

A Novel Physics-Informed Recurrent Neural Network Approach for State Estimation of Autonomous Platforms

1st Adolfo Perrusquía

*School of Aerospace, Transport and Manufacturing
Cranfield University
Bedford, UK
adolfo.perrusquia-guzman@cranfield.ac.uk*

2nd Weisi Guo

*School of Aerospace, Transport and Manufacturing
Cranfield University
Bedford, UK
weisi.guo@cranfield.ac.uk*

Abstract—State estimation of autonomous platforms is a crucial element in the design and test of perception algorithms. The nonlinear nature of autonomous platforms makes hard to design accurate state estimation algorithms without using linearization techniques, large amount of data or knowledge of the physical parameters of the platform. This paper reports a novel state estimation algorithm of autonomous platforms. The proposed approach is based on a physics informed recurrent neural network (PIRNN) that combines the power of recurrent nets with an estimate structure of the autonomous platform model. This estimated model regularises the weights' manifold of the network for the accurate estimation of the states. Boundedness of the proposed PIRNN is verified using Lyapunov stability theory as long as the physics-informed signals satisfy a persistent of excitation condition. Simulations are conducted to test the PIRNN model and show its benefits and challenges.

Index Terms—state estimation, autonomous platform, physics informed recurrent neural network, memory terms, persistent of excitation

I. INTRODUCTION

State estimation is a well known research topic in control and signal processing communities that has increased its relevance due to the proliferation of autonomous platforms. The main goal of state estimation algorithms is to estimate the states of the system using only the observable states and a high fidelity model [1], [2], whilst reducing the variance associated to noise in sensors' measurements [3], [4]. The state estimation process is crucial for the design of controllers [5], [6] and for data collection in modern data-driven [7] and machine learning architectures [8]–[10].

State observers [11], [12] and Kalman filters are the most common techniques used for state estimation. Here, Kalman filter and its variants [13], [14] are widely used as a signal processing module in on-board control platforms embedded in the autonomous platform [15]–[17]. The Kalman filter module uses a simplistic kinematic model under a Gaussian distributed noise [18]–[20] assumption as a prior model to estimate the unobservable states. However, this simplistic model cannot cover all the properties of the real dynamics, such that biased state estimates are obtained and, in the worst case, the algorithm diverges [21]. KalmanNet [22] has been

used to estimate the covariance matrices associated to both the model and measurement noise. This is done by combining neural networks with Kalman filter such that prior models with low representation capabilities are enhanced with a more precise Gaussian distributed noise [23]. Nevertheless, poor state estimations can be obtained due to the estimation of a large Gaussian distribution which increases the variance and deteriorate the state estimates.

Recurrent neural networks (RNN) [24], [25] have been applied to model a nonlinear system dynamics as a sum of a linear system with stable eigenvalues and a feed-forward neural network as nonlinear compensator [26]. The approach uses s-shaped functions as activation functions (e.g., sigmoid and hyperbolic tangent functions) to model the non-linearities of the real model. However, the approach is quite sensitive to the proposed stable linear system. Furthermore, the learned weights and network structure do not have a physical relation with the real system which may pose a challenge in the generalization power of the network.

In the last decade, physics-informed neural networks (PINN) [27] have been adopted to incorporate prior information of the system dynamics to regularize the learning process, whilst satisfying the governing physical laws. This allows to improve the states estimation with noise suppression. Three different physics informed architectures can be found in the literature [28] which are either applied: i) as a regularization term in the loss function, ii) as physical laws that ensures the learning weights have a physical meaning [29], and iii) as a model structure that constraints the weights manifold [30]. Here, the main assumption of PINN is knowledge of the system parameters and structure, which in most of cases are unknown or partially known. In this scenario, inaccurate models produce biased estimates that can compromise the performance of control architectures or machine learning algorithms [4], [31].

Parameter identification algorithms [32]–[34] are used to estimate the parameters of a given system using either gradient [35], [36] or least-squares rules [37]. However, noisy signals degenerate the parameter estimates values. This issue has

been solved by incorporating well-designed filters to reduce the noise, but with the compromise of adding some time-delay or losing information. Additionally, these models require measurements of the time-derivative of the states which are usually not available or require additional sensors that produce high levels of noise.

In light of the above, the current challenges in state estimation algorithms and neural networks approaches are: 1) an accurate model is required as a prior to estimate accurately the complete states vector, 2) states measurements with enough richness are required to cover all the properties of the autonomous platform, 3) basis functions or activation functions in neural networks lacks of physical meaning, and 4) linearisation methods or large amount of data are required to construct a local linear model that captures the dynamic properties of the nonlinear model.

To alleviate the aforementioned issues, a novel physics-informed recurrent neural network (PIRNN) for state estimation is proposed. The approach constructs an RNN based on a physics-informed structure obtained from a state parameterization of the autonomous platform nonlinear differential equation solution. This parameterization is obtained from an estimated model of the autonomous platform. The states of the estimated model are noise free and feed the RNN for the weights learning. The PIRNN does not require knowledge of the parameters or noisy signals. Stability and boundedness of the PIRNN architecture is verified using Lyapunov stability theory as long as the physics-informed signals satisfy a persistent of excitation condition. The combined contribution of this paper with respect to previous works in state estimation of autonomous platforms are stated as follows:

- A novel physics-informed recurrent neural network (PIRNN) for state estimation is proposed.
- The physics-informed element appears as memory functions that feed the RNN to regularise the weights' learning manifold.
- The basis/activation functions of the proposed PIRNN are constructed from the nonlinear structure of the autonomous platform.
- Neural weights convergence is achieved under the fulfilment of a persistent of excitation (PE) condition.

The paper outline is as follows: Section II provides the preliminaries of the problem. Section III establishes the physics-informed recurrent neural network architecture for state estimation under a novel parameterization of the solution of the autonomous platform nonlinear differential equation. Section IV reports the simulation studies using two different autonomous platforms: a 1 degree-of-freedom (DOF) robot a 3-DOF drone's attitude dynamics. The conclusions are presented in Section V.

II. PRELIMINARIES

The autonomous platforms discussed in this paper fulfil the following nonlinear system structure

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}), \quad \mathbf{x}(t_0) = \mathbf{x}_0, \\ \mathbf{z} &= \mathbf{x} + \boldsymbol{\omega}, \quad \boldsymbol{\omega} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}), \end{aligned} \quad (1)$$

where $\mathbf{x} \in \mathbb{R}^n$ defines the state vector, $\mathbf{u} \in \mathbb{R}^m$ is a known control input vector, $\mathbf{z} \in \mathbb{R}^n$ defines the output linear measurements, $\boldsymbol{\omega} \in \mathbb{R}^n$ denotes a Gaussian distributed noise with a zero mean vector and covariance $\mathbf{R} = \mathbf{R}^\top \succ 0 \in \mathbb{R}^{n \times n}$, $\mathbf{f}(\mathbf{x}, \mathbf{u}) \in \mathbb{R}^n$ is the autonomous platform dynamics in function of both the state and control vectors.

The following assumption is required for the construction of the proposed approach,

Assumption 1: The function $\mathbf{f}(\mathbf{x}, \mathbf{u})$ is Locally Lipschitz and $\mathbf{f}(\mathbf{0}) = \mathbf{0}$. Additionally, the structure of $\mathbf{f}(\mathbf{x}, \mathbf{u})$ is assumed to be known such that it can be written as linear combination of known basis functions $\{\phi_1(\mathbf{x}, \mathbf{u}), \dots, \phi_r(\mathbf{x}, \mathbf{u})\}$.

According to Assumption 1 and the Weierstrass approximation theorem [38], the nonlinear function $\mathbf{f}(\mathbf{x}, \mathbf{u})$ can be linearly approximated by a matrix of basis functions $\boldsymbol{\Phi}(\mathbf{x}, \mathbf{u}) \in \mathbb{R}^r$ as follows

$$\mathbf{f}(\mathbf{x}, \mathbf{u}) = \mathbf{W}^\top \boldsymbol{\Phi}(\mathbf{x}, \mathbf{u}), \quad (2)$$

where $\mathbf{W} \in \mathbb{R}^{r \times n}$ defines an unknown weights matrix with constant parameters and $\boldsymbol{\Phi}(\mathbf{x}, \mathbf{u})$ is a vector composed by the basis functions $\{\phi_1(\mathbf{x}, \mathbf{u}), \dots, \phi_r(\mathbf{x}, \mathbf{u})\}$. The nonlinear autonomous platform dynamics (1) is equivalently written as

$$\dot{\mathbf{x}} = \mathbf{W}^\top \boldsymbol{\Phi}(\mathbf{x}, \mathbf{u}). \quad (3)$$

Remark 1: The basis functions $\{\phi_1(\mathbf{x}, \mathbf{u}), \dots, \phi_r(\mathbf{x}, \mathbf{u})\}$ define a physics informed structure that is used to regularize the weights' learning manifold.

The challenge for state estimation of nonlinear autonomous platforms is that most of the algorithmic tools require unmeasurable signals given by the time-derivative of the states [30]. This avoid the construction of standard system identification or state estimation techniques. A state parameterization [12], [39] is used to write the nonlinear differential equation (3) in terms of its solution and the available states measurements $\mathbf{z}$. This is achieved by incorporating the following linear dynamics into (3) as

$$\dot{\mathbf{x}} = -\boldsymbol{\Lambda} \mathbf{x} + \mathbf{W}^\top \boldsymbol{\Phi}(\mathbf{x}, \mathbf{u}) + \boldsymbol{\Lambda} \mathbf{x}, \quad (4)$$

where $\boldsymbol{\Lambda} = \lambda \mathbf{I}_n \in \mathbb{R}^{n \times n}$ is a positive definite diagonal matrix with values $\lambda > 0$. This new term helps to construct a solution of the differential equation which exhibits input-to-state stability (ISS) [40]. The solution of (4) is

$$\begin{aligned} \mathbf{x}(t) &= e^{-\boldsymbol{\Lambda}(t-t_0)} \mathbf{x}(t_0) + \mathbf{W}^\top \int_{t_0}^t e^{\lambda(\sigma-t)} \boldsymbol{\Phi}(\mathbf{x}(\sigma), \mathbf{u}(\sigma)) d\sigma \\ &\quad + \boldsymbol{\Lambda} \int_{t_0}^t e^{\lambda(\sigma-t)} \mathbf{x}(\sigma) d\sigma. \end{aligned}$$

Define

$$\begin{aligned} \mathbf{m}(\mathbf{x}) &:= \int_{t_0}^t e^{\lambda(\sigma-t)} \boldsymbol{\Phi}(\mathbf{x}(\sigma), \mathbf{u}(\sigma)) d\sigma \\ \mathbf{n}(\mathbf{x}) &:= \int_{t_0}^t e^{\lambda(\sigma-t)} \mathbf{x}(\sigma) d\sigma. \end{aligned}$$

The solution of the autonomous platform parameterization in (1) is rewritten as

$$\mathbf{x}(t) = e^{-\boldsymbol{\Lambda}(t-t_0)} \mathbf{x}(t_0) + \mathbf{W}^\top \mathbf{m}(\mathbf{x}) + \boldsymbol{\Lambda} \mathbf{n}(\mathbf{x}). \quad (5)$$

The integral functions $\mathbf{m}(x)$ and $\mathbf{n}(x)$ are memory terms equivalent to momentum terms in neural networks which can be computed as follows

$$\begin{aligned}\dot{\mathbf{m}}(x) &= -\lambda \mathbf{m}(x) + \Phi(x, \mathbf{u}), \quad \mathbf{m}(x(t_0)) = \mathbf{0}_{r \times 1}, \\ \dot{\mathbf{n}}(x) &= -\Lambda \mathbf{n}(x) + x, \quad \mathbf{n}(x(t_0)) = \mathbf{0}_{n \times 1}.\end{aligned}\quad (6)$$

Remark 2: The parameterization model (3) can be used directly as a physics-informed structure for state estimation. However, the update rule (12) will depend on the time-derivative of the states vector which is not available in this approach. Well-designed filters can be used to estimate them from the raw measurements but this is time consuming and incorporates time-delays in the filtered signals.

In the following section, this issue is solved by using an estimated model of the nonlinear dynamics (3) which uses estimates of the states to build the basis functions $\Phi(\cdot)$ and the memory terms $\mathbf{m}(\cdot)$ and $\mathbf{n}(\cdot)$.

III. PHYSICS INFORMED RECURRENT NEURAL NETWORK (PIRNN) MODEL

The proposed physics-informed recurrent neural network (PIRNN) architecture is depicted in Fig. 1. An estimated model is constructed as a physics-informed model of the autonomous platform whose states are used to train the RNN [41]. Here, the difference between the output measurements $\mathbf{z}$ and the estimated model $\mathbf{w}$ are used to update the RNN weights. The RNN uses the two memory functions constructed from the autonomous platform structure and the estimated states to regularize and constraint the learning manifold of the RNN weights. This provides to the weights of a physical meaning that improves the performance of the RNN for state estimation and noise suppression.

A PIRNN network is constructed using a model estimate of (1) with the following equivalent structure

$$\begin{aligned}\dot{\mathbf{y}} &= \hat{\mathbf{f}}(\mathbf{y}, \mathbf{v}), \quad \mathbf{y}(t_0) = \mathbf{x}_0, \\ \mathbf{w} &= \mathbf{y},\end{aligned}\quad (7)$$

where $\mathbf{y} \in \mathbb{R}^n$ defines the state vector of (7), $\mathbf{v} \in \mathbb{R}^m$ is the control input vector with the same form as the control input $\mathbf{u}$ in (1), $\mathbf{w} \in \mathbb{R}^n$ denotes the output vector of (7), $\hat{\mathbf{f}}(\mathbf{y}, \mathbf{v}) \in \mathbb{R}^n$ is an estimate of the drift dynamics $\mathbf{f}$ which satisfies the following parameterization

$$\hat{\mathbf{f}}(\mathbf{y}, \mathbf{v}) = \widehat{\mathbf{W}}^\top \Phi(\mathbf{y}, \mathbf{v}), \quad (8)$$

where $\widehat{\mathbf{W}} \in \mathbb{R}^{r \times n}$ is the estimate of $\mathbf{W}$. Then, the estimated model (7) is written as a RNN architecture of the form

$$\dot{\mathbf{y}} = \widehat{\mathbf{W}}^\top \Phi(\mathbf{y}, \mathbf{v}). \quad (9)$$

The solution of the RNN model (9) can be written in terms of the memory terms as

$$\mathbf{y}(t) = e^{-\Lambda(t-t_0)} \mathbf{y}(t_0) + \widehat{\mathbf{W}}^\top \mathbf{m}(\mathbf{y}) + \Lambda \mathbf{n}(\mathbf{y}). \quad (10)$$

In contrast to (6), the RNN solution (10) is based on the estimated states of (7) that do not contain noise. Thus, we only require to design a mechanism that connects both the

real nonlinear trajectories with the estimated ones of the RNN to estimate the weights of the linear parameterization and infer the system states with noise attenuation. To this end, the output error $\mathbf{e} \in \mathbb{R}^n$ between the system outputs $\mathbf{z}$ and the output of the RNN model $\mathbf{w}$ is used as main signal to bridge both models. The output error $\mathbf{e}$ is given by

$$\begin{aligned}\mathbf{e} &= \mathbf{z} - \mathbf{w} = \mathbf{x} + \boldsymbol{\omega} - \mathbf{y} \\ &= \mathbf{W}^\top \mathbf{m}(\mathbf{x}) - \widehat{\mathbf{W}}^\top \mathbf{m}(\mathbf{y}) + \Lambda[\mathbf{n}(\mathbf{x}) - \mathbf{n}(\mathbf{y})] + \boldsymbol{\omega} \\ &= -\widetilde{\mathbf{W}}^\top \mathbf{m}(\mathbf{y}) + \boldsymbol{\varepsilon},\end{aligned}\quad (11)$$

where the weights error is given by $\widetilde{\mathbf{W}} = \widehat{\mathbf{W}} - \mathbf{W} \in \mathbb{R}^{r \times n}$ and $\boldsymbol{\varepsilon} = \mathbf{W}^\top [\mathbf{m}(\mathbf{x}) - \mathbf{m}(\mathbf{y})] + \Lambda[\mathbf{n}(\mathbf{x}) - \mathbf{n}(\mathbf{y})] + \boldsymbol{\omega}$ is an approximation error with upper bound $\bar{\varepsilon} > 0$, i.e., $\|\boldsymbol{\varepsilon}\| \leq \bar{\varepsilon}$.

Here, the RNN weights are estimated with the following update rule based on a modified Levenberg–Marquardt algorithm

$$\dot{\widetilde{\mathbf{W}}} = \dot{\widehat{\mathbf{W}}} = \Gamma \frac{\mathbf{m}(\mathbf{y})}{\bar{m}^2} \mathbf{e}^\top, \quad (12)$$

where $\Gamma \succ 0 \in \mathbb{R}^{r \times r}$ is a diagonal gain matrix and $\bar{m} = 1 + \mathbf{m}^\top(\mathbf{y})\mathbf{m}(\mathbf{y})$. Weights convergence is ensured if the inputs in the basis functions $\phi_i(\mathbf{y}, \mathbf{v})$ fulfils a persistent of excitation (PE) condition (also known as uniform complete observability (UCO) [42]). This is usually achieved by adding a probing noise into the control input signal to excite the network modes such that the weights learning is improved.

The following theorem is used to establish the uniform ultimate boundedness (UUB) of the trajectories of the weights estimates (12) as long as the signals within $\mathbf{m}(\mathbf{y})$ fulfil a PE condition.

Theorem 1: Consider the output error function (11) between the nonlinear autonomous platform model (5) and the model estimate (10). Let the weights being updated by the update rule (12) and assume that $\mathbf{m}(\mathbf{y})$ is PE. Then the trajectories are UUB and $\widetilde{\mathbf{W}}$ remain bounded.

Proof: Consider the following Lyapunov function

$$V = \frac{1}{2} \text{tr}\{\widetilde{\mathbf{W}}^\top \Gamma^{-1} \widetilde{\mathbf{W}}\}.$$

The time-derivative of the Lyapunov function under the output error trajectories (11) and the update rule (12) is

$$\begin{aligned}\dot{V} &= \text{tr}\{\widetilde{\mathbf{W}}^\top \Gamma^{-1} \dot{\widetilde{\mathbf{W}}}\}, \\ &= \text{tr}\{\widetilde{\mathbf{W}}^\top \frac{\mathbf{m}(\mathbf{y})}{\bar{m}^2} \mathbf{e}^\top\} \\ &= -\text{tr}\{\widetilde{\mathbf{W}}^\top \frac{\mathbf{m}(\mathbf{y})\mathbf{m}^\top(\mathbf{y})}{\bar{m}^2} \widetilde{\mathbf{W}}\} + \text{tr}\{\widetilde{\mathbf{W}}^\top \frac{\mathbf{m}(\mathbf{y})}{\bar{m}^2} \boldsymbol{\varepsilon}^\top\} \\ &\leq -\left\|\widetilde{\mathbf{W}}^\top \frac{\mathbf{m}(\mathbf{y})}{\bar{m}}\right\|^2 + \left\|\widetilde{\mathbf{W}}^\top \frac{\mathbf{m}(\mathbf{y})}{\bar{m}}\right\| \left\|\frac{\boldsymbol{\varepsilon}^\top}{\bar{m}}\right\| \\ &= -\left\|\widetilde{\mathbf{W}}^\top \frac{\mathbf{m}(\mathbf{y})}{\bar{m}}\right\| \left(\left\|\widetilde{\mathbf{W}}^\top \frac{\mathbf{m}(\mathbf{y})}{\bar{m}}\right\| - \left\|\frac{\boldsymbol{\varepsilon}^\top}{\bar{m}}\right\|\right)\end{aligned}$$

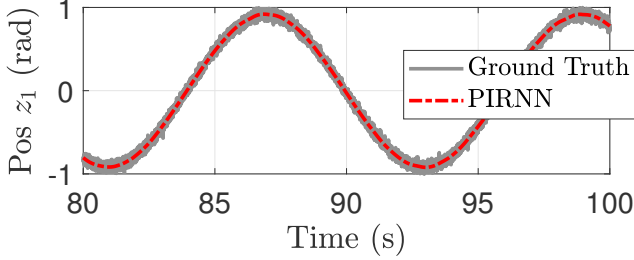
Therefore, $\dot{V} \leq 0$ if

$$\left\|\widetilde{\mathbf{W}}^\top \frac{\mathbf{m}(\mathbf{y})}{\bar{m}}\right\| > \bar{\varepsilon} \equiv \mu > \left\|\frac{\boldsymbol{\varepsilon}^\top}{\bar{m}}\right\|,$$

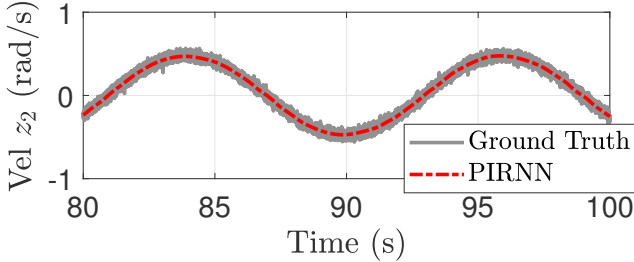
The vector of basis functions based on an estimated model is $\Phi(\mathbf{y}) = [y_2, \sin(y_1), \nu]^\top \in \mathbb{R}^3$, where ν has the following structure

$$\nu = 40(x_1^d - y_1) + 30(x_2^d - y_2) + \delta.$$

The gains of the memory vectors and weights are set to $\lambda = 2000$ and $\Gamma = 5000\mathbf{I}_3$, respectively, where $\mathbf{I}_3 \in \mathbb{R}^{3 \times 3}$ denotes a 3×3 identity matrix. The state inference results are given in Fig. 2. The results demonstrate that the PIRNN network is capable to accurately infer the trajectory with noise suppression. One of the major benefits is the incorporation of a physics-informed structure that regularizes the weights. This is shown in Fig. 3(a) where the weights of the network converges and remain bounded. Moreover, these weights satisfy the physics model avoiding large learning weights. Here, the weights are not equivalent to the physical model since the proposed PIRNN finds a physical linear combination of the weights based on the basis function in $\Phi(\mathbf{y}, \nu)$.



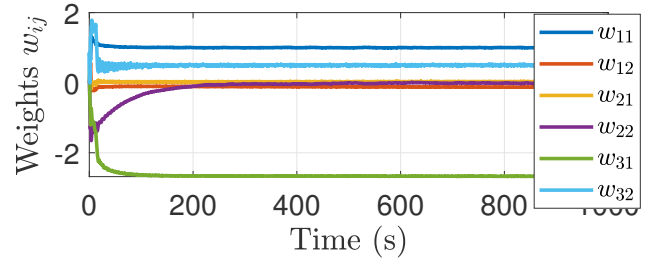
(a) Output z_1 vs w_1



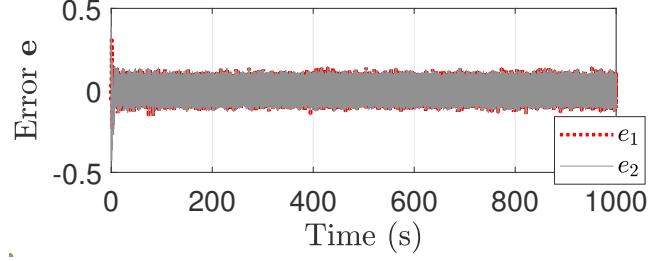
(b) Output z_2 vs w_2

Fig. 2. State Inference results

One important property of the proposed PIRNN is that maintains similar properties to Kalman filter algorithms. The output error e is equivalent to the innovation in Kalman filter, that is, only the noise generated by the sensors is obtained (see Fig. 3(b)), whilst the real signal without noise (or with noise attenuation) is recovered as state estimate.



(a) Physical weights learning



(b) Output error e

Fig. 3. Weights convergence and output error results

B. Drone Attitude Dynamics

Consider the drone attitude dynamics given by

$$\begin{aligned} \dot{x}_1 &= x_4, & \dot{x}_2 &= x_5, & \dot{x}_3 &= x_6, \\ \dot{x}_4 &= \frac{1}{I_x} \tau_\phi - \frac{I_y - I_z}{I_x} x_5 x_6, \\ \dot{x}_5 &= \frac{1}{I_y} \tau_\theta - \frac{I_z - I_x}{I_y} x_4 x_6, \\ \dot{x}_6 &= \frac{1}{I_z} \tau_\psi - \frac{I_x - I_y}{I_z} x_4 x_5, \end{aligned}$$

where $\mathbf{x}_p = [x_1, x_2, x_3]^\top \in \mathbb{R}^3$ and $\mathbf{x}_v = [x_4, x_5, x_6]^\top \in \mathbb{R}^3$ denote the roll, pitch, yaw angles vector and their corresponding angular velocities vector, I_x, I_y, I_z are the moments of inertia in the principal axes, and $\boldsymbol{\tau} = [\tau_\phi, \tau_\theta, \tau_\psi]^\top \in \mathbb{R}^3$ stands to the input torque. The following values are considered in this experiment $I_x = I_y = 4.856 \times 10^{-3}$ and $I_z = 8.801 \times 10^{-3}$ kgm², which are nominal values in several real-world drones.

A simple PD controller is used with the following structure

$$\boldsymbol{\tau} = 4(\mathbf{x}_p^d - \mathbf{x}_p) + 3(\mathbf{x}_v^d - \mathbf{x}_v) + \delta,$$

where $\mathbf{x}_p^d = [x_1^d, x_2^d, x_3^d]^\top \in \mathbb{R}^3$ and $\mathbf{x}_v^d = [x_4^d, x_5^d, x_6^d]^\top \in \mathbb{R}^3$ denote the desired positions and velocities for the attitude dynamics, and δ is a PE signal given by an exponential decay sinusoidal function with different frequencies.

The control input of the reference model has the same structure as $\boldsymbol{\tau}$, i.e.,

$$\mathbf{v} = 4(\mathbf{x}_p^d - \mathbf{y}_p) + 3(\mathbf{x}_v^d - \mathbf{y}_v) + \delta.$$

The physics structure of the attitude model written in terms of an estimated model with state $\mathbf{y} = [y_1, y_2, y_3, y_4, y_5, y_6] \in \mathbb{R}^6$

$\mathbb{R}^6$ and control $\mathbf{v} = [v_1, v_2, v_3]^\top \in \mathbb{R}^3$ is

$$\Phi(\mathbf{y}, \mathbf{v}) = [y_4, y_5, y_6, v_1, y_5 y_6, v_2, y_4 y_6, v_3, y_4 y_5]^\top \in \mathbb{R}^9.$$

The hyperparameter of the memory terms is set to $\lambda = 200$. The gain of the weights update rule is set to $\Gamma = 200\mathbf{I}_9$.

The estimation results are shown in Fig. 4. Similarly to the 1-DOF robot, the proposed PIRNN is capable to infer the states of the model accurately with noise suppression. Notice that, estimation models based in linearization techniques such as EKF will loss the information associated to the Coriolis forces which play a pivotal contribution in the observed behaviour of the drone. In contrast, the PIRNN maintains this physical property in the model structure such that the evolution of the states are well represented.

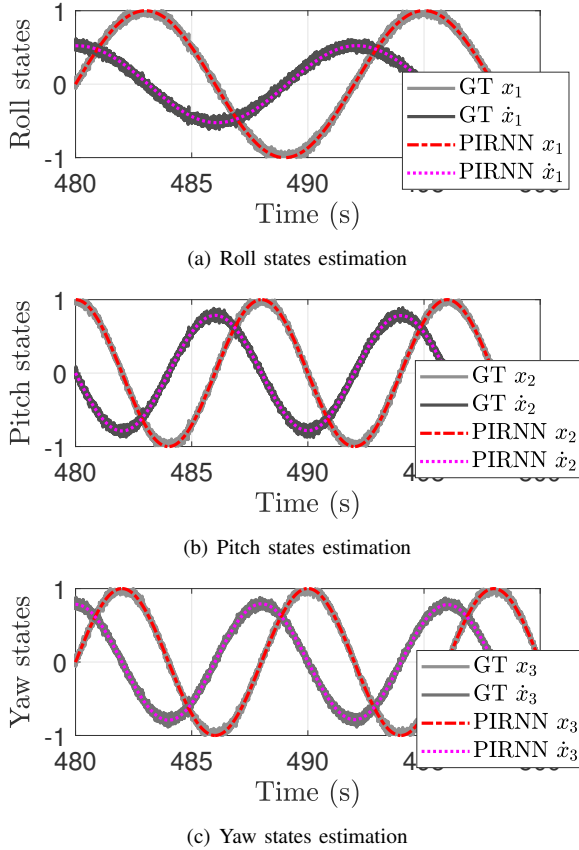


Fig. 4. State Estimation results. GT stands to Ground Truth.

The physics structure helps to regularize the weights manifold and ensure convergence under the fulfilment of a PE condition. This fact is observed in Fig. 5, where the weights of the RNN remain bounded and satisfy the physical model structure imposed by Φ . One important property of the proposed PIRNN, is that it does not require measurements of the derivative of the state vector and knowledge of the parameters of the platform. This is an advantage to previous PINNs that assume knowledge of the exact platform model. Additionally, the physical structure is not embedded at the loss function as regularization term. In contrast, the physics structure is

given as basis functions that allow the construction of memory vectors (equivalent to momentum terms) that enhance the prediction capabilities of the network.

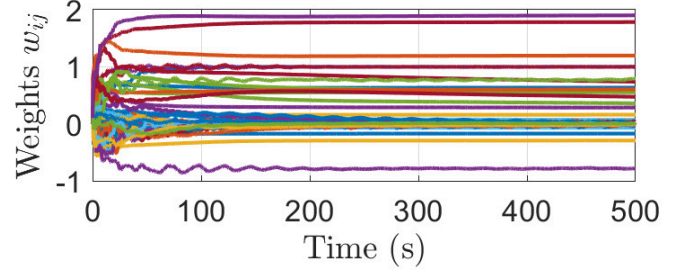


Fig. 5. Physical weights convergence

Similarly to the 1-DOF robot, the output error between the estimated model and the real measurements of the drone attitude states is equivalent to the noise of the real-input measurements (see Fig. 6).

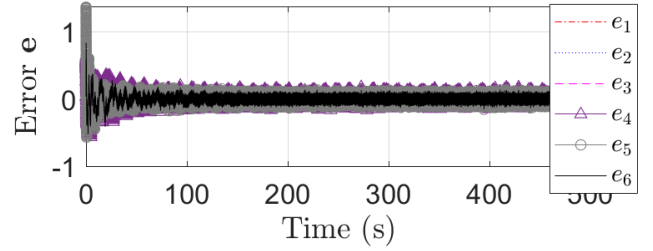


Fig. 6. Output error results

V. CONCLUSIONS

This paper reports a physics-informed recurrent neural network for state estimation of autonomous platforms. The proposed approach incorporates a physical structure to regularize the weights leaning manifold of the network and enhance its stability. To this end, a parameterization of the nonlinear differential equation solution is constructed to naturally obtain two memory vectors that helps to stabilize and enhance the learning capabilities of the network. Lyapunov stability theory is used to verify the boundedness and stability of the network under the fulfilment of a persistent of excitation condition. Simulation experiments are carried out to demonstrate the benefits of the approach.

Future work will consider the extension of the approach to partial states measurements with both linear and nonlinear models.

ACKNOWLEDGEMENT

This work was supported by the Engineering and Physical Sciences Research Council under Grant EP/V026763/1.

REFERENCES

- [1] R. Rezaie and X. R. Li, "Destination-directed trajectory modeling, filtering, and prediction using conditionally markov sequences," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 2, pp. 820–833, 2020.
- [2] X. Li, G. Fan, K. Pan, G. Wei, C. Zhu, G. Rizzoni, and M. Canova, "A physics-based fractional order model and state of energy estimation for lithium ion batteries. part i: Model development and observability analysis," *Journal of Power Sources*, vol. 367, pp. 187–201, 2017.
- [3] S. Zeng and E. Fernandez, "Adaptive controller design and disturbance attenuation for sequentially interconnected SISO linear systems under noisy output measurements with partly measured disturbances," *2008 American Control Conference*, 2008.
- [4] K. KN, D. Ignatyev, A. Tsourdos, and A. Perrusquia, "Advancing fault diagnosis in aircraft landing gear: An innovative two-tier machine learning approach with intelligent sensor data management," in *AIAA SCITECH 2024 Forum*, 2024, p. 0759.
- [5] W. Yu and A. Perrusquia, "Simplified stable admittance control using end-effector orientations," *International Journal of Social Robotics*, vol. 12, no. 5, pp. 1061–1073, 2020.
- [6] J. Mendoza, A. Perrusquia, and J. A. Flores-Campos, "Mechanical advantage assurance control of quick-return mechanisms in task space," in *2022 19th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*. IEEE, 2022, pp. 1–6.
- [7] B. Kiumarsi, K. G. Vamvoudakis, H. Modares, and F. L. Lewis, "Optimal and autonomous control using reinforcement learning: a survey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 6, pp. 2042–2062, 2018.
- [8] W. Gilpin, "Chaos as an interpretable benchmark for forecasting and data-driven modelling," *arXiv preprint arXiv:2110.05266*, 2021.
- [9] G. Singh, A. Perrusquia, and W. Guo, "A two-stages unsupervised/supervised statistical learning approach for drone behaviour prediction," in *2023 9th International Conference on Control, Decision and Information Technologies (CoDIT)*. IEEE, 2023, pp. 1–6.
- [10] J. A. Flores-Campos and A. Perrusquia, "Robust control of linear systems: A min-max reinforcement learning formulation," in *2023 20th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*. IEEE, 2023, pp. 1–6.
- [11] Z. Peng, Y. Jiang, L. Liu, and Y. Shi, "Path-guided model-free flocking control of unmanned surface vehicles based on concurrent learning extended state observers," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2023.
- [12] A. Perrusquia, "Solution of the linear quadratic regulator problem of black box linear systems using reinforcement learning," *Information Sciences*, vol. 595, pp. 364–377, 2022.
- [13] R. Corteso, J. Park, and O. Khatib, "Real-time adaptive control for haptic telemanipulation with kalman active observers," *IEEE Transactions on Robotics*, vol. 22, no. 5, pp. 987–999, 2006.
- [14] C. Wang, H. Han, J. Wang, H. Yu, and D. Yang, "A robust extended kalman filter applied to ultrawideband positioning," *Mathematical Problems in Engineering*, vol. 2020, 2020.
- [15] T. Raïssi and M. Aoun, "On robust pseudo state estimation of fractional order systems," in *International Symposium on Positive Systems*. Springer, 2016, pp. 97–111.
- [16] B. Fraser, A. Perrusquia, D. Panagiotakopoulos, and W. Guo, "A deep mixture of experts network for drone trajectory intent classification and prediction using non-cooperative radar data," in *2023 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2023, pp. 1–6.
- [17] A. Azami, S. V. Naghavi, R. Dadkhah Tehrani, M. H. Khooban, and F. Shabaninia, "State estimation strategy for fractional order systems with noises and multiple time delayed measurements," *IET Science, Measurement & Technology*, vol. 11, no. 1, pp. 9–17, 2017.
- [18] L. B. White and F. Carravetta, "State-space realizations and optimal smoothing for gaussian generalized reciprocal processes," *IEEE Transactions on Automatic Control*, vol. 65, no. 1, pp. 389–396, 2019.
- [19] R. Rezaie and X. R. Li, "Gaussian conditionally markov sequences: Modeling and characterization," *Automatica*, vol. 131, p. 109780, 2021.
- [20] P. Geragersian, I. Petrunin, W. Guo, and R. Grech, "An ins/gnss fusion architecture in gnss denied environment using gated recurrent unit," in *AIAA SCITECH 2022 Forum*, 2022, p. 1759.
- [21] R. G. Brown and P. Y. Hwang, "Introduction to random signals and applied kalman filtering: with matlab exercises and solutions," *Introduction to random signals and applied Kalman filtering: with MATLAB exercises and solutions*, 1997.
- [22] G. Revach, N. Shlezinger, X. Ni, A. L. Escoriza, R. J. van Sloun, and Y. C. Eldar, "Kalmannet: Neural network aided kalman filtering for partially known dynamics," *arXiv preprint arXiv:2107.10043*, 2021.
- [23] L. Xu and R. Niu, "Ekfnet: Learning system noise statistics from measurement data," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 4560–4564.
- [24] X. Zheng, M. Zaheer, A. Ahmed, Y. Wang, E. P. Xing, and A. J. Smola, "State space lstm models with particle mcmc inference," *arXiv preprint arXiv:1711.11179*, 2017.
- [25] P. Becker, H. Pandya, G. Gebhardt, C. Zhao, C. J. Taylor, and G. Neumann, "Recurrent kalman networks: Factorized inference in high-dimensional deep feature spaces," in *International Conference on Machine Learning*. PMLR, 2019, pp. 544–552.
- [26] W. Yu and A. Perrusquia, *Human-Robot Interaction Control Using Reinforcement Learning*. Wiley-IEEE Press, 2022.
- [27] C. M. Legaard, T. Schranz, G. Schweiger, J. Drgoňa, B. Falay, C. Gomes, A. Iosifidis, M. Abkar, and P. G. Larsen, "Constructing neural network-based models for simulating dynamical systems," *arXiv preprint arXiv:2111.01495*, 2021.
- [28] A. Perrusquia and W. Guo, "A closed-loop output error approach for physics-informed trajectory inference using online data," *IEEE Transactions on Cybernetics*, vol. 53, pp. 1379–1391, 2023.
- [29] P. J. Baddoo, B. Herrmann, B. J. McKeon, J. Nathan Kutz, and S. L. Brunton, "Physics-informed dynamic mode decomposition," *Proceedings of the Royal Society A*, vol. 479, no. 2271, p. 20220576, 2023.
- [30] A. Perrusquia and W. Guo, "Physics informed trajectory inference of a class of nonlinear systems using a closed-loop output error technique," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2023.
- [31] B. Fraser, A. Perrusquia, D. Panagiotakopoulos, and W. Guo, "Hybrid deep neural networks for drone high level intent classification using non-cooperative radar data," in *2023 3rd International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*. IEEE, 2023, pp. 1–6.
- [32] O. Nelles, "Nonlinear dynamic system identification," in *Nonlinear System Identification*. Springer, 2020, pp. 831–891.
- [33] A. Perrusquia and W. Guo, "Closed-loop output error approaches for drone's physics informed trajectory inference," *IEEE Transactions on Automatic Control*, 2023.
- [34] D. Li, B. Zhang, P. Li, E. Q. Wu, R. Law, X. Xu, A. Song, and L.-M. Zhu, "Parameter estimation and anti-sideslip line-of-sight method-based adaptive path-following controller for a multi-joint snake robot," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2023.
- [35] A. Perrusquia, R. Garrido, and W. Yu, "Stable robot manipulator parameter identification: A closed-loop input error approach," *Automatica*, vol. 141, p. 110294, 2022.
- [36] T. Yamamoto, M. Bernhardt, A. Peer, M. Buss, and A. M. Okamura, "Techniques for Environment parameter Estimation during telemanipulation," *Proceedings of the 2nd Biennial IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechanics*, 2008.
- [37] I. D. Landau and A. Karimi, "An output error recursive algorithm for unbiased identification in closed loop," *Automatica*, vol. 33, no. 5, 1997.
- [38] K. G. Vamvoudakis, "Q-learning for continuous-time linear systems: A model-free infinite horizon optimal control approach," *Systems & Control Letters*, pp. 14–20, 2017.
- [39] S. A. A. Rizvi and Z. Lin, "Output feedback q-learning control for the discrete-time linear quadratic regulator problem," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 5, 2018.
- [40] H. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2002.
- [41] A. Perrusquia, W. Guo, B. Fraser, and Z. Wei, "Uncovering drone intentions using control physics informed machine learning," *Communications Engineering*, vol. 3, no. 1, p. 36, 2024.
- [42] K. Vamvoudakis and F. L. Lewis, "On-line actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, pp. 878–888, 2010.
- [43] A. Perrusquia, "Robust state/output feedback linearization of direct drive robot manipulators: A controllability and observability analysis," *European Journal of Control*, 2022.
- [44] J. A. Flores-Campos, A. Perrusquia, L. H. Hernández-Gómez, N. González, and A. Armenta-Molina, "Constant speed control of slider-crank mechanisms: A joint-task space hybrid control approach," *IEEE Access*, vol. 9, pp. 65 676–65 687, 2021.



DSPACE

<https://dspace.org/>

A novel physics-informed recurrent neural network approach for state estimation of autonomous platforms

Perrusquía, Adolfo; Guo, Weisi

2024-06-30

Institute of Electrical and Electronics Engineers (IEEE)

<https://doi.org/10.1109/ijcnn60899.2024.10650856>

Perrusquía A, Guo W. (2024) A novel physics-informed recurrent neural network approach for state estimation of autonomous platforms. In: 2024 International Joint Conference on Neural Networks (IJCNN), 30 June - 5 July 2024, Yokohama, Japan