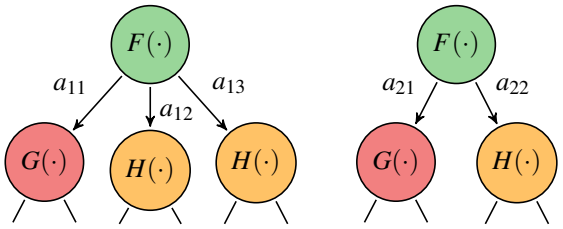

Figure 5. Illustration of two computational graphs having the same structure.

Finally, a continuation of a nonlinear ODE system can be performed if all the computational graphs computing the dynamics for all states ρ_i have the same structure. Indeed, in this case it is possible to perform a continuation for any set of similar subgraphs for each node as in the linear case of (32)-(33). Moreover, by Definition 4 these sets of similar subgraphs for different positions serve as inputs to the same nonlinearities, therefore a unique PDE with space-dependent coefficients can be obtained.

Remark 2. In theory, it is possible to satisfy Definition 4 for any nonlinear system formulated through computational graphs. Indeed, assume two computational graphs have two different root node expressions, denoted as $F(\cdot)$ and $G(\cdot)$ respectively. Then we can artificially create a new common root node which will compute $1 \cdot F(\cdot) + 0 \cdot G(\cdot)$ for the first graph and $0 \cdot F(\cdot) + 1 \cdot G(\cdot)$ for the second. Thus we can satisfy the first condition of Definition 4, and recursively applying this idea one can transform any pair of computational graphs into the pair which has the same structure. However, if the computational graphs of the system are too different in different points, it can make no sense to represent a system as a PDE, since it means that the dynamics of different parts of the system has nothing in common.

B. Boundary conditions

Imagine there is a Heat PDE defined on an interval $x \in [0, +\infty)$, that is there is a boundary in the point $x = 0$. Assume also a Dirichlet boundary condition is supplied, namely the state on the boundary is fixed to some $a \in \mathbb{R}$:

$$\frac{\partial \rho}{\partial t}(t,x) = \frac{\partial^2 \rho}{\partial x^2}(t,x), \quad \rho(t,0) = a. \tag{34}$$

If the Heat Equation (34) is discretized in stencil points $\{i-1, i, i+1\}$, the result is

$$\dot{\rho}_i = \frac{1}{\Delta x^2}\left(\rho_{i-1} - 2\rho_i + \rho_{i+1}\right). \tag{35}$$

Assume that there exists $i_0 = 1$ such that $x_{i_0-1} = 0$. The equation for the state ρ_1 can be obtained by the discretization of the boundary value problem (34):

$$\dot{\rho}_1 = \left(a - 2\rho_1 + \rho_2\right)/\Delta x^2, \tag{36}$$

Now imagine the system (34) is obtained by the continuation process from the system (35). We can notice that states of (35) are governed by the same dynamics except for the boundary state ρ_1. The question is how to recover the boundary conditions (34) of the PDE from the dynamics of ρ_1 in (36).

This indeed can be done if one assumes that there exists a "ghost cell" ρ_0 such that it has no dynamics, but is algebraically connected with adjacent states. With a proper definition of ρ_0 the equation for $\dot{\rho}_1$ can be represented in the same way as for other states (35) and thus has the same continuation (34). For example, the algebraic equation for ρ_0 representing (35)-(36) is $\rho_0 = a$, which can be directly continualized, obtaining the boundary condition in (34).

This procedure can be generalized to any ODE system and any type of boundaries: once the states near the boundaries change their dynamics with respect to the general governing equation, this change can be represented by "ghost cells" with algebraic dependences on the "real" states. Continualizing these algebraic equations leads to the boundary conditions for the obtained PDE.

C. PDE with index derivatives

Usually PDEs have derivatives written with respect to the time and space variables, thus their physical meaning is in the function continuously varying in time and space. However, in general no one prevents us from writing a PDE with respect to some other variables.

Assume a physical system is given by a set of interacting agents, with agents being indexed by an integer index $i \in \mathbb{Z}$ (a general multiindex space $\mathbb{Z}^n$ can also be used). Let an agent i have a state ρ_i. The index variable i is by definition discrete. However we can make an assumption that in between of two agents with consecutive indexes i and $i+1$ there is a continuum of virtual agents having state varying from ρ_i to ρ_{i+1}. Denoting this continuously varying index by $M \in \mathbb{R}$ we can say that the state of the system ρ is a continuous and smooth function $\rho(t,M)$ with the property $\rho(t,i) = \rho_i(t)$. This definition of M coincides with the definition of Moskowitz function used to describe the number of vehicles passed through a fixed point in traffic modeling [32].

Once the index variable is continuous, we can think about it as a new space variable. Thus it is possible to use a continuation described in previous sections, where the distance between two consecutive agents is obviously $\Delta M = 1$. The derivatives of the state with respect to the index can be obtained by continuation, for example $\rho_{i+1} - \rho_i \to \partial\rho/\partial M$.

PDEs with index derivatives are very useful in multi-agent setups, when states of the agents are represented by their positions. Examples include traffic systems [19] with agents being cars, or systems of interacting particles and robot formations which will be discussed in the following sections.

D. Algorithm for general continuation procedure

The general continuation procedure for different kinds of systems can be summarized in the Algorithm 1. It checks for boundedness and space-dependency of the system and uses nonlinear continuation based on computational graphs. In case of multi-agent systems indices are treated as space coordinates. Linear systems are also covered by the algorithm since their computational graph is trivial.

V. Derivation of the Euler Equations

In the beginning of the XX century Hilbert posed his 6th problem, where he suggested to develop a rigorous way leading from the atomistic view to the laws of motion of continua. In particular, the problem can be formulated as a derivation of Euler equations for compressible fluids from the Newton's dynamics of individual particles.

For the most famous case of particles interacting through collision the Boltzmann equation was developed, describing evolution of the joint position-velocity probability distribution of particles. The method of how to transform individual's dynamics into Boltzmann equation is based on the Boltzmann-Grad limit [33], assuming velocities of colliding particles being independent. The following transformation from the Boltzmann equation to the Euler equations uses either Hilbert or Chapman-Erskog expansions with space contration limits [34], [35], Grad moments [36] or the method of invariant manifolds [37].

Another situation arises when the particles interact through long-range forces. In this case the Vlasov equation can be

Algorithm 1: General continuation procedure

```
Input: System of ODEs, d
    if system consists of moving agents then
        treat indices as coordinate space;   // Sec IV-C
    end
    if system has boundaries then            // Sec IV-B
        create ghost cells on boundaries such
            that inner nodes become homogeneous;
    end
    if system is space-dependent then        // Sec IV-A
        for each node do
            build computational graph;
        end
        find the most general structure
            of the computational graph;
        for each node do
            adjust computational graph such that it has
              the same structure with others;
            continuation();
        end
        approximate PDE coefficients
            by space-dependent functions;
    else
        build computational graph;
        continuation();                      // same for all
    end
Procedure continuation()
    Input: Computational graph of ODE, d
    for each node in graph, starting from leaves, do
        for each group of children with similar
          subgraphs do                       // Sec III
            compute PDE coefficients by (11) using d;
            replace group by PDE;
        end
    end
```

used instead of the Boltzmann equation to describe the joint position-velocity probability distribution. The derivation of the Euler equations from the Vlasov equation was performed in [38] using space-contracting limit. In particular it was shown that the resulting system has zero temperature, i.e. the velocities of individual particles coincide with the velocity field. However, due to the space contraction the particular form of the potential function was lost and the obtained pressure was just a square of the density.

Here we present a derivation of Euler equations directly from individual's dynamics using the continuation method described in previous sections. Contrary to other works, we do not use any kind of limits and we use only one assumption on the isotropy of the space. The assumption requires that for any particle its nearest neighbours are distributed around uniformly in every direction, which can be seen as a counterpart to the molecular chaos hypothesis for the standard derivation of the Boltzmann equation.

A. System of particles

It is assumed that the fluid consists of small particles interacting with each other, with every particle following simple Newton laws. We will study the system with n space dimensions, and the particles are assumed to have unit mass.

We further assume that there is an interaction between each pair of particles which is given by a force

$$F(x_i - x_j) = \frac{x_i - x_j}{\|x_i - x_j\|} f(\|x_i - x_j\|) = (x_i - x_j)\phi(\|x_i - x_j\|), \tag{37}$$

thus the force acts along the line connecting two particles with the smooth magnitude f depending only on the distance between particles. For simplicity we also define a function $\phi(s) = f(s)/s$ representing the scaled magnitude. We will consider an infinite number of particles and an infinitely large space, therefore we should assume that the cumulative force on any particle is finite. In particular for an equally distributed grid this implies that the magnitude of the force should satisfy

$$\int_{\varepsilon}^{+\infty} s^{n-1} f(s)\,ds < \infty \quad \forall \varepsilon > 0, \tag{38}$$

thus the interaction should be fast-decaying.

We then need to enumerate all particles. For this we will use multiindex $i \in \mathbb{Z}^n$. Now let us write the dynamics of a particle with multiindex i using the second Newton's Law:

$$\begin{cases} \dot{x}_i = v_i, \\ \dot{v}_i = \sum\limits_{q \neq 0} F(x_i - x_{i+q}), \end{cases} \tag{39}$$

where the summation is performed among all multiindices q in $\mathbb{Z}^n \setminus \{0\}$, since all the particles interact with each other. Both the position x_i and the velocity v_i are vectors in $\mathbb{R}^n$.

B. Derivation in the Euclidean space

Treating the coordinate x_i as a state and using the idea written in section IV-C we define a multiindex function $M(t,x)$ which is the inverse function of the coordinate: $M(t,x_i) := i$. Likewise, $x(t,i) = x_i(t)$ and thus $x(t,M(t,x)) \equiv x \ \forall x \in \mathbb{R}^n$.

Now let us write a property of inverse function of multiindex as $M(t,x(t,M)) \equiv M \ \forall M \in \mathbb{R}^n$, where the space for multiindices is continuous by the assumption in section IV-C. Taking the time and the index derivatives, we obtain the following very useful relations on Jacobians:

$$\frac{\partial M}{\partial t} + \frac{\partial M}{\partial x}\frac{\partial x}{\partial t} = 0, \tag{40}$$

$$\frac{\partial M}{\partial x}\frac{\partial x}{\partial M} = I. \tag{41}$$

Equation (40) can be seen as a PDE where the function M depends both on t and x. Recalling that the multiindex is assumed to be continuous, we can further utilize the first equation of (39) written in a form $\partial x(t,M)/\partial t = v(t,M)$, substitute it in (40) and obtain the following equation on the multiindex evolution:

$$\frac{\partial M}{\partial t} = -\frac{\partial M}{\partial x} v(t,M(t,x)) = -\frac{\partial M}{\partial x} u(t,x), \tag{42}$$

where the velocity function $u(t,x) = v(t,M(t,x))$ is defined as a velocity of a particle at some given point in space. Finally, taking the derivative with respect to space, we obtain

$$\frac{\partial}{\partial t}\left(\frac{\partial M}{\partial x}\right) = -\frac{\partial}{\partial x}\left(\frac{\partial M}{\partial x}u\right). \tag{43}$$

The Jacobian matrix $\frac{\partial M}{\partial x}(t,x)$ represents a *compression tensor*, which measures how close are neighbour particles with respect to different directions in the euclidean space. Evolution of this Jacobian in the euclidean space is described by the matrix PDE (43), which is essentially a transport equation with flow velocity given by $u(t,x)$.

Now we approach the second equation in (39). It would be desirable to transform it in such a way that we could obtain an evolution equation for the flow velocity $u(t,x)$. First of all, let us rewrite the second equation of (39) in a way more suitable for continuation, namely

$$\dot{v}_i = -\sum_{q>0}\left(F(x_{i+q}-x_i) - F(x_i - x_{i-q})\right), \tag{44}$$

where the summation is performed among all multiinidices which are greater than zero in lexicographical order, i.e. the first nonzero element of q should be positive.

Now we use the continuation of order 1 on a multidimensional system such that

$$x_{i+q} - x_i \;\to\; \frac{\partial x}{\partial M}\left(t, x_{i+q/2}\right)q, \quad x_i - x_{i-q} \;\to\; \frac{\partial x}{\partial M}\left(t, x_{i-q/2}\right)q,$$

which means that (44) becomes

$$\dot{v}_i = -\sum_{q>0}\left(F\left(\frac{\partial x}{\partial M}q\right)_{i+q/2} - F\left(\frac{\partial x}{\partial M}q\right)_{i-q/2}\right).$$

Applying the continuation further to the forces, we obtain

$$F_{i+q/2} - F_{i-q/2} \quad \to \quad \frac{\partial F}{\partial M}(t,x_i)q.$$

Thus (44) transforms into

$$\frac{\partial v}{\partial t} = -\sum_{q>0}\frac{\partial}{\partial M}\left(\left[\frac{\partial x}{\partial M}q\right]\phi\left(\left\|\frac{\partial x}{\partial M}q\right\|\right)\right)q, \tag{45}$$

where we used a definition of the force (37).

Now, we state the following result:

Proposition 1. *For any $q \in \mathbb{Z}^n$ and for any smooth scalar field ϕ the following identity holds:*

$$\begin{aligned}&\left[\frac{\partial}{\partial M}\left(\frac{\partial x}{\partial M}q\phi\right)q\right]^T = \\ &\nabla\cdot\left(\frac{\partial x}{\partial M}qq^T\frac{\partial x}{\partial M}^T\phi\right) - \left(\nabla\cdot\left(\frac{\partial x}{\partial M}\right)qq^T\frac{\partial x}{\partial M}^T\right)\phi,\end{aligned} \tag{46}$$

where ∇ denotes a row vector of derivatives with respect to x.

Proof. First, for convenience denote the left-hand side as a vector Q:

$$Q := \frac{\partial}{\partial M}\left(\frac{\partial x}{\partial M}q\phi\right)q = \frac{\partial}{\partial x}\left(\frac{\partial x}{\partial M}q\phi\right)\frac{\partial x}{\partial M}q. \tag{47}$$

Also define $h = (\partial x/\partial M)q$. Expanding $\partial(h\phi)/\partial x$, we get

$$Q = h\frac{\partial\phi}{\partial x}h + \frac{\partial h}{\partial x}h\phi = hh^T\frac{\partial\phi}{\partial x}^T + \frac{\partial h}{\partial x}h\phi. \tag{48}$$

Now, for any $h \in \mathbb{R}^n$

$$\nabla\cdot(hh^T) = \begin{pmatrix}\sum_i h_1\frac{\partial h_i}{\partial x_i} + \sum_i h_i\frac{\partial h_1}{\partial x_i} & \cdots & \sum_i h_n\frac{\partial h_i}{\partial x_i} + \sum_i h_i\frac{\partial h_n}{\partial x_i}\end{pmatrix},$$

which means that

$$\left(\nabla\cdot(hh^T)\right)^T = \frac{\partial h}{\partial x}h + (\nabla\cdot h)h. \tag{49}$$

Therefore the transpose of (48) is

$$Q^T = \frac{\partial\phi}{\partial x}hh^T + \nabla\cdot(hh^T)\phi - (\nabla\cdot h)h^T\phi. \tag{50}$$

Since for any matrix J and for any scalar field α

$$\nabla\cdot(\alpha J) = \frac{\partial\alpha}{\partial x}J + (\nabla\cdot J)\,\alpha, \tag{51}$$

we can simplify (50) as $Q^T = \nabla\cdot(hh^T\phi) - (\nabla\cdot h)h^T\phi$. The result of the proposition follows by substituting h and noticing that $\nabla\cdot((\partial x/\partial M)q) = (\nabla\cdot(\partial x/\partial M))q$. □

Proposition 1 allows us to rewrite (45) as being dependent only on the euclidean space divergences and the inverse of the compression tensor $\partial M/\partial x$. To finalize the derivation of a complete set of equations, recall the definition of the velocity field $u(t,x) = v(t,M(t,x))$. Taking the time derivative:

$$\frac{\partial u}{\partial t} = \frac{\partial v}{\partial t} + \frac{\partial v}{\partial M}\frac{\partial M}{\partial t},$$

which by (42) is

$$\frac{\partial u}{\partial t} = -\frac{\partial v}{\partial M}\frac{\partial M}{\partial x}u + \frac{\partial v}{\partial t}.$$

This equation can be simplified by $\partial u/\partial x = \partial v/\partial M\cdot\partial M/\partial x$. Finally, substituting (45) and (46) and combining the result with (43) we obtain a system

$$\begin{cases}\dfrac{\partial}{\partial t}\left(\dfrac{\partial M}{\partial x}\right) = -\dfrac{\partial}{\partial x}\left(\dfrac{\partial M}{\partial x}u\right), \\ \dfrac{\partial u}{\partial t} = -\dfrac{\partial u}{\partial x}u - \displaystyle\sum_{q>0}\left[\nabla\cdot\left(\frac{\partial x}{\partial M}qq^T\frac{\partial x}{\partial M}^T\phi\right)\right. \\ \qquad\left. -\left(\nabla\cdot\left(\dfrac{\partial x}{\partial M}\right)qq^T\frac{\partial x}{\partial M}^T\right)\phi\right]^T,\end{cases} \tag{52}$$

where $\phi = \phi(\|(\partial x/\partial M)q\|)$.

The system (52) has 12 states in 3-dimensional space, 9 for $\partial M/\partial x\,(t,x)$ and 3 for $u(t,x)$. It resembles the famous Grad 13-moment system [36], which extends the Euler equations by considering directional-dependent pressure tensor. The last state of the Grad 13-moment system is the inner energy, which does not appear in (52). The reason for this is that we derive a continuous interaction term explicitly from the interaction forces, which is possible only if the forces are defined by long-range potentials. As it was shown in [38], expressing a system with long-range potentials by the Euler equations leads to the solution with zero temperature, therefore the inner energy becomes functionally dependent on the velocity field and its evolution equation can be omitted.

C. Dimensionality reduction

It appears that in some special cases it is possible to reduce the system (52) by considering only one scalar characteristic of a compression in any space point instead of the whole compression tensor.

Indeed, we define a *density* as a determinant of the compression tensor, $\rho(t,x) := \det(\partial M/\partial x)(t,x)$. Not only the compression tensor itself, but also its determinant satisfies (43). This nontrivial fact holds because the compression tensor is the Jacobian, and the proof is given in Lemma 1 in the Appendix. Therefore from (43)

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho u). \tag{53}$$

This equation is the first of the complete set of Euler equations.

Unfortunately, the second equation of (52) depends on the whole compression tensor and thus it cannot be described only by the means of density. This is reasonable since in general the system can have different pressures in different directions in response to different compressions. Therefore in order to simplify the system we need to assume that the compression can be represented by a single number, i.e. that it is compressed equally in all directions.

Assumption 1 (*Isotropy*)**.** Compression tensor $\partial M/\partial x(t,x)$ is isotropic (equal in all directions), thus it can be represented as a rotation matrix multiplied by a scalar.

This assumption looks restricting at first glance, but for the infinitely large system with infinitely many particles the system indeed "looks the same" in all directions at every point, thus we can say it is isotropic.

Assumption 1 has long-lasting implications. Define $l(t,x) := \lambda(\partial x/\partial M(t,x))$, since all the eigenvalues are equal. This variable, called *specific distance*, represents an average distance between two neighbouring particles at point x. By definition of the density $\rho = l^{-n}$. Further, $\left\| \frac{\partial x}{\partial M} q \right\| = l\|q\|$. Breaking the summation in (52) in a sum of all possible lengths r of multiindex vectors, we can rewrite the summation term as

$$\sum_{r^2\in\mathbb{N}} \left[\nabla \cdot \left(\phi(rl) \frac{\partial x}{\partial M} \sum_{\substack{q>0 \\ \|q\|=r}} \left(qq^T\right) \frac{\partial x}{\partial M}^T \right) - \right. \\ \left. - \phi(rl) \left(\nabla \cdot \left(\frac{\partial x}{\partial M} \right) \sum_{\substack{q>0 \\ \|q\|=r}} \left(qq^T\right) \frac{\partial x}{\partial M}^T \right) \right]^T. \tag{54}$$

Proposition 2. *Given r such that $r^2 \in \mathbb{N}$, the summation over all outer products of multiindices of a length r is proportional to the identity matrix, i.e. there exists $\beta(r)$ such that*

$$\sum_{\substack{q>0 \\ \|q\|=r}} qq^T = \beta(r) I. \tag{55}$$

Proof. First of all, we will show that all nondiagonal elements in (55) are zero. Indeed, for any positive q its contribution to kj-th element of matrix (55) is given by $q_k q_j$. But for any $k \neq j$ we can pick $\bar{q}$ such that it equals q except $\bar{q}_{\max(k,j)} = -q_{\max(k,j)}$. In this case $\bar{q}$ is also positive and thus is included into the summation, while the contribution to kj-th element of (55) has opposite sign. Therefore all nondiagonal elements of (55) are zero.

Further, all diagonal elements of (55) are equal. This can be proven by analogous argument. Indeed, we can take a positive q and look at the elements q_k^2 and q_j^2. Then $\bar{q}$ which is equal to q except for $\bar{q}_k = \operatorname{sgn}(q_k)|q_j|$ and $\bar{q}_j = \operatorname{sgn}(q_j)|q_k|$ is also positive, but swaps the contributions between k-th and j-th diagonal elements. Thus all the contributions to the diagonal elements are equal. Finally,

$$\operatorname{Tr} \sum_{\substack{q>0 \\ \|q\|=r}} qq^T = \sum_{\substack{q>0 \\ \|q\|=r}} q^T q = r^2 \cdot \#_r q = n\beta(r), \tag{56}$$

where $\#_r q$ denotes the number of positive multiindices q with length r and we define $\beta(r) = r^2/n \cdot \#_r q$. It is worth noticing that by [39] the average approximate behaviour of the number of positive multiindices q with length r is $\#_r q \propto r^{n-1}$ as $r \to +\infty$, thus $\beta(r) \propto r^n$. □

By Assumption 1

$$\frac{\partial x}{\partial M} \frac{\partial x}{\partial M}^T = l^2 I. \tag{57}$$

Using Proposition 2 and (57), (54) becomes

$$\sum_{r^2\in\mathbb{N}} \beta(r) \left[\nabla \cdot \left(\phi(rl) l^2 I \right) - \phi(rl) \left(\nabla \cdot \left(\frac{\partial x}{\partial M} \right) \frac{\partial x}{\partial M}^T \right) \right]^T.$$

The value inside of the square brackets can be simplified further. Indeed, by (51) it is possible to inject density inside, which gives

$$\frac{1}{\rho} \left[\nabla \cdot (\rho \phi(rl) l^2 I) - \frac{\partial \rho}{\partial x} \phi(rl) l^2 I - \phi(rl) \rho \left(\nabla \cdot \left(\frac{\partial x}{\partial M} \right) \frac{\partial x}{\partial M}^T \right) \right]^T \\ = \frac{1}{\rho} \left[\nabla \cdot (\rho \phi(rl) l^2 I) - \phi(rl) \left(\nabla \cdot \left(\rho \frac{\partial x}{\partial M} \right) \frac{\partial x}{\partial M}^T \right) \right]^T.$$

Finally, the second term in the square brackets appears to be zero, since Lemma 2 in the Appendix proves that $\nabla \cdot \left(\rho \frac{\partial x}{\partial M} \right) \frac{\partial x}{\partial M}^T = 0$. Using this Lemma and the fact that $\nabla \cdot (\rho \phi(rl) l^2 I) = \nabla(\rho \phi(rl) l^2)$, we can define the *pressure*:

$$P = \sum_{r^2\in\mathbb{N}} \beta(r) \rho \phi(lr) l^2 = \sum_{r^2\in\mathbb{N}} \frac{\beta(r)}{r} l^{1-n} f(lr). \tag{58}$$

Note that the pressure is well-defined since the sum is convergent by the property (38). With this definition, the system (52) together with (53) turns into the famous *Euler equations*:

$$\begin{cases} \dfrac{\partial \rho}{\partial t} = -\nabla \cdot (\rho u), \\ \dfrac{\partial u}{\partial t} = -\dfrac{\partial u}{\partial x} u - \dfrac{\nabla P}{\rho}^T. \end{cases} \tag{59}$$

Therefore the following theorem was proven:

Theorem 3. *There exists a valid continuation process which leads from the Newtonian system* (39) *to the Euler equations* (59) *under the assumption that the system is locally isotropic in every point in space.*

Remark 3 (Non-complete interaction topologies). In the original ODE system (39) we assumed that an interaction exists between every pair of particles, i.e. that the topology of interactions is all-to-all. In general in order to obtain (39) it would be sufficient to use any topology for which the isotropy required in Assumption 1 is possible. The difference in topologies would modify the definitions of density $P(t,x)$ in (58).

For example, for the grid topology with equations given by

$$\begin{cases} \dot{x}_i = v_i, \\ \dot{v}_i = \sum_{k=1}^{n} \left(F(x_i - x_{i-e_k}) - F(x_i - x_{i+e_k}) \right), \end{cases} \tag{60}$$

where e_k denotes the k-th basis vector of $\mathbb{R}^n$, the continuation renders the same Euler equations (59) with the pressure given by $P = f(l)/l^{n-1}$.

VI. Control of Robotic Swarm

In this section we will demonstrate how the continuation method described above can help in the analysis and design of control laws for large-scale systems. We will do it by using an example of a robotic swarm, i.e. a formation of robots whose goal is to follow some desired trajectory while passing through obstacles and preserving relative agents' positions.

Control of robotic formations is an extensively studied topic, see recent reviews [40], [41]. However most of the methods rely on the graph-theoretic properties of interaction topology and on simple linear controllers to provide stability. A PDE approach was taken in [42] where the Euler PDE with diffusion terms was used to model the flocks of birds. The authors proposed a PDE to describe the behaviour of agents and analyzed it to study a symmetry breaking which leads to a coherent movement of birds. Similar PDE model was used to control 3D agent formation with 2D disc communication topology via backstepping in [43]. Lattice-based spatially-invariant models for platooning were considered in [44], [45], where stability properties of infinite systems were studied in various space dimensions.

Works mentioned above which use PDE representations of multi-agent systems just assume a PDE model, which can be justified by a limiting case of the infinite number of agents. Contrary, we will base our analysis on the continuation procedure, rigorously introducing a PDE to describe a finite formation of drones. We will study this PDE and recover a nonlinear local control law which, being applied to the agents, forces the whole formation to follow the desired density profile.

A. Continuation and PDE Control

Let us start from a system of drones having double integrator dynamics:

$$\ddot{x}_i = \tau_i. \tag{61}$$

Here $x_i \in \mathbb{R}^n$ is a position of the i-th drone in n-dimensional space and $\tau_i \in \mathbb{R}^n$ is a control we want to design. The drones are enumerated with multiindices $i \in \mathbb{Z}^n$. Define $v_i = \dot{x}_i$. Similarly to the previous section we introduce multiindex function $M(t,x)$ such that $M(t,x_i) \equiv i$ and then perform a continuation. The resulting system is

$$\begin{cases} \dfrac{\partial \rho}{\partial t} = -\nabla \cdot (\rho u), \\ \dfrac{\partial u}{\partial t} = -\dfrac{\partial u}{\partial x} u + \tau(x,t), \end{cases} \tag{62}$$

where $\tau(t,x) = \tau(t, M(t,x))$ is a continuation of the control τ_i.

Now let us formulate a desired system which will be used as a reference which the real formation should converge to. Given a velocity profile $u_d(x)$, we define the desired density $\rho_d(t,x)$ to follow this velocity profile. Essentially this means "desired agents" have single-integrator dynamics. Note that in general u_d can be dependent on time but we don't consider it for simplicity of writing.

Thus we assume the desired system is governed by

$$\frac{\partial \rho_d}{\partial t} = -\nabla \cdot (\rho_d u_d). \tag{63}$$

Our goal is to derive $\tau(t,x)$ such that $\rho \to \rho_d$. First, direct calculations from (62) and (63) lead to the following systems in terms of flows (ρu) and $(\rho_d u_d)$:

$$\begin{aligned} \frac{\partial (\rho u)}{\partial t} &= -\nabla \cdot (\rho u) u - \rho \frac{\partial u}{\partial x} u + \rho \tau(x,t), \\ \frac{\partial (\rho_d u_d)}{\partial t} &= -\nabla \cdot (\rho_d u_d) u_d. \end{aligned} \tag{64}$$

Define the deviation from the desired density $\tilde{\rho} = \rho - \rho_d$. Then the second-order equation for the deviation is

$$\frac{\partial^2 \tilde{\rho}}{\partial t^2} = \nabla \cdot \left[\nabla \cdot (\rho u) u - \nabla \cdot (\rho_d u_d) u_d + \rho \frac{\partial u}{\partial x} u - \rho \tau(x,t) \right].$$

In order to cancel the nonlinear terms, define now the control τ as

$$\begin{aligned} \tau = \frac{\partial u}{\partial x} u + \frac{1}{\rho} \Big[& \nabla \cdot (\rho u) u - \nabla \cdot (\rho_d u_d) u_d + \\ & + \alpha (\rho_d u_d - \rho u) + \beta \nabla (\rho_d - \rho)^T \Big], \end{aligned} \tag{65}$$

where α and β are some positive gains. Then the equation for the density deviation transforms into

$$\frac{\partial^2 \tilde{\rho}}{\partial t^2} = -\alpha \frac{\partial \tilde{\rho}}{\partial t} + \beta \nabla^2 \tilde{\rho}. \tag{66}$$

This equation is a wave equation with damping and thus it is asymptotically stable if $\tilde{\rho} = 0$ on the boundary of the domain [46]. Choosing a desired system such that $\rho_d = 0$ on the boundary and using a continuation of ρ such that $\rho = 0$ on the boundary ensures satisfaction of the boundary condition.

B. Discretization of the control

Formula (65) for PDE (62) is local by its nature, but it should be discretized to be implemented on every agent of the original ODE (61). One particular discretization is described next.

First of all, for the agent i define a matrix G_i as a discretization of the compression tensor:

$$[G_i]_j = (x_{i+e_j} - x_{i-e_j})/2 \approx \frac{\partial x}{\partial M_j}(t, x_i), \tag{67}$$

where e_j is the j-th unit basis vector and $[G_i]_j$ represent the j-th column of G_i. The matrix G_i depends on the positions of $2n$ neighbouring agents of the i-th agent, thus the interaction topology is a lattice. In the same way as G_i we define a matrix W_i representing a velocity Jacobian:

$$[W_i]_j = (v_{i+e_j} - v_{i-e_j})/2 \approx \frac{\partial u}{\partial M_j}(t, x_i). \tag{68}$$

Now we can write formulas for all terms inside of (65) depending on the real system:

$$\begin{aligned}
&1). \quad \frac{\partial u}{\partial x}u = \frac{\partial u}{\partial M}\frac{\partial M}{\partial x}u \approx W_i G_i^{-1} v_i,\\
&2). \quad \nabla \cdot u = \sum_{j=1}^{n} \frac{\partial u_j}{\partial M}\frac{\partial M}{\partial x_j} \approx \sum_{j=1}^{n} [W_i^T]_j \cdot [G_i^{-1}]_j,\\
&3). \quad \rho \approx 1/\det G_i,\\
&4). \quad \nabla\rho \approx -\rho^2 \nabla(\det G_i) \approx -\rho^2 \frac{\partial(\det G_i)}{\partial M} G_i^{-1},
\end{aligned} \tag{69}$$

where the gradient of the determinant $\det G_i$ should be computed according to the determinant formula, using second derivatives of the positions discretized similarly to (67):

$$\begin{aligned}
\frac{\partial^2 x}{\partial M_j \partial M_k}(x_i, t) &\approx (x_{i+e_j+e_k} + x_{i-e_j-e_k} - x_{i+e_j-e_k} - x_{i-e_j+e_k})/4,\\
\frac{\partial^2 x}{\partial M_j^2}(x_i, t) &\approx x_{i+e_j} - 2x_i + x_{i-e_j}.
\end{aligned}$$

Since the gradient of the determinant depends on the second derivatives, in total each agent requires information about the velocities of its $2n$ neighbouring agents and the positions of its $2n^2$ neighbouring agents, including diagonal ones.

Finally, substituting (69) into (65), the formula for the control action τ_i appears as

$$\begin{aligned}
\tau_i = &\left[W_i G_i^{-1} + \sum_{j=1}^{n} [W_i^T]_j \cdot [G_i^{-1}]_j - \alpha \right] v_i +\\
&+ \left[\beta I - v_i v_i^T\right] \frac{1}{\det G_i} G_i^{-T} \frac{\partial(\det G_i)}{\partial M}^T +\\
&+ \det G_i \Big[\alpha \rho_d u_d + (\beta I - u_d u_d^T)\nabla\rho_d^T - \rho_d (\nabla \cdot u_d) u_d \Big].
\end{aligned} \tag{70}$$

C. Boundary conditions

For the system (66) to converge to zero proper boundary conditions should be used. Namely, the continuation should be chosen such that $\rho = 0$ outside of the formation. As it was shown in IV-B, boundary conditions for PDE correspond to "ghost agents" in the ODE case. In particular, information about neighbour agents is used in (67) and (68). Therefore specifying boundary conditions means specifying positions $x_{i\pm e_j}$ and velocities $v_{i\pm e_j}$ for the nonexisting agents.

Proposition 3. *Assume agent $i - e_j$ is a ghost agent. Then*

$$x_{i-e_j} = 3x_i - 2x_{i+e_j}, \quad v_{i-e_j} = 2v_i - v_{i+e_j} \tag{71}$$

ensures $\rho_{i-e_j} = 0$ and a correct computation of $[W_i]_j$ by (68).

Proof. Choice (71) for velocities is natural, since being substituted in (68) this leads to an approximation of the velocity gradient based solely on the i and $i+e_j$ agents.

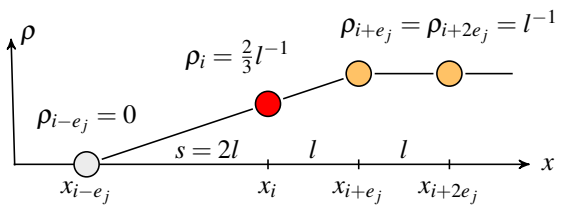

Figure 6. Left boundary of the system (61) with control (70). Agent i is on the boundary, the position of the "ghost agent" $i - e_j$ is chosen such that ρ linearly goes to zero at x_{i-e_j}.

For the position we want that the compression tensor (67) "feels" that the drone i is on the border. For this we can use such an approximation that the density near the border will linearly diminish to zero, see Fig. 6. Namely, let us look at 1D case and fix i-th agent to be on the left border, with $\rho_{i-e_j} = 0$ for the ghost agent. Assume further that the distance between each pair of existing agents is constant and equal to l. Then $\rho_{i+e_j} = l^{-1}$. Define an unknown distance $s := x_i - x_{i-e_j}$. Then asking for a linear dependency of a density on position, we have necessarily

$$\rho_i = \frac{l\rho_{i-e_j} + s\rho_{i+e_j}}{l+s} = \frac{s}{l(l+s)}.$$

But by (67) $\rho_i = 2/(l+s)$, which immediately gives the answer $s = 2l$, or $x_{i-e_j} = x_i + 2(x_i - x_{i+e_j})$, which is (71). □

Proposition 3 finalizes the formulation of the boundary conditions and thus the correct implementation of (70).

D. Numerical Simulation

To demonstrate the control policy (70) we performed a numerical simulation of a cubic formation of 512 drones in 3D space. The goal was to reach a cubic formation, fly through a window and restore the cubic formation after the maneuver.

Assume the center of the window is placed at the point $(x_0, 0, 0)$, and the formation should fly through it starting from the origin. The desired velocity field $u_d(x, y, z)$ able to fulfill the task was constructed as

$$u_{d_x} = 1, \quad u_{d_{y|z}} = 0.05 \operatorname{atan}(x - x_0) e^{-\frac{(x-x_0)^2}{100}} y|z,$$

where $y|z$ denotes y or z, see the left panel of Fig. 7 for the streamlines projected on the x-y plane. For simplicity the desired system (63) was simulated by first-order integrators following the desired velocity profile, and the density $\rho_d(x, t)$ was interpolated between agents.

Both the desired system (63) and the real system (61) were simulated for the cubic formation of $8 \times 8 \times 8$ drones using Euler method. The initial positions for the real system were multiplied by 2 in comparison to the desired system and a uniform noise $U(-2, 2)$ was added. The control gains were chosen as $\alpha = 3$ and $\beta = 100$. The convergence of the real density to the desired one is shown on the right panel of Fig. 7 and snapshots of the simulation are presented in Fig. 8. It is clear that the real formation, being heavily disturbed in the beginning, converges to the desired shape in less than 5 seconds and then follows the desired pattern, successfully passing through the window.

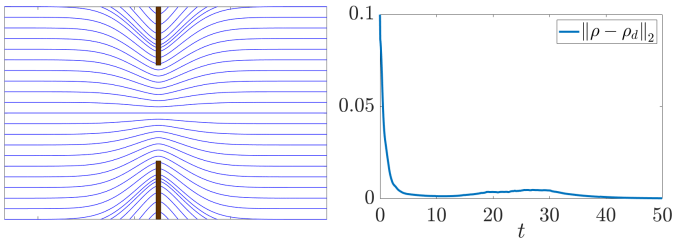

Figure 7. Left: streamlines of the desired velocity field $u_d(x,y)$. Right: convergence of the L_2 norm of the density deviation.

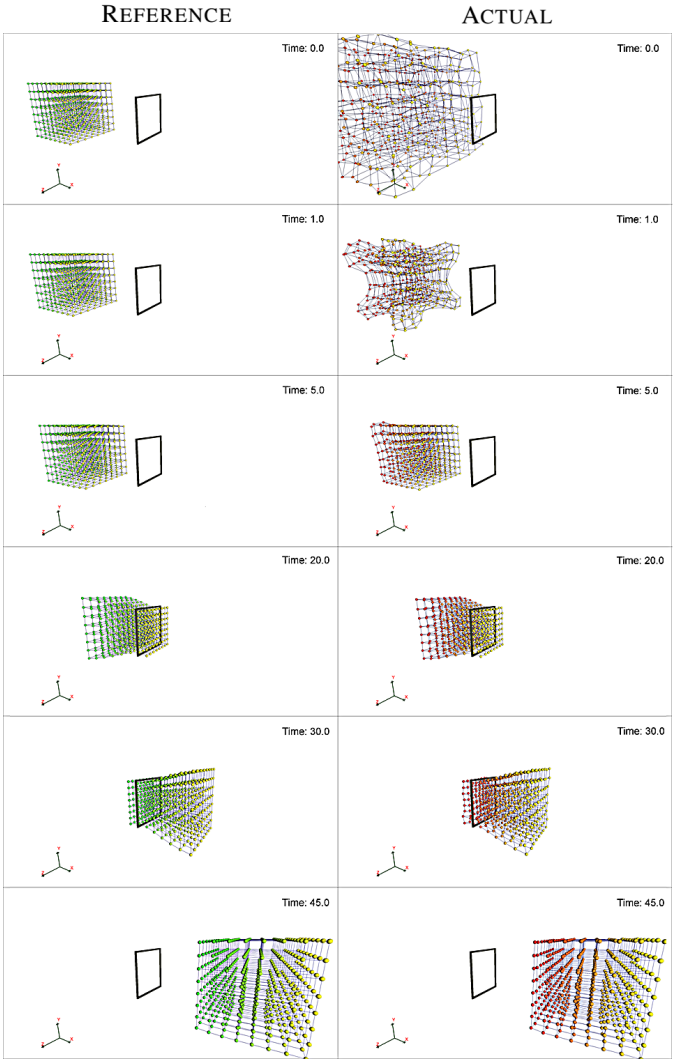

Figure 8. Simulation of drones flying through window. Rows correspond to times $t = \{0s, 1s, 5s, 20s, 30s, 45s\}$. Left column, reference: desired system (63), governed by single integrators. Right column, actual: heavility perturbed real system (61) with control (70) which converges to the desired one.

VII. Conclusion

We presented a general process of transformation of ODE systems into their PDE counterparts, defining the continuation to be valid if the original ODE system could be obtained from the PDE version by a correct discretization. We further showed that the spectrum of PDE converges to the spectrum of ODE. The continuation method was then elaborated for many classes of systems including nonlinear, multidimensional and space- and time-varying. Based on this method, new continuous models can be derived and further utilized for analysis and control purposes.

As an example we used the continuation to show how the Euler equations for compressible fluid can be derived from the newtonian particle interactions, providing more intuition into Hilbert's 6th problem. The same continuation was then used to describe a robot formation flying through window. We developed a control algorithm to stabilize a desired trajectory based on a continuous representation of the formation. This algorithm is distributed as every robot requires information only about neighbouring robots.

Appendix

Lemma 1. *Let $J(t,x) \in \mathbb{R}^{n\times n}$ be the Jacobian matrix of function $M(t,x)$. Let $J(t,x)$ satisfies the dynamic equation*

$$\frac{\partial J}{\partial t} = -\frac{\partial (Ju)}{\partial x}, \tag{72}$$

where $u = u(t,x)$ is some vector field. Then the determinant $\det J$ satisfies the same equation:

$$\frac{\partial \det J}{\partial t} = -\frac{\partial}{\partial x}\cdot(\det J \cdot u). \tag{73}$$

Proof. First of all let us rewrite (72) for one element J_{ik} of the matrix J:

$$\begin{aligned}\frac{\partial J_{ik}}{\partial t} = -\frac{\partial (J_i u)}{\partial x_k} = -\sum_{j=1}^{n}\frac{\partial^2 M_i}{\partial x_k \partial x_j}u_j - \sum_{j=1}^{n} J_{ij}\frac{\partial u_j}{\partial x_k} = \\ = -\sum_{j=1}^{n}\frac{\partial J_{ik}}{\partial x_j}u_j - \sum_{j=1}^{n} J_{ij}\frac{\partial u_j}{\partial x_k},\end{aligned} \tag{74}$$

where we used the fact that $J = \partial M/\partial x$. Now let us recall the definition of the determinant: $\det J = \sum_\sigma \operatorname{sgn}(\sigma)\prod_{i=1}^n J_{\sigma_i,i}$, where σ is a permutation of the set $\{1,2,...,n\}$ and $\sum_\sigma$ is taken over all possible permutations, with $\operatorname{sgn}(\sigma)$ being the sign of the permutation. Take the time derivative and substitute (74):

$$\begin{aligned}\frac{\partial \det J}{\partial t} = \sum_\sigma \operatorname{sgn}(\sigma)\sum_{k=1}^{n}\frac{\partial J_{\sigma_k,k}}{\partial t}\prod_{i=1,i\neq k}^{n} J_{\sigma_i,i} = \\ = -\sum_{j=1}^{n}\sum_\sigma \operatorname{sgn}(\sigma)\sum_{k=1}^{n}\left[\frac{\partial J_{\sigma_k,k}}{\partial x_j}u_j + J_{\sigma_k,j}\frac{\partial u_j}{\partial x_k}\right]\prod_{i=1,i\neq k}^{n} J_{\sigma_i,i}\end{aligned} \tag{75}$$

We will investigate two parts of (75), corresponding to the two terms inside the square brackets. For the first term we have

$$-\sum_{j=1}^{n}\sum_\sigma \operatorname{sgn}(\sigma)\sum_{k=1}^{n}\frac{\partial J_{\sigma_k,k}}{\partial x_j}u_j\prod_{i=1,i\neq k}^{n} J_{\sigma_i,i} = -\sum_{j=1}^{n}\frac{\partial \det J}{\partial x_j}u_j = -\frac{\partial \det J}{\partial x}u.$$

The second term is a little more tricky:

$$-\sum_{j=1}^{n}\sum_{\sigma}\operatorname{sgn}(\sigma)\sum_{k=1}^{n}J_{\sigma_k,j}\frac{\partial u_j}{\partial x_k}\prod_{i=1,i\neq k}^{n}J_{\sigma_i,i}=-\det J\sum_{j=1}^{n}\frac{\partial u_j}{\partial x_j}-$$
$$-\sum_{j=1}^{n}\sum_{\sigma}\operatorname{sgn}(\sigma)\sum_{k=1,k\neq j}^{n}J_{\sigma_k,j}\frac{\partial u_j}{\partial x_k}\prod_{i=1,i\neq k}^{n}J_{\sigma_i,i}.$$

Here we split the summation over k into the term with $k=j$ and all other terms. The former immediately gives the determinant multiplied by the divergence of the vector field, where the latter sum over all other terms is zero. Indeed, imagine a permutation $\bar{\sigma}$ such that it is equal to σ except σ_j and σ_k are swapped. Then the sign of $\bar{\sigma}$ is opposite to the sign of σ. Further, since the product $J_{\sigma_k,j}J_{\sigma_j,j}$ is the only way in which σ_k and σ_j enter the formula, the absolute value does not change with the change of permutation. Therefore for each j,k and for each permutation there exists a permutation which cancels them out. Finally, substitution of the nonzero terms of the last equations into (75) leads to (73). □

Lemma 2. *Let $\partial x/\partial M$ be isotropic, i.e. represented by a scalar multiplied by a rotation matrix, and let $\rho = \det(\partial M/\partial x)$. Then*

$$\nabla\cdot\left(\rho\frac{\partial x}{\partial M}\right)\frac{\partial x}{\partial M}^{T}=0. \tag{76}$$

Proof. Define $\lambda=\lambda(\partial M/\partial x)$, thus $\rho=\lambda^n$. By isotropy,

$$\frac{\partial x}{\partial M}=\lambda^{-2}\frac{\partial M}{\partial x}^{T}$$

and therefore, by using (51), the left-hand side of (76) is

$$\begin{aligned}&\nabla\cdot\left(\lambda^{n-2}\frac{\partial M}{\partial x}^{T}\right)\frac{\partial M}{\partial x}\lambda^{-2}=\\&=\lambda^{n-4}\nabla\cdot\left(\frac{\partial M}{\partial x}^{T}\right)\frac{\partial M}{\partial x}+(n-2)\lambda^{n-3}\frac{\partial\lambda}{\partial x}.\end{aligned} \tag{77}$$

Now let us investigate the first term more closely. Taking the divergence and looking at j-th element, we see that

$$\left[\nabla\cdot\left(\frac{\partial M}{\partial x}^{T}\right)\frac{\partial M}{\partial x}\right]_j=\sum_{k=1}^{n}\frac{\partial^2 M}{\partial x_k^2}^{T}\frac{\partial M}{\partial x_j}. \tag{78}$$

Now, by isotropy

$$\frac{\partial M}{\partial x_j}^{T}\frac{\partial M}{\partial x_k}=0\ \ \forall j\neq k,\quad \frac{\partial M}{\partial x_k}^{T}\frac{\partial M}{\partial x_k}=\lambda^2. \tag{79}$$

Taking the derivative of the multiplication of basis vectors:

$$\frac{\partial}{\partial x_j}\left(\frac{\partial M}{\partial x_k}^{T}\frac{\partial M}{\partial x_k}\right)=2\frac{\partial^2 M}{\partial x_j\partial x_k}^{T}\frac{\partial M}{\partial x_k}, \tag{80}$$

but at the same time the value under the derivative is λ^2 by (79), therefore

$$\frac{\partial}{\partial x_j}\left(\frac{\partial M}{\partial x_k}^{T}\frac{\partial M}{\partial x_k}\right)=\frac{\partial\lambda^2}{\partial x_j}=2\lambda\frac{\partial\lambda}{\partial x_j}. \tag{81}$$

Then, taking the derivative of multiplication of different basis vectors with $j\neq k$, by (79) we obtain zero:

$$\frac{\partial}{\partial x_k}\left(\frac{\partial M}{\partial x_j}^{T}\frac{\partial M}{\partial x_k}\right)=\frac{\partial^2 M}{\partial x_j\partial x_k}^{T}\frac{\partial M}{\partial x_k}+\frac{\partial M}{\partial x_j}^{T}\frac{\partial^2 M}{\partial x_k^2}=0,$$

which by equality of (80) and (81) means that for $j\neq k$

$$\frac{\partial M}{\partial x_j}^{T}\frac{\partial^2 M}{\partial x_k^2}=-\frac{\partial^2 M}{\partial x_j\partial x_k}^{T}\frac{\partial M}{\partial x_k}=-\lambda\frac{\partial\lambda}{\partial x_j}. \tag{82}$$

In the case of $j=k$ by equality of (80) and (81) we have

$$\frac{\partial^2 M}{\partial x_j^2}^{T}\frac{\partial M}{\partial x_j}=\lambda\frac{\partial\lambda}{\partial x_j}. \tag{83}$$

Combination of (82) and (83) means that (78) is

$$\left[\nabla\cdot\left(\frac{\partial M}{\partial x}^{T}\right)\frac{\partial M}{\partial x}\right]_j=(2-n)\lambda\frac{\partial\lambda}{\partial x_j}. \tag{84}$$

Finally, substituting (84) in (77) gives zero. □

ACKNOWLEDGMENT

The Scale-FreeBack project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement N 694209).

REFERENCES

[1] F. V. Atkinson, *Discrete and continuous boundary problems*. Academic Press, 1964.

[2] F. R. Gantmakher and M. G. Krejn, *Oscillation matrices and kernels and small vibrations of mechanical systems*. American Mathematical Soc., 2002, no. 345.

[3] M. Aoki, "Control of large-scale dynamic systems by aggregation," *IEEE Transactions on Automatic Control*, vol. 13, no. 3, pp. 246–253, 1968.

[4] M. U. B. Niazi, X. Cheng, C. Canudas-de Wit, and J. M. Scherpen, "Structure-based clustering algorithm for model reduction of large-scale network systems," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 5038–5043.

[5] M. U. B. Niazi, D. Deplano, C. Canudas-de Wit, and A. Y. Kibangou, "Scale-free estimation of the average state in large-scale systems," *IEEE Control Systems Letters*, vol. 4, no. 1, pp. 211–216, 2019.

[6] D. Nikitin, C. C. de Wit, and P. Frasca, "Control of average and deviation in large-scale linear networks," *IEEE Transactions on Automatic Control*, 2022.

[7] J. A. Acebrón, L. L. Bonilla, C. J. P. Vicente, F. Ritort, and R. Spigler, "The Kuramoto model: A simple paradigm for synchronization phenomena," *Reviews of modern physics*, vol. 77, no. 1, p. 137, 2005.

[8] H. Grabert, *Projection operator techniques in nonequilibrium statistical mechanics*. Springer, 2006, vol. 95.

[9] D. Q. Nykamp and D. Tranchina, "A population density approach that facilitates large-scale modeling of neural networks: Analysis and an application to orientation tuning," *Journal of computational neuroscience*, vol. 8, no. 1, pp. 19–50, 2000.

[10] Y. Yang, D. V. Dimarogonas, and X. Hu, "Shaping up crowd of agents through controlling their statistical moments," in *2015 European Control Conference (ECC)*. IEEE, 2015, pp. 1017–1022.

[11] S. Zhang, A. Ringh, X. Hu, and J. Karlsson, "Modeling collective behaviors: A moment-based approach," *IEEE Transactions on Automatic Control*, vol. 66, no. 1, pp. 33–48, 2021.

[12] C. Kuehn, "Moment closure—a brief review," *Control of self-organizing nonlinear systems*, pp. 253–271, 2016.

[13] D. Nikitin, C. Canudas-De-Wit, and P. Frasca, "Shape-based nonlinear model reduction for 1D conservation laws," in *IFAC World Congress*, 2020.

[14] L. Lovász, *Large networks and graph limits*. American Mathematical Soc., 2012, vol. 60.

[15] S. Gao and P. E. Caines, "Graphon control of large-scale networks of linear systems," *IEEE Transactions on Automatic Control*, vol. 65, no. 10, pp. 4090–4105, 2019.

[16] R. Vizuete, P. Frasca, and F. Garin, "Graphon-based sensitivity analysis of SIS epidemics," *IEEE Control Systems Letters*, vol. 4, no. 3, pp. 542–547, 2020.

[17] G. S. Medvedev, "The nonlinear heat equation on dense graphs and graph limits," *SIAM Journal on Mathematical Analysis*, vol. 46, no. 4, pp. 2743–2766, 2014.

[18] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind, "Automatic differentiation in machine learning: a survey," *Journal of machine learning research*, vol. 18, 2018.

[19] T. G. Molnár, D. Upadhyay, M. Hopka, M. Van Nieuwstadt, and G. Orosz, "Lagrangian models for controlling large-scale heterogeneous traffic," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 3152–3157.

[20] F. van Wageningen-Kessels, H. Van Lint, K. Vuik, and S. Hoogendoorn, "Genealogy of traffic flow models," *EURO Journal on Transportation and Logistics*, vol. 4, no. 4, pp. 445–473, 2015.

[21] P. Barooah, P. G. Mehta, and J. P. Hespanha, "Mistuning-based control design to improve closed-loop stability margin of vehicular platoons," *IEEE Transactions on Automatic Control*, vol. 54, no. 9, pp. 2100–2113, 2009.

[22] U. Biccari, D. Ko, and E. Zuazua, "Dynamics and control for multi-agent networked systems: A finite-difference approach," *Mathematical Models and Methods in Applied Sciences*, vol. 29, no. 04, pp. 755–790, 2019.

[23] D. Nikitin, C. Canudas-De-Wit, and P. Frasca, "Boundary control for stabilization of large-scale networks through the continuation method," in *CDC 2021-60th IEEE Conference on Decision and Control*, 2021.

[24] L. Tumash, C. Canudas-De-Wit, and M. L. Delle Monache, "Multi-directional continuous traffic model for large-scale urban networks," *Submitted to Transportation Research: Part B*, 2021.

[25] ——, "Boundary control for multi-directional traffic on urban networks," in *CDC 2021-60th IEEE Conference on Decision and Control*, 2021.

[26] D. Nikitin, C. Canudas-De-Wit, P. Frasca, and U. Ebels, "Synchronization of spin-torque oscillators via continuation method," *Submitted to IEEE Transactions on automatic control*, 2021.

[27] B. Bamieh, F. Paganini, and M. A. Dahleh, "Distributed control of spatially invariant systems," *IEEE Transactions on automatic control*, vol. 47, no. 7, pp. 1091–1107, 2002.

[28] A. E. Frazho and W. Bhosri, "Toeplitz and Laurent operators," in *An Operator Perspective on Signals and Systems*. Springer, 2010.

[29] J. V. Fischer, "Four particular cases of the Fourier transform," *Mathematics*, vol. 6, no. 12, p. 335, 2018.

[30] D. S. Mitrinovic and P. M. Vasic, *Analytic inequalities*. Springer, 1970.

[31] A. N. Kolmogorov, "On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition," in *Doklady Akademii Nauk*, vol. 114, no. 5. Russian Academy of Sciences, 1957, pp. 953–956.

[32] G. F. Newell, "A simplified theory of kinematic waves in highway traffic, part I: General theory," *Transportation Research Part B: Methodological*, vol. 27, no. 4, pp. 281–287, 1993.

[33] I. Gallagher, L. Saint-Raymond, and B. Texier, *From Newton to Boltzmann: hard spheres and short-range potentials*. European Mathematical Society, 2013.

[34] L. Saint-Raymond, *Hydrodynamic limits of the Boltzmann equation*. Springer Science & Business Media, 2009, no. 1971.

[35] S. Chapman and T. G. Cowling, *The mathematical theory of non-uniform gases: an account of the kinetic theory of viscosity, thermal conduction and diffusion in gases*. Cambridge university press, 1990.

[36] H. Grad, "On the kinetic theory of rarefied gases," *Communications on pure and applied mathematics*, vol. 2, no. 4, pp. 331–407, 1949.

[37] A. Gorban and I. Karlin, "Hilbert's 6th problem: exact and approximate hydrodynamic manifolds for kinetic equations," *Bulletin of the American Mathematical Society*, vol. 51, no. 2, pp. 187–246, 2014.

[38] S. Caprino, R. Esposito, R. Marra, and M. Pulvirenti, "Hydrodynamic limits of the Vlasov equation," *Communications in partial differential equations*, vol. 18, no. 5-6, pp. 805–820, 1993.

[39] R. Takloo-Bighash, "How many lattice points are there on a circle or a sphere?" in *A Pythagorean Introduction to Number Theory*. Springer, 2018, pp. 151–164.

[40] K.-K. Oh, M.-C. Park, and H.-S. Ahn, "A survey of multi-agent formation control," *Automatica*, vol. 53, pp. 424–440, 2015.

[41] S.-J. Chung, A. A. Paranjape, P. Dames, S. Shen, and V. Kumar, "A survey on aerial swarm robotics," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 837–855, 2018.

[42] J. Toner and Y. Tu, "Long-range order in a two-dimensional dynamical XY model: how birds fly together," *Physical review letters*, vol. 75, no. 23, p. 4326, 1995.

[43] J. Qi, R. Vazquez, and M. Krstic, "Multi-agent deployment in 3-D via PDE control," *IEEE Transactions on Automatic Control*, vol. 60, no. 4, pp. 891–906, 2014.

[44] M. R. Jovanovic and B. Bamieh, "On the ill-posedness of certain vehicular platoon control problems," *IEEE Transactions on Automatic Control*, vol. 50, no. 9, pp. 1307–1321, 2005.

[45] B. Bamieh, M. R. Jovanovic, P. Mitra, and S. Patterson, "Coherence in large-scale networks: Dimension-dependent limitations of local feedback," *IEEE Transactions on Automatic Control*, vol. 57, no. 9, pp. 2235–2249, 2012.

[46] G. B. Folland, *Introduction to partial differential equations*. Princeton university press, 2020.

Denis Nikitin received both the B.Sc. degree in 2016 and the M.Sc. degree in 2018 in Mathematics and Mechanics Faculty of Saint Petersburg State University, Saint Petersburg, Russia, specializing on the control theory and cybernetics. He won several international robotics competitions while being student and was a teacher of robotics in the Math&Phys Lyceum 239 in Saint Petersburg. He is currently a doctoral researcher at CNRS, GIPSA-Lab, Grenoble, France. His current research mainly focuses on control of large-scale systems.

Carlos Canudas-de-Wit (F'16) was born in Villahermosa, Mexico, in 1958. He received the B.S. degree in electronics and communications from the Monterrey Institute of Technology and Higher Education, Monterrey, Mexico, in 1980, and the M.S. and Ph.D. degrees in automatic control from the Department of Automatic Control, Grenoble Institute of Technology, Grenoble, France, in 1984 and 1987, respectively. Dr. Canudas-de-Wit is an IFAC Fellow and an IEEE Fellow. He was an Associate Editor of the IEEE Transactions on Automatic Control, Automatica, the IEEE Transactions on Control Systems Technology. He is an Associate Editor of the Asian Journal of Control and the IEEE Transactions on Control of Network Systems. He served as the President of the European Control Association from 2013 to 2015 and as member of the Board of Governors of the IEEE Control System Society from 2011 to 2014. He holds the ERC Advanced Grant Scale-FreeBack from 2016 to 2021.

Paolo Frasca (M'13–SM'18) received the Ph.D. degree from Politecnico di Torino, Turin, Italy, in 2009. After Postdoctoral appointments in Rome and in Turin, he has been an Assistant Professor with the University of Twente, Enschede, The Netherlands, from 2013 to 2016. Since October 2016, he has been CNRS Researcher with GIPSA-lab, Grenoble, France. His research interests cover the theory of network systems with applications to infrastructural and social networks. He has (co)authored more than 40 journal publications and the book Introduction to Averaging Dynamics Over Networks (Springer). He has been an Associate Editor for the Editorial Boards of several conferences and journals, including the IEEE Control Systems Letters, the Asian Journal of Control, and Automatica.