NEURAL NETWORK FITTING FOR INPUT-OUTPUT MANIFOLDS OF ONLINE OBSERVERS WITH ERROR LIMITATION

MARCONI O. DE ALMEIDA* SAMARONNE N. DO CARMO* FLÁVIO A. DE CASTRO* RAFAEL F.R. CAMPOS* JOSÉ M. ARAÚJO* CARLOS E.T. DOREA†

*Instituto Federal da Bahia, Departamento de Automação e Sistemas
Rua Emídio dos Santos, S/N, Barbalho
Salvador, BA, 40301015, Brazil

†Universidade Federal do Rio Grande do Norte, Centro de Tecnologia
Departamento de Engenharia de Computação e Automação
Lagoa Nova, Natal, RN, 59078900, Brazil

Email: araujo@ieee.org

Abstract— This paper aims to present the use of Neural Networks (NN) to approximate the input-output map manifold for online output injection laws in the context of observers with error limitation. The set invariance approach is applied for computation of as small as possible conditioned invariant sets that confine the estimation error of a full-order observer. Online output injection laws are attractive due to its ability to cope with the peak phenomenon, but the runtime of its computation may turn its use unfeasible in systems with fast dynamics. The input-output map of such injection law is then approximated via NN training. Structures with NNs present processing times as fast as the adequate technology is employed for their implementation, for instance, FPGAs. Two examples, one of which the classical well-known magnetic levitation ball system, are presented to illustrate the merit of the approach.

Keywords— Set invariance, Observers, Neural Networks, Function Approximation.

1 Introduction

The design of observers with error limitation is a relevant issue in control systems theory and practice. In linear observer design, the trade-off between fast convergence and inrush in initial performance arises from the peak phenomenon verified in high gain observers (Gauthier and Kupka (2001)). In the last two decades, some important contributions on the design of observers with error limitation, including systems subject to measurement noise and disturbances, have been reported. Two of them are the so-called set-valued and set-invariant state estimators. The first (Shamma and Tu (1999);Lin et al. (2003);Blanchini and Minami (2007)) consists in computing online (at each step) the set of admissible states with respect to the measurement, taking into account noise and disturbance. An optimal state estimate is then picked from such a set according to a given criterion. The main disadvantage in this case is the time needed for online computation that may increase quickly with the dimension of the problem. Some contributions have succeeded in mitigating this problem as in Haimovich et al. (2004).

The set invariance approach, on the other hand, is interesting because a substantial part of the computation, if not all of it, can be performed offline. In some cases, namely with a single measurement output, analytical expressions for the output injection law can be constructed by means of piecewise affine functions. Important contributions on this subject can be found in Araujo et al. (2011);Dórea and Pimenta (2005a);Dórea and Pimenta (2005b). The peak phenomenon is still present as seen in Araujo et al. (2011), as a consequence of high gain structures. Nevertheless, yet in Araujo et al. (2011), an online output injection law shows a good response, overcoming the peak phenomenon.

Artificial Neural Networks or simple Neural Networks (NNs) form a class in Artificial Intelligence techniques with several successful applications, among them control and modeling and estimation of dynamical systems (Narendra and Parthasarathy (1990),Yamada (2008),Mini and Padma Suresh (2014),Alessandri et al. (2011),Alessandri et al. (2008)). func-
tion approximation (Andras (2014), Yang et al. (2013)); pattern recognition and time series prediction (Azad et al. (2014), Bishop (1995), Haykin (1999), Ouyang and Yin (2014), Wu et al. (2014)). The present state of NN firmware implementations is such that the execution times for relatively large NNs are of microseconds order, and it suggests applications even in the so-called real-time systems (Krips et al. (2002); Omondi and Rajapakse (2006)).

This note focuses in the application of NNs in the approximation of the map manifold of online output injection laws computed from conditioned invariant sets for error limitation of observers in discrete time linear systems. Since the simulation phase of any implementation is an offline task, the set of training points can be obtained this way based on the computed online output injection laws. The advantage is that the NN training, subject to unknown but bounded disturbances and measurement noise in state-space form:

$$x(k + 1) = Ax(k) + B_1 d(k) \quad y(k) = C x(k) + \eta(k),$$

with $$x \in \mathbb{R}^n$$, $$d \in \mathbb{R}^q$$, $$y, \eta \in \mathbb{R}^l$$, being respectively the system state, the exogenous disturbance, the measurement and the measurement noise. State estimation can be performed using a full order observer given by:

$$\hat{x}(k + 1) = A \hat{x}(k) - \Gamma(z(k), k)$$
$$\hat{y}(k) = C \hat{x}(k),$$

Figure 1: Signal-flow graph for a single perceptron.

$$e(k + 1) = A \eta e(k) + B_1 d(k) + \Gamma(z(k), k)$$
$$z(k) = C \eta e(k) + \eta(k)$$

Let the disturbance $$d$$ belong to a compact 0-symmetrical polyhedron $$\mathcal{D} \subset \mathbb{R}^q$$, and noise measurements to the set $$\mathcal{N} = \{ \eta : |\eta| \leq \bar{\eta} \}$$. The estimation errors which are consistent with each $$z \in \mathcal{Z}(\mathcal{\Omega})$$ compose the set:

$$\mathcal{E}(z) = \{ e : Ce = z - \eta, \eta \in \mathcal{N} \}.$$

A polyhedron $$\mathcal{\Omega} \subset \mathbb{R}^n$$, is said to be conditioned-invariant $$\lambda$$-contractive with respect to system (3) if:

$$\forall z \in \mathcal{Z}(\mathcal{\Omega}), \exists \Gamma : Ae + B_1 d + \Gamma \in \lambda \mathcal{\Omega},$$

$$\forall d \in \mathcal{D}, \forall e \in \mathcal{E}(z) \cap \mathcal{\Omega}.$$

Actually, a small ultimate boundedness set confines $$e(k)$$ for some $$k$$ in the presence of disturbance and measurement noise.

In a few cases for single output systems, a linear or piecewise linear output injection law $$\Gamma$$ can be constructed, as seen in Dória and Pimenta (2005b). The main advantage is that such laws are offline in nature. But the peak phenomenon may still be present in this case, as shown in Araújo et al. (2011). For multi-output systems, an offline injection law may even exist, but a systematic way to determine its structure is unknown to date for general systems (Dorea (2006)).

2.2 Neural Networks

Neural Networks are sets of basic units called neurons, whose diagram representation is shown in fig. 1. The development of these systems have experimented a fast growth since they were introduced in (Farley and Clark, 1954), and belongs to the class of Artificial Intelligence (AI)
techniques. Their main feature is the capability of learning input-output pattern from given data and has been successfully used in pattern recognition, time-series prediction and function approximation. In the control systems field, some important applications include modeling of dynamical systems and neural estimators and controllers (Narendra and Parthasarathy (1990)). In fig. 1, a flow-graph for a single neuron or perceptron is depicted, and fig. 2 shows the classical architecture of multilayer perceptrons (MLP). The transfer function of a single perceptron is given by:

\[ y = \varphi \left( \sum_{i=1}^{m} w_i x_i + b \right), \tag{5} \]

where \( w_i \) is the \( i \)-th input weight, \( b \) is the constant bias and \( \varphi(\cdot) \) is the so-called activation function - generally a sigmoid hyperbolic tangent or logistic function.

### 3 The Proposed Approach

In the general case, for a given \( \lambda \)-contractive polyhedron \( \Omega = \{ e : Ge \leq \rho \} \), the online computation at each step can be performed as (Dorea (2006)):

\[ \min_{\Gamma(z(k), k), \varepsilon} \varepsilon \tag{6} \]

s.t. \( \phi(\Omega, z(k)) + GT(z(k), k) \leq \varepsilon \rho - \delta \),

where \( \delta \) is a vector representing the worst case of disturbance action, given by:

\[ \delta_i = \max \{G_i B_i d\} \tag{7} \]

s.t. \( Dd \leq w \).

The vector \( \phi \) represents the worst case of all admissible errors associated to a given output \( z \). It must be computed, row by row, at each step by:

\[ \phi_i(\Omega, z) = \max_{e, \eta} G_i A e \tag{8} \]

s.t. \( Ge \leq \rho, N(Ce - z) \leq \nu \).

The main disadvantage of this online scheme is noted in systems with fast dynamics. A number of \( g \) linear programmes (LPs) with \( n \) variables and \( g \) constraints for \( \phi \), plus the linear program (6) must be solved at each step. It is clear that such computation can become quite expensive. However, the peak phenomenon can be mitigated if one employs this approach Araujo et al. (2011).

Since the input-output manifold of online computed injection laws can be obtained by simulation from the solutions of LPs (6), (7) and (8), the proposed approach consists in training multilayer NNs perceptrons to emulate the manifold \( \Gamma(z(k)) \) with high accuracy. The natural choice for NNs is the feedforward type with backpropagation learning algorithm, and Levenberg-Marquardt strategy is suggested to give good convergence for moderate size networks with random initial weights (Hagan and Menhaj (1994)). The training data is obtained from the system simulation error trajectories with initial error being random points of the initial confidence polyhedron. The minimal conditioned invariant polyhedron is computed by means of the algorithms presented in Araujo et al. (2011).

### 4 Numerical Examples

#### 4.1 A Second Order System

This example is borrowed from (Le et al., 2011). The system model is given by:

\[
\begin{align*}
    x(k+1) &= \begin{bmatrix} 0 & -\frac{1}{2} \\ 1 & 1 \end{bmatrix} x(k) + \begin{bmatrix} -\frac{3}{2} \\ \frac{1}{2} \end{bmatrix} d(k) \tag{9} \\
    y(k) &= \begin{bmatrix} -2 & 1 \end{bmatrix} + \eta(k),
\end{align*}
\]

in which \(|d| \leq 1 \) and \(|\eta| \leq 0.2 \). The confidence polyhedron of initial error is the box:

\[ \Omega = \{ e_0 : |e_0| \leq \begin{bmatrix} 3 & 3 \end{bmatrix}^T \} \]

A minimal conditioned invariant polyhedron with contraction rate \( \lambda_o = 0.8 \) was then computed. This set is seen in fig. 3 together with the box \( \Omega \). A MLP with 9 neurons at hidden layer is then trained from error trajectories generated by the simulated online output injection law starting from random initial conditions. A number of five training starting from random initial weights were tested, with almost the same performance index being obtained for the trained NN. In fig. 4, a simulation of the neural and the online observer is shown, starting from an initial condition that does not belong to the training set. A good fitting was obtained for the two error trajectories. The input-output map of the neural observer \( \Gamma(z) = [\gamma_1(z) \gamma_2(z)]^T \times z \) is sketched in fig. 5. A piecewise affine-like behavior is evident in this non-linear manifold. This fact had already
been observed in previous works, as in (Araujo et al., 2012).

4.2 Magnetic Ball Levitation System

In this classical example, taken from Golnaraghi and Kuo (2009) and sketched in fig. 6, a discretization with sampling time $T_s = 2ms$ was applied, and the linearized, discrete-time model is given by eq. (10).

$$
\begin{align*}
\dot{x}(k+1) & = 
\begin{bmatrix}
1.0001 & 0.0020 & 0 \\
0.1288 & 1.0001 & -0.0290 \\
0 & 0 & 0.8178
\end{bmatrix}
\begin{bmatrix}
x(k) \\
u(k)
\end{bmatrix} \\
y(k) & = 
\begin{bmatrix}
0 \\
-0.0030 \\
0.1813
\end{bmatrix}
\begin{bmatrix}
x(k) \\
u(k)
\end{bmatrix} + \eta(k).
\end{align*}
$$

The set of the confident constraints on the initial error is composed by $|e_1| \leq 0.005m$, $|e_2| \leq 0.1m/s$ and $|e_3| \leq 0.1A$, and the noise in velocity measurement is bounded as $|\eta| \leq 0.01m/s$. By using the algorithms presented in Araujo et al. (2011), an as small as possible, 230 faces conditioned invariant polyhedron was computed; this big number of faces indicates that the computation time of the online injection law must be appreciable. This set is showed at fig. 7. Using a generated training set composed by points of the trajectories for the error starting from some edge points in the confidence initial set, the online injection results were used to train a NN for function approximation with one hidden layer with 9 neurons and inverse hyperbolic tangent activation function. The good fitting between the neural and the online manifold is seen in fig. 8. In figs. 9 and 10, two noise free state trajectories are depicted, respectively, in the error space and in time domain, one using the online injection law computed for the system and other using the trained NN, and the absence of peak is notable in the later. Also, the average time for computation of a 100 steps simulation using an Intel(R) Core(TM)i3 - 2310M CPU@2.10GHz is showed at tab. 1. It must be noted that the sim function from Neural Network Toolbox was used, and still a better runtime was observed for...
Table 1: Mean runtime performance for a 100 steps simulation of the online and neural observers.

<table>
<thead>
<tr>
<th></th>
<th>Online</th>
<th>Neural</th>
</tr>
</thead>
<tbody>
<tr>
<td>135ms/step</td>
<td>7.5ms/step</td>
<td></td>
</tr>
</tbody>
</table>

Figure 7: Conditioned invariant polyhedron for the observer in magnetic ball system.

Figure 8: Fitting of the online observer. The initial condition does not belong to the initial conditions in the training set.

5 Concluding Remarks

A novel approach which combines the best of both worlds - the online and the offline injection law - was proposed in this paper, by training NNs with the goal of approximating the input-output manifold of an online output injection law, in the context of observers with error limitation. The piecewise affine like behavior of this laws could be observed in the input-output map captured by NNs. Two simulation examples were carried out to show the merit of the proposed approach with respect to runtime and peak phenomenon mitigation, one of which a classical control system benchmark found in the literature.

Future works include the use of NNs to emulate the input-output map for the problem of static or dynamic output feedback control of constrained systems.

Acknowledgment

The authors would like to thank their Institutions and FAPESB for financial support.

References


Andras, P. (2014). Function approximation using combined unsupervised and supervised learn-
Bishop, C. (1995). *Neural Networks for Pattern Recognition*, Neural Networks for Pattern Recognition, Oxford University Press, USA.


