On Stabilization of Large-Scale Systems of Linear Hyperbolic PDEs via Continuum Approximation of Exact Backstepping Kernels*

Jukka-Pekka Humaloja[1] and Nikolaos Bekiaris-Liberis[1]

Abstract— **We establish that stabilization of a class of linear, hyperbolic PDEs with a large (nevertheless finite) number of components, can be achieved via employment of a backstepping-based control law, which is constructed for stabilization of a continuum version (i.e., as the number of components tends to infinity) of the PDE system. This is achieved by proving that the exact backstepping kernels, constructed for stabilization of the large-scale system, can be approximated (in certain sense such that exponential stability is preserved) by the backstepping kernels constructed for stabilization of a continuum version (essentially an infinite ensemble) of the original PDE system. The proof relies on construction of a convergent sequence of backstepping kernels that is defined such that each kernel matches the exact backstepping kernels (derived based on the original, large-scale system), in a piecewise constant manner with respect to an ensemble variable; while showing that they satisfy the continuum backstepping kernel equations. We present a numerical example that reveals that complexity of computation of stabilizing backstepping kernels may not scale with the number of components of the PDE state, when the kernels are constructed on the basis of the continuum version, in contrast to the case in which they are constructed on the basis of the original, large-scale system. Thus, this approach can be useful for design of computationally tractable, stabilizing backstepping-based control laws for large-scale PDE systems.**

I. INTRODUCTION

Large-scale systems of 1-D hyperbolic PDEs appear in a variety of applications involving transport phenomena and which incorporate different, interconnected components. Among them, large-scale interconnected hyperbolic systems may be used to describe the dynamics of blood flow, from the location of the heart all the way through to points where non-invasive measurements can be obtained [1], [2], of epidemics spreading, to describe transport of epidemics among different geographical regions [3], and of traffic flows, to model density and speed dynamics in interconnected highway segments [4], [5] and in urban networks [6], to name a few [7]. Backstepping is a systematic design approach to construction of explicit feedback laws for general classes of such systems [8], [9], [10], [11]. Due to potentially large number of interacting components, incorporated in such systems, computational complexity of exact backstepping-based control designs may increase significantly, proportionally to the number of state components. Motivated by this, we aim at developing an approach to computing backstepping kernels for large-scale hyperbolic PDEs, such that computational complexity remains tractable, even when the number of state components becomes very large, while at the same time, provably retaining the stability guarantees of backstepping. We achieve this via approximating the exact backstepping kernels, computed based on a large number of PDEs, utilizing a single kernel that is derived based on a continuum version of the exact kernels PDEs, and capitalizing on the robustness properties of backstepping for the class of systems considered, to additive control gain errors.

The approach of design of feedback laws for large-scale systems based on a continuum version of the system considered has been utilized for large-scale ODE systems, such as, for example, in [12], [13], [14], [15], [16]. However, such an approach has not been utilized so far for large-scale systems whose state components are PDEs. The main goals of the approach developed here, may be viewed also as related to control design approaches that aim at providing computational means towards implementation of PDE backstepping-based control laws with provable stability guarantees, such as, for example, neural operators-based [17], late-lumping-based [18], and power series-based [19], backstepping control laws. Our approach may be viewed as complementary and different from these results, in that the main goal is to address complexity due to a potential radical increase in the number of state components, instead of complexity of actual numerical implementation (even though these existing results can be combined with the approach presented here, for numerical implementation of the controllers).

In the present paper, we provide backstepping-based feedback laws for a class of large-scale systems of 1-D hyperbolic PDEs, which are described by the class of systems considered in [9], when the number of state components n is large. The key idea of our approach is to construct approximate backstepping kernels for stabilization of the large-scale (nevertheless, with a finite number of components) system relying on the continuum backstepping kernels developed in [20] for a continuum version of the original, large-scale system. We establish stability of the closed-loop system consisting of the original, large-scale PDE system under a backstepping-based feedback law that employs the approximate kernels, constructed based on the continuum version of the PDE system. The proof utilizes the well-posedness of the respective kernel equations, and the ability to approximate (in L^2) the parameters of the continuum kernel equations to arbitrary accuracy by step functions.

We then present a numerical example that illustrates that

*Funded by the European Union (ERC, C-NORA, 101088147). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

[1]The authors are with the Department of Electrical and Computer Engineering, Technical University of Crete, Chania, Greece. Emails: jhumaloja@tuc.gr and nlimperis@tuc.gr.

computation of (approximate) stabilizing kernels based on the continuum kernel may provide flexibility in computation, as well as it may significantly reduce computational complexity. In particular, in this specific example, although computation of the exact backstepping kernels may require to solve implicitly the corresponding $n+1$ hyperbolic kernels PDEs, as closed-form solutions may not be available, the approximate kernels can be computed with only algebraic computations, since the continuum kernel is available in closed form. We also present respective simulation investigations, which validate the theoretical developments, showing that as the number of components of the large-scale system increases, the performance of the closed-loop systems, under the approximate control laws, is improved.

We start in Sections II and III presenting both the large-scale PDE system and its continuum counterpart, together with the respective exact and continuum backstepping kernels PDEs. In Section IV we establish stability of the large-scale, closed-loop system under the approximate control law. In Section V we present a numerical example and consistent simulation results. In Section VI we provide concluding remarks and discuss related topics of our current research.

We use the standard notation $L^2(\Omega;\mathbb{R})$ for real-valued Lebesque integrable functions on a domain Ω. Similarly, $C(\Omega;\mathbb{R}), C^1(\Omega;\mathbb{R})$ denote continuous and continuously differentiable functions, respectively, on Ω. We denote by E the Hilbert space $L^2([0,1];\mathbb{R}^{n+1})$ equipped with the inner product

$$\left\langle \left(\begin{smallmatrix}\mathbf{u}_1\\ v_1\end{smallmatrix}\right), \left(\begin{smallmatrix}\mathbf{u}_2\\ v_2\end{smallmatrix}\right)\right\rangle_E = \int_0^1 \left(\frac{1}{n}\sum_{i=1}^n u_1^i(x)u_2^i(x) + v_1(x)v_2(x)\right)dx, \quad (1)$$

which induces the norm $\|\cdot\|_E = \sqrt{\langle\cdot,\cdot\rangle_E}$. Finally, a system is called exponentially stable (on E) if for any initial condition $z_0 \in E$ the (weak) solution of the system satisfies $\|z(t)\|_E \leq Me^{-ct}\|z_0\|_E$ for some $M, c > 0$.

II. Stabilization of Large-Scale Systems of Linear Hyperbolic PDEs via Backstepping

For $n \geq 1$ consider the following set of $n+1$ transport PDEs on $x \in [0,1]$ for $i = 1,\ldots,n$

$$u_t^i(t,x) + \lambda_i(x)u_x^i(t,x) = \frac{1}{n}\sum_{j=1}^n \sigma_{i,j}(x)u^j(t,x) + W_i(x)v(t,x), \quad (2a)$$

$$v_t(t,x) - \mu(x)v_x(t,x) = \frac{1}{n}\sum_{j=1}^n \theta_j(x)u^j(t,x), \quad (2b)$$

with boundary conditions

$$u^i(t,0) = q_i v(t,0), \qquad v(t,1) = U(t), \quad (3)$$

where $U \in L^2([0,m];\mathbb{R})$, for any $m \in \mathbb{N}$, is the control input. The initial conditions of (2) are $u^i(0,x) = u_0^i(x), v(0,x) = v_0(x)$, where $u_0^i, v_0 \in L^2([0,1];\mathbb{R})$. The parameters of the system (2), (3) satisfy the following assumption.

Assumption 2.1: We assume that $\mu, \lambda_i \in C^1([0,1];\mathbb{R})$, $\sigma_{i,j}, W_i, \theta_i \in C([0,1];\mathbb{R})$ and $q_i \in \mathbb{R}$ for all $i,j = 1,2,\ldots,n$. Moreover, the transport velocities are assumed to satisfy $-\mu(x) < 0 < \lambda_i(x)$, for all $x \in [0,1]$ and $i = 1,2,\ldots,n$.

Remark 2.2: The presentation of the system (2), (3) is motivated from [9]. However, here we also make the following modifications. Most notably, the factor $1/n$ appears in (2). This is equivalent to equipping the n-part of the system with the scaled inner product $n^{-1}\langle\cdot,\cdot\rangle_{\mathbb{R}^n}$. With the scaling, we guarantee that the sums remain bounded and convergent as $n \to \infty$ without having to pose any additional constraints on the parameters of (2), (3). If one wishes to proceed without scaling the sums, then some additional assumptions are needed, e.g., that the respective parameters form ℓ^p sequences for some $p \in [1,+\infty]$, such that the sums are well-defined as $n \to \infty$. Moreover, Assumption 2.1 is sufficient to guarantee well-posedness of (2), (3), e.g., based on the port-Hamiltonian approach [21, Sect. 13.2].

It follows from [9, Thm 3.2] that the system (2), (3) is exponentially stabilizable by a state feedback law of the form

$$U(t) = \int_0^1 \left[\frac{1}{n}\sum_{i=1}^n k^i(1,\xi)u^i(t,\xi) + k^{n+1}(1,\xi)v(t,\xi)\right]d\xi, \quad (4)$$

where, for $i = 1,\ldots,n+1$, k^i satisfy

$$\mu(x)k_x^i(x,\xi) - \lambda_i(\xi)k_\xi^i(x,\xi) = \lambda_i'(\xi)k^i(x,\xi) + \frac{1}{n}\sum_{j=1}^n \sigma_{j,i}(\xi)k^j(x,\xi) + \theta_i(\xi)k^{n+1}(x,\xi), \quad (5a)$$

$$\mu(x)k_x^{n+1}(x,\xi) + \mu(\xi)k_\xi^{n+1}(x,\xi) = -\mu'(\xi)k^{n+1}(x,\xi) + \frac{1}{n}\sum_{j=1}^n W_j(\xi)k^j(x,\xi), \quad (5b)$$

on a triangular domain $0 \leq \xi \leq x \leq 1$ with boundary conditions

$$k^i(x,x) = -\frac{\theta_i(x)}{\lambda_i(x)+\mu(x)}, \quad (6a)$$

$$\mu(0)k^{n+1}(x,0) = \frac{1}{n}\sum_{j=1}^n q_j\lambda_j(0)k^j(x,0), \quad (6b)$$

for all $x \in [0,1]$. Note the scaling of the sums by $1/n$ as per Remark 2.2.

III. Stabilization of a Continuum of Linear Hyperbolic PDEs via Continuum Backstepping

While large-scale, yet, consisting of a finite-number of components, systems of hyperbolic PDEs can be studied in the framework of Section II, we also consider the continuum limit case as $n \to \infty$, for which we present the generic framework of a continuum of hyperbolic PDEs studied in [20] and sketched in Fig. 1. That is, instead of having n rightward transport PDEs as in Section II, consider a continuum of such PDEs as in [20] with $y \in [0,1]$ being the index variable[2]

$$u_t(t,x,y)+\lambda(x,y)u_x(t,x,y)=\int_0^1 \sigma(x,y,\eta)u(t,x,\eta)d\eta + W(x,y)v(t,x), \tag{7a}$$

$$v_t(t,x)-\mu(x)v_x(t,x)=\int_0^1 \theta(x,y)u(t,x,y)dy, \tag{7b}$$

with boundary conditions

$$u(t,0,y)=q(y)v(t,0), \qquad v(t,1)=U(t), \tag{8}$$

for almost every $y\in[0,1]$, i.e., the continuum variables and parameters are considered $L^2([0,1];\mathbb{R})$ functions in y. The following assumption is needed, for the parameters involved in (7), (8), to guarantee the existence of a continuum backstepping control law [20, Thm 3].

Assumption 3.1: We assume that $\mu\in C^1([0,1];\mathbb{R}), \lambda\in C^1\left([0,1]^2;\mathbb{R}\right), W,\theta \in C([0,1];\mathbb{R})\times L^2([0,1];\mathbb{R}), \sigma \in C([0,1];\mathbb{R})\times L^2\left([0,1]^2;\mathbb{R}\right)$ and $q\in C([0,1];\mathbb{R})$. Moreover, $\lambda(x,y)>0$ uniformly for all $x\in[0,1]$ and almost every $y\in[0,1]$, and $-\mu(x)<0$ for all $x\in[0,1]$.

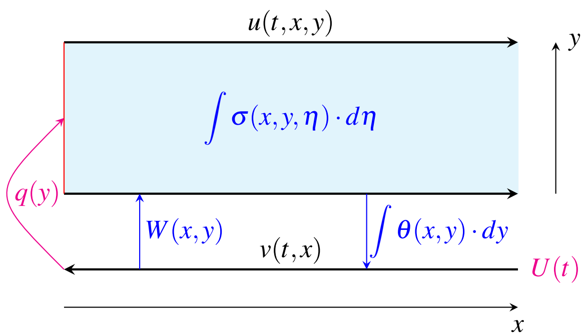

Fig. 1. Schematic view of the continuum PDE system (7), (8). Boundary terms are denoted in magenta and in-domain terms are denoted in blue.

By [20, Thm 1], the system (7), (8) is exponentially stabilizable by a state feedback law of the form

$$U(t)=\int_0^1\left[\int_0^1 k(1,\xi,y)u(t,\xi,y)dy+\bar{k}(1,\xi)v(t,\xi)\right]d\xi, \tag{9}$$

where $k,\bar{k}$ satisfy

$$\mu(x)k_x(x,\xi,y)-\lambda(\xi,y)k_\xi(x,\xi,y)-\theta(\xi,y)\bar{k}(x,\xi)= \lambda_\xi(\xi,y)k(x,\xi,y)+\int_0^1\sigma(\xi,\eta,y)k(x,\xi,\eta)d\eta, \tag{10a}$$

$$\mu(x)\bar{k}_x(x,\xi)+\mu(\xi)\bar{k}_\xi(x,\xi)= -\mu'(\xi)\bar{k}(x,\xi)+\int_0^1 W(\xi,y)k(x,\xi,y)dy, \tag{10b}$$

[2] In the scope of this paper, we do not formally consider whether (7) is the continuum limit of (2) as $n\to\infty$, but rather use (7) as an educated guess to obtain a continuum version of the respective backstepping kernels $k^i, i=1,\dots,n$, given in (5).

on a triangular domain $0\le\xi\le x\le 1$ with boundary conditions

$$k(x,x,y)=-\frac{\theta(x,y)}{\lambda(x,y)+\mu(x)}, \tag{11a}$$

$$\mu(0)\bar{k}(x,0)=\int_0^1 q(y)\lambda(0,y)k(x,0,y)dy, \tag{11b}$$

for almost every $y\in[0,1]$.

IV. Stabilization of the Large-Scale System via Continuum Approximation of Exact Kernels

A. Statement of the Main Result

The core idea of the continuum approximation that we present here is that we approximate $n+1$ kernel equations by a continuum of kernel equations. Provided that the approximation is sufficiently accurate, we show that the backstepping controller derived from the continuum kernel equations exponentially stabilizes the $n+1$ system associated with the original finite system of $n+1$ PDEs.

Thus, consider an $n+1$ system (2), (3) with parameters $\lambda_i, W_i, \theta_i, \sigma_{i,j}, q_i$ for $i,j=1,2,\dots,n$ satisfying Assumption 2.1 and consider any continuous functions $\lambda,W,\theta,\sigma,q$ that satisfy Assumption 3.1 with

$$\lambda(x,i/n)=\lambda_i(x), \tag{12a}$$

$$W(x,i/n)=W_i(x), \tag{12b}$$

$$\theta(x,i/n)=\theta_i(x), \tag{12c}$$

$$\sigma(x,i/n,j/n)=\sigma_{i,j}(x), \tag{12d}$$

$$q(i/n)=q_i, \tag{12e}$$

for all $x\in[0,1]$ and $i,j=1,2,\dots,n$. There are infinitely many functions satisfying (12) and Assumption 3.1, as well as ways to construct them, e.g., by utilizing auxiliary functions $\{p_i\}_{i=1}^n$ that satisfy $p_i(i/n)=1$ and $p_i(\ell/n)=0$ for $\ell\neq i$. The relations in (12) could as well be defined in other ways, e.g., using $(i-1)/n$ in place of i/n, but we find (12) the most convenient option for our developments. The continuum kernel equations (10), (11) with parameters $\lambda,\mu,W,\theta,\sigma,q$, satisfying (12) and Assumption 3.1 have a unique, continuous solution $(k,\bar{k})$ by [20, Thm 3]. Thus, construct the following functions for all $0\le\xi\le x\le 1$

$$\widetilde{k}^i(x,\xi)=k(x,\xi,i/n), \quad i=1,2,\dots,n, \tag{13a}$$

$$\widetilde{k}^{n+1}(x,\xi)=\bar{k}(x,\xi). \tag{13b}$$

Our main result is the following.

Theorem 4.1: Consider an $n+1$ system (2), (3) with parameters $\lambda_i,\mu,W_i,\theta_i,\sigma_{i,j},q_i$ for $i,j=1,2,\dots,n$ satisfying Assumption 2.1. Let the parameters $\lambda,\mu,W,\theta,\sigma,q$ satisfy Assumption 3.1 and relations (12). Then

$$U(t)=\int_0^1\left[\frac{1}{n}\sum_{i=1}^n \widetilde{k}^i(1,\xi)u^i(t,\xi)+\widetilde{k}^{n+1}(1,\xi)v(t,\xi)\right]d\xi, \tag{14}$$

where $\left(\widetilde{k}^i\right)_{i=1}^{n+1}$ are given in (13), with $(k,\bar{k})$ being the solution to (10), (11), exponentially stabilizes system (2), (3).

B. Proof of Theorem 4.1

The proof of Theorem 4.1 relies on Lemmas 4.2 and 4.3 presented below. We show first that the functions defined in (13) approximate the solutions to the $n+1$ kernel equations (5), (6) to arbitrary accuracy as n increases. In order to do this, we first interpret the solutions to the $n+1$ kernels equations (5), (6) as piecewise constant solutions with respect to y, to the continuum kernels equations (10), (11). One way to do this is highlighted in the following lemma, which is to transform the $\mathbb{R}^n$-valued components of the $n+1$ kernel equations (5), (6) into step functions in $y \in [0,1]$.

Lemma 4.2: Consider the $n+1$ kernel equations (5), (6) where the parameters satisfy Assumption 2.1 and define the following functions for all $x \in [0,1]$, piecewise in y for $i, j = 1, 2, \ldots, n$[3]

$$\lambda^n(x,y) = \lambda_i(x), \quad y \in ((i-1)/n, i/n], \tag{15a}$$

$$\sigma^n(x,y,\eta) = \sigma_{i,j}(x), \quad y \in ((i-1)/n, i/n], \tag{15b}$$

$$\eta \in ((j-1)/n, j/n], \tag{15c}$$

$$W^n(x,y) = W_i(x), \quad y \in ((i-1)/n, i/n], \tag{15d}$$

$$\theta^n(x,y) = \theta_i(x), \quad y \in ((i-1)/n, i/n], \tag{15e}$$

$$q^n(y) = q_i, \quad y \in ((i-1)/n, i/n]. \tag{15f}$$

Construct the following function for all $0 \le \xi \le x \le 1$, piecewise in y for $i = 1, 2, \ldots, n$

$$k^n(x,\xi,y) = k^i(x,\xi), \quad y \in ((i-1)/n, i/n], \tag{16}$$

where $\left(k^i\right)_{i=1}^{n+1}$ is the solution to (5), (6). Then, $\left(k^n, k^{n+1}\right)$ satisfies the kernel equations (10), (11) for the parameters defined in (15) and the original μ.

Proof: See the proof of [22, Lem. 4.2]. ∎

Let us next consider the continuum of kernel equations (10), (11) with continuous parameters $\lambda, \mu, W, \theta, \sigma, q$ that satisfy (12) and Assumption 3.1, together with the respective kernel equations (10), (11) with piecewise constant parameters $\lambda^n, \mu, W^n, \theta^n, \sigma^n, q^n$ in y constructed in Lemma 4.2. In the next lemma, we show that the solution $\left(k^n, \bar{k}^n\right)$ to the latter approximates the solution $\left(k, \bar{k}\right)$ to the former to arbitrary accuracy, provided that n is sufficiently large.

Lemma 4.3: Consider the solutions $\left(k^n, \bar{k}^n\right)$ to the kernel equations (10), (11) with parameters $\lambda^n, \mu, W^n, \theta^n, \sigma^n, q^n$ from Lemma 4.2. There exist continuous parameters $\lambda, \mu, W, \theta, \sigma, q$ constructed such that they satisfy Assumption 3.1 and (12), and for any such parameters the solution $\left(k, \bar{k}\right)$ to the respective kernel equations (10), (11) exists and satisfies the following implications. For any $\delta > 0$, there exists an $n_\delta \in \mathbb{N}$ such that for all $n \ge n_\delta$ we have

$$\max_{(x,\xi)\in\mathscr{T}} \|k(x,\xi,\cdot) - k^n(x,\xi,\cdot)\|_{L^2([0,1];\mathbb{R})} \le \delta, \tag{17a}$$

$$\max_{(x,\xi)\in\mathscr{T}} |\bar{k}(x,\xi) - \bar{k}^n(x,\xi)| \le \delta, \tag{17b}$$

where we denote $\mathscr{T} = \{(x,\xi) \in [0,1]^2 : 0 \le \xi \le x \le 1\}$.

[3]These functions can be extended to $y \in [0,1]$ by assigning the value at $y = 0$ arbitrarily, which does not affect the functions in the L^2 sense. The same applies to (16).

Proof: See the proof of [22, Lem. 4.3]. ∎

We have by now established convergence of the solutions $\left(k^n, \bar{k}^n\right)$ to the kernel equations (10), (11) with parameters $\lambda^n, \mu, W^n, \theta^n, \sigma^n, q^n$ defined in (15), to the solutions $\left(k, \bar{k}\right)$ to (10), (11) with parameters $\lambda, \mu, W, \theta, \sigma, q$. Since the solutions $\left(k^n, \bar{k}^n\right)$ are piecewise constant in y satisfying (16) and (17), where $\left(k^i\right)_{k=1}^{n+1}$ are the solutions to (5), (6) with parameters $\left(\lambda^i\right)_{i=1}^n, \mu, \left(W^i\right)_{i=1}^n, \left(\theta\right)_{i=1}^n, \left(q^i\right)_{i=1}^n$, the solutions $\left(k^n, \bar{k}^n\right)$ can, in fact, approximate the kernels $\left(k^i\right)_{i=1}^{n+1}$ to arbitrary accuracy as n gets sufficiently large. This in turn implies that the control law (14), constructed based on the solutions $\left(k, \bar{k}\right)$ to the continuum kernel equations (10), (11), approximates (arbitrarily close as n gets sufficiently large) the original control law (4) constructed based on the solutions $\left(k^i\right)_{i=1}^{n+1}$ to the kernels equations (5), (6). For this, we present the following lemma.

Lemma 4.4: The control law (14) can be written as

$$\begin{aligned} U(t) = &\int_0^1 \left(\frac{1}{n} \sum_{i=1}^n k^i(1,\xi) u^i(t,\xi) + k^{n+1}(1,\xi) v(t,\xi) \right) d\xi \\ &+ \int_0^1 \left(\frac{1}{n} \sum_{i=1}^n \Delta k^i(1,\xi) u^i(t,\xi) + \Delta k^{n+1}(1,\xi) v(t,\xi) \right) d\xi, \end{aligned} \tag{18}$$

where $\left(k^i\right)_{i=1}^{n+1}$ is the solution to the $n+1$ kernel equations (5), (6), and the approximation error terms $\left(\Delta k^i(1,\cdot)\right)_{i=1}^{n+1}$ become arbitrarily small, uniformly in $\xi \in [0,1]$, when n is sufficiently large.

Proof: See the proof of [22, Lem. 4.5]. ∎

Proof of Theorem 4.1. By Lemma 4.4, the control law (14) can be split into the part that exponentially stabilizes the large-scale $n+1$ system (2), (3) and to the Δ-part which we treat as a perturbation that becomes arbitrarily small when n is sufficiently large. Thus, the stability of the $n+1$ system under the control law (18) can be established based on existing results for well-posed infinite-dimensional linear systems, e.g., as in [23, Sect. IV.C]. For details, we refer to the proof of [22, Thm 4.1].

V. NUMERICAL EXAMPLE

As an example, consider an $n+1$ system (2), (3) with parameters $\mu(x) = 1, \lambda_i(x) = 1$, and

$$\sigma_{i,j}(x) = x^3(x+1)\left(\frac{i}{n} - \frac{1}{2}\right)\left(\frac{j}{n} - \frac{1}{2}\right), \tag{19a}$$

$$W_i(x) = x(x+1)e^x\left(\frac{i}{n} - \frac{1}{2}\right), \tag{19b}$$

$$\theta_i(x) = -70e^{x\frac{35}{\pi^2}}\frac{i}{n}\left(\frac{i}{n} - 1\right), \tag{19c}$$

$$q_i = \cos\left(2\pi\frac{i}{n}\right), \tag{19d}$$

for $i,j=1,\dots,n$ such that continuous functions satisfying (12) can be constructed as $\lambda(x,y)=1$, and

$$\sigma(x,y,\eta)=x^3(x+1)\left(y-\frac{1}{2}\right)\left(\eta-\frac{1}{2}\right), \tag{20a}$$

$$W(x,y)=x(x+1)e^x\left(y-\frac{1}{2}\right), \tag{20b}$$

$$\theta(x,y)=-70e^{x\frac{35}{\pi^2}}y(y-1), \tag{20c}$$

$$q(y)=\cos(2\pi y). \tag{20d}$$

The latter parameter values correspond to the example considered in [20, Sect. VII], where it is shown that the solutions of the corresponding continuum kernel equations are given by

$$k(x,\xi,y)=35y(y-1)e^{2\xi\bar{k}(x,\xi)}, \tag{21a}$$

$$\bar{k}(x,\xi)=\frac{35}{2\pi^2}. \tag{21b}$$

We note that while a closed-form solution exists to the continuum kernel equations (10), (11) with parameters (20), we were not able to solve the corresponding $n+1$ kernel equations (5), (6) with parameters (19) in closed-form when $n\in\mathbb{N}$ is arbitrary, nor do we expect that a closed-form solution can be constructed (even for small n). This is also consistent with, e.g., [24], in which an explicit solution is possible to obtain for the specific case $n=1$ and for spatially invariant parameters (19). Regardless, in this particular example, the continuum approximation significantly simplifies the computation of the stabilizing control kernels.

We simulate the $n+1$ system (2), (3) with parameters (19) under the control law (14) computed based on the continuum kernels (21). Various values of n are considered to illustrate the behavior of the closed-loop system as n increases. In fact, the closed-loop system is stable for any $n\geq 2$, but the performance is improved for larger n. However, when $n=1$, the system (2), (3) is open-loop stable and $k^1(x,\xi)=k^2(x,\xi)=0$ is the solution to the kernel equations (5), (6), in which case the approximate control law (14) destabilizes the system.

In the simulations, the system (2), (3) is approximated using finite differences with 256 grid points in $x\in[0,1]$. The ODE resulting from the finite-difference approximation is solved using `ode45` in MATLAB. The initial conditions for all n are $u_0^i(x)=q_i$ for $i=1,\dots,n$ and $v_0(x)=1$. Fig. 2 displays the control law (14) when $n=2,\dots,6$. We note that the control law also acts as a weighted average of the solution to (2), (3), i.e., we can also assess the exponential decay rate of the solutions based on $U(t)$. However, as $k(x,\xi,1)=0$ in (21), the component $u^n(t,x)$ does not affect the control in any way. Therefore, $u^n(t,x)$ is displayed separately in Fig. 3 for $n=2,\dots,5$, which shows that also $u^n(t,x)$ decays to zero as $t\to\infty$.

Fig. 2 and Fig. 3 show that the approximate control law based on the continuum kernels (21) is indeed stabilizing already when $n=2$, even if the rate of decay is very slow. However, Fig. 2 and Fig. 3 show that the closed-loop performance significantly improves when n becomes

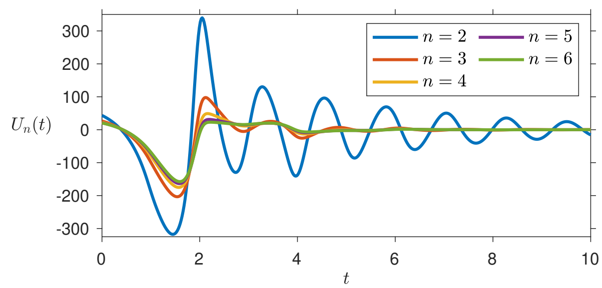

Fig. 2. The controls $U(t)$ based on the approximate control law (14) when $n=2,\dots,6$.

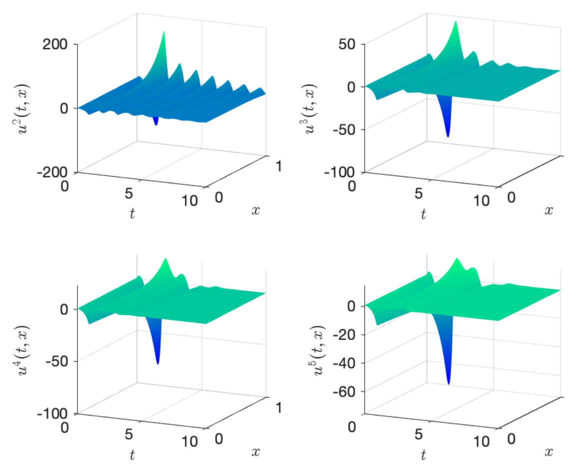

Fig. 3. The solution component $u^n(t,x)$ when $n=2,\dots,5$.

larger, and in Fig. 2 the controls for $n=5$ and $n=6$ are virtually indistinguishable. However, as we consider larger values of n separately in Fig. 4 and Fig. 5, some changes in $U(t)$ and $u^n(t,\xi)$ are still noticeable between $n=6$ and $n=10$. Regardless, in all studied cases beyond $n>5$, the controls, along with the solutions, have practically converged to zero by $t=6$. We note that it takes the control input $1/\mu+1/\lambda=2$ time units to traverse through the system (2), (3) for any n, which is why, e.g., the state component $u^n(t,\xi)$, may grow rapidly in the beginning of the simulation as seen in Fig. 3 and Fig. 5 before getting stabilized by the controls. However, due to the in-domain coupling between u and v in (2), the controls do affect the u^i components through v already before entering the u^i channels at time $t=1/\mu=1$.

Overall, the simulations demonstrate that the approximate control law (14) based on the continuum kernels (21) exponentially stabilizes the $n+1$ system (2), (3) when n is sufficiently large, and that the approximation error of the control law decreases as n increases. Thus, the simulations are well in accordance with the theoretical results. Moreover, in this example the approximate control law had good performance already for very moderate n, showing that the sufficiently large n appearing in the theoretical results may be, in practice, relatively small. However, one should not expect this to be the case in general, as this is dependent on

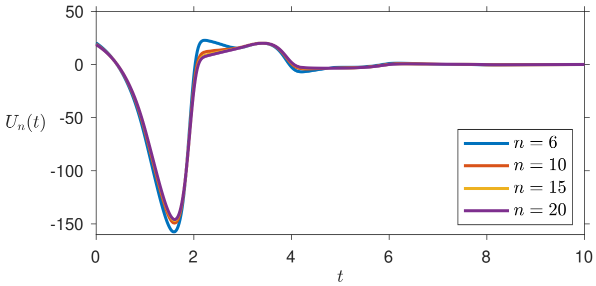

Fig. 4. The controls $U(t)$ based on the approximate control law (14) when $n = 6, 10, 15, 20$.

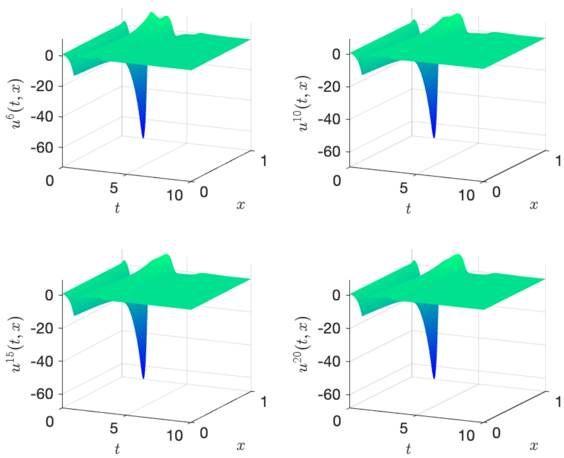

Fig. 5. The solution component $u^n(t,x)$ when $n = 6, 10, 15, 20$.

the parameters of both the system (2), (3) and the continuum approximation.

VI. Conclusions and Discussion

Computation of approximate stabilizing kernels based on the continuum kernel may provide flexibility in computation, as well as it may significantly improve computational complexity (although, practically, such computation also depends on the sampling method chosen for the continuum kernel). This is confirmed in the numerical example in which computational burden of stabilizing kernels is significantly improved, since the approximate kernels computed based on the continuum can be computed in closed form, in contrast to the exact kernels that would have to be computed implicitly based on the solution to the kernel PDEs.

In general, we may expect that the complexity of computation of stabilizing control gains via the continuum approximation approach to not scale with n, i.e., to be $\mathcal{O}(1)$; while the complexity of computation of the exact control kernels to be $\mathcal{O}(n)$, i.e., to grow with the number of state components. Thus, this approach may be useful for computationally efficient control of large-scale PDE systems.

References

[1] V. Bikia, "Non-invasive monitoring of key hemodynamical and cardiac parameters using physics-based modelling and artificial intelligence," Ph.D. dissertation, EPFL, 2021.

[2] P. Reymond, F. Merenda, F. Perren, D. Rufenacht, and N. Stergiopulos, "Validation of a one-dimensional model of the systemic arterial tree," *Am. J. Physiol. Heart Circ. Physiol.*, vol. 297, pp. H208–H222, 2009.

[3] L. Guan, C. Prieur, L. Zhang, C. Prieur, D. Georges, and P. Bellemain, "Transport effect of COVID-19 pandemic in France," *Annu. Rev. Control*, vol. 50, pp. 394–408, 2020.

[4] J. Friedrich, S. Göttlich, and M. Osztfalk, "Network models for nonlocal traffic flow," *ESAIM Math. Model. Numer. Anal.*, vol. 56, no. 1, pp. 213–235, 2022.

[5] L. Zhang, H. Luan, Y. Lu, and C. Prieur, "Boundary feedback stabilization of freeway trafficnetworks: ISS control and experiments," *IEEE Trans. Control Syst. Technol.*, vol. 30, pp. 997–1008, 2022.

[6] L. Tumash, C. Canudas-de-Wit, and M. L. Delle Monache, "Multi-directional continuous traffic model for large-scale urban networks," *Transportation Research Part B: Methodological*, vol. 158, pp. 374–402, 2022.

[7] G. Bastin and J.-M. Coron, *Stability and Boundary Sabilization of 1-D Hyperbolic Systems*. Birkhäuser/Springer, [Cham], 2016.

[8] M. Krstic and A. Smyshlyaev, *Boundary Control of PDEs: A Course on Backstepping Designs*. SIAM, 2008.

[9] F. Di Meglio, R. Vazquez, and M. Krstic, "Stabilization of a system of $n+1$ coupled first-order hyperbolic linear PDEs with a single boundary input," *IEEE Trans. Automat. Control*, vol. 58, no. 12, pp. 3097–3111, 2013.

[10] J. Redaud, J. Auriol, and S.-I. Niculescu, "Output-feedback control of an underactuated network of interconnected hyperbolic PDE-ODE systems," *Syst. Control Lett.*, vol. 154, p. Paper No. 104984, 2021.

[11] J. Auriol and D. Bresch-Pietri, "Robust state-feedback stabilization of an underactuated network of interconnected $n+m$ hyperbolic PDE systems," *Automatica*, vol. 136, p. Paper No. 110040, 2022.

[12] G. Ferrari-Trecate, A. Buffa, and M. Gati, "Analysis of coordinations in multi-agent systems through partial difference equations," *IEEE Trans. Automat. Control*, vol. 51, no. 6, pp. 1058–1063, 2006.

[13] T. Meurer and M. Krstic, "Finite-time multi-agent deployment: A nonlinear PDE motion planning approach," *Automatica*, vol. 37, pp. 2534–2542, 2011.

[14] D. Nikitin, C. Canudas-de-Wit, and P. Frasca, "A continuation method for large-scale modeling and control: from ODEs to PDE, a round trip," *IEEE Trans. Automat. Control*, vol. 67, pp. 5118–5133, 2022.

[15] J. Wei, E. Fridman, and K. H. Johansson, "A PDE approach to deployment of mobile agents under leader relative position measurements," *Automatica*, vol. 106, pp. 47–53, 2019.

[16] J. Zhang, R. Vazquez, J. Qi, and M. Krstic, "Multi-agent deployment in 3-D via reaction-diffusion system with radially-varying reaction," *Automatica*, vol. 161, p. 111491, 2024.

[17] L. Bhan, Y. Shi, and M. Krstic, "Neural operators for bypassing gain and control computations in PDE backstepping," *IEEE Trans. Automat. Control*, 2024.

[18] J. Auriol, K. A. Morris, and F. Di Meglio, "Late-lumping backstepping control of partial differential equations," *Automatica*, vol. 100, pp. 247–259, 2019.

[19] R. Vazquez, G. Chen, J. Qiao, and M. Krstic, "The power series method to compute backstepping kernel gains: theory and practice," in *IEEE Conference on Decision and Control*, 2023, pp. 8162–8169.

[20] V. Alleaume and M. Krstic, "Ensembles of hyperbolic PDEs: Stabilization by backstepping," *IEEE Trans. Automat. Control*, pp. 1–16, 2024.

[21] B. Jacob and H. Zwart, *Linear Port-Hamiltonian Systems on Infinite-dimensional Spaces*, ser. Operator Theory: Advances and Applications. Birkhäuser, 2012, vol. 223.

[22] J.-P. Humaloja and N. Bekiaris-Liberis, "Stabilization of a class of large-scale systems of linear hyperbolic PDEs via continuum approximation of exact backstepping kernels," arXiv, 2403.19455, 2024.

[23] L. Paunonen and J.-P. Humaloja, "On robust regulation of PDEs: from abstract methods to PDE controllers," in *IEEE Conference on Decision and Control*, 2022, pp. 7352–7357.

[24] R. Vazquez and M. Krstic, "Marcum Q-functions and explicit kernels for stabilization of 2×2 linear hyperbolic systems with constant coefficients," *Syst. Control Lett.*, vol. 68, pp. 33–42, 2014.