ARTICLE IN PRESS

ISA Transactions ∎ (∎∎∎∎) ∎∎∎–∎∎∎

Contents lists available at ScienceDirect

ISA Transactions

journal homepage: www.elsevier.com/locate/isatrans

Research article

Model predictive control for systems with fast dynamics using inverse neural models

Marios Stogiannos [a,b], Alex Alexandridis [a,*], Haralambos Sarimveis [b]

[a] Department of Electronic Engineering, Technological Educational Institute of Athens, Agiou Spiridonos, Aigaleo 12243, Greece
[b] School of Chemical Engineering, National Technical University of Athens, Iroon Polytechneiou 9, Zografos 15780, Greece

ARTICLE INFO

Article history:
Received 6 April 2017
Received in revised form
19 September 2017
Accepted 22 September 2017

Keywords:
Applicability domain
Inverse models
Inverted pendulum
Model predictive control
Neural networks
Radial basis function

ABSTRACT

In this work, a novel model predictive control (MPC) scheme is introduced, by integrating direct and indirect neural control methodologies. The proposed approach makes use of a robust inverse radial basis function (RBF) model taking into account the applicability domain criterion, in order to provide a suitable initial starting point for the optimizer, thus helping to solve the optimization problem faster. The performance of the proposed controller is evaluated on the control of a highly nonlinear system with fast dynamics and compared with different control schemes. Results show that the proposed approach outperforms the rivaling schemes in terms of response; moreover, it solves the optimization problem in less than one sampling period, thus effectively rendering MPC-based controllers capable of handling systems with fast dynamics.

1. Introduction

Artificial neural networks (ANNs or simply NNs) [1] are considered an ideal solution for the modeling of highly nonlinear systems or processes, and during the last twenty years they have been extensively used for the realization of such models [2–5], in order to design novel control schemes. The success of NN-based controllers originates from their inherent ability to model unknown systems or processes by applying specialized training algorithms exclusively on experimental data, providing an outstanding alternative in cases where conventional methods fail to form appropriate control laws.

Among the different NN architectures used to design control schemes, radial basis function neural networks (RBFNNs) [6] present many advantages, including increased accuracy, better interpolation capability, simpler structures and faster training algorithms [7,8]. These particular characteristics have made RBFNNs a preferred choice in formulating state-of-the-art monitoring [9,10] and control [3,5,11] schemes. On the other hand, the main disadvantage of all black-box modeling techniques including NNs, is extrapolation [12], a phenomenon appearing in cases where the training dataset is not sufficiently covering the input space. Extrapolation results in unreliable predictions, which can ultimately lead the system to instability.

There are two main design approaches when it comes to implementing NN-based control strategies, namely direct design and indirect design. In indirect design control techniques the NN acts as a dynamical model of the system, predicting the system state vector and/or outputs [13], whereas in the case of direct design, the NN approximates the inverse dynamics of the system [14] and acts directly as a controller. Indirect design is usually integrated in an appropriate model predictive control (MPC) methodology [15]; in this case an NN-based nonlinear dynamic system model is constructed using historical input-output data, so that receding horizon predictions can be successfully obtained [16–19]. An optimization problem must then be formulated and solved, in order to obtain the optimal series of actions, so as to drive the system to the desired state. MPC techniques can manage MIMO systems, while also taking into account input-output-state restrictions and model-system mismatches; due to these advantages, MPC has found many successful applications in diverse fields [20–22].

Despite all their merits, MPC-based indirect design control methodologies share a significant drawback, namely the restriction that the nonlinear optimization problem must be solved in real-time [15]. The time required to solve the on-line optimization problem must be less than the sampling time period, so that the control scheme has enough time to obtain the control action and apply it to the system. If the solution of the optimization problem requires more time, there is a risk of control failure that may lead to instability, as the optimal value of the manipulated variable will not be computed and applied on time. However, the available time window between two consecutive control steps may not even be

* Corresponding author.
E-mail address: alexx@teiath.gr (A. Alexandridis).

http://dx.doi.org/10.1016/j.isatra.2017.09.016
0019-0578/

Please cite this article as: Stogiannos M, et al. Model predictive control for systems with fast dynamics using inverse neural models. ISA Transactions (2017), http://dx.doi.org/10.1016/j.isatra.2017.09.016

adequate for solving a sub-optimal MPC [23]. For these reasons, standard MPC methodologies are not applicable to systems with fast dynamics. The explicit MPC [24,25] technique was invented in order to overcome this problem, by partitioning the input space and assigning a different optimal control law in each region. Current explicit MPC theory guarantees global optimality for linear systems [26], but application on nonlinear systems requires approximating techniques [27], which may be even more time-consuming than solving the real-time optimization problem itself. The reduction of the computational cost in MPC methodologies is the prime aspect of current relevant literature [28,29], but, as of this time, fast optimization algorithms [30,31] are limited to linear systems. An interesting alternative for achieving results that can be similar to those produced by fully nonlinear MPC controllers in terms of performance, while at the same time significantly reducing the computational burden, is to linearize online the model around the current operating point, thus resulting to a quadratic optimization problem. Due to its effectiveness and simplicity, the online linearization approach has found many successful industrial applications [32–36]. A different method which eliminates the computational burden associated with solving the optimization problem, involves the NN approximation of the suboptimal control signal of MPC formulations [37,38].

On the other hand, direct design control techniques avoid altogether the optimization problem by using the NN as an explicit control law, directly predicting the manipulated variable values that are used as system inputs at each control step [4], while previous input-output-state values comprise the inverse neural model's input vector. The performance of direct design control methodologies stems from the fact that their implementation is as simple as computing a nonlinear function at each time step, without the hassle of solving an optimization problem. One of the most common direct control schemes are the inverse neural controllers (INCs), which have been extensively used in modern applications, as they are very fast in calculating control actions [4]. Other implementations suggest that model parameter adaptation with offset-free control is also possible in real-time [39]. A more recent study [40] has shown that INCs can be made robust in multiple ways, in order to avoid extrapolation, suppress any steady-state error, reject external disturbances, adapt to unknown system parameters and account for system-model mismatches.

Notwithstanding the aforementioned advancements offered by direct approaches regarding disturbance handling and steady-state error elimination, it should be noted that these methods usually deliver just one feasible trajectory towards the setpoint, totally ignoring the aspect of optimality. Indirect methods are usually better equipped to handle the latter, albeit at the cost of abolishing the obvious advantages in terms of control action calculation speed offered by direct methods. The scope of this work is the formulation of a novel indirect control scheme, in a way that retains all the advantages of nonlinear MPC, while also tackling its main disadvantage, which is the increased time required to solve the optimization problem. In order to do that, an inverse dynamic RBFNN-based system model is built and robustified with the applicability domain (AD) technique, which enhances the prediction reliability of the model [40]. The inverse model is incorporated to the MPC scheme, in order to provide a suitable initial vector to the optimization problem, aiming to decrease the solution time. The proposed controller performance is compared to a direct inverse neural controller employing the applicability domain and an error-correcting technique (INCADEC) [40], a nonlinear MPC controller and a discrete PID (DPID). All control formulations are evaluated on the control of the nonlinear system of the inverted pendulum on cart [41], which is a well-known benchmark for automatic control methodologies sharing a common operating principle with many commercial, industrial and military applications. To perform the evaluation, appropriate control scenarios are used to test set-point tracking and disturbance rejection, the controllers' stand-up and balancing capabilities, as well as their ability to handle noise and system/model mismatches.

The rest of this paper is organized as follows. The next section describes the radial basis function architecture, as well as the selected training algorithm. The third section presents the theory behind the INNEM initialization routine, as well as its incorporation into the MPC framework. Section 4 presents the test cases and thoroughly discusses the results. Finally, the last section provides the concluding remarks produced after the test cases are examined and interpreted.

2. Radial basis function neural network architecture and training techniques

RBF neural networks belong to the feedforward neural network architectures. The main difference of RBF networks compared to the well-known multi-layer perceptron (MLP) architecture is that they employ only one hidden layer, every node of which makes use of a radially symmetrical activation function in order to compute the hidden node response. The advantages of RBFs over other feedforward architectures include simple network structures, fast training algorithms and increased prediction accuracy, but on the side effect their extrapolation ability is smaller. This drawback is alleviated to a great extent by the proposed methodology.

The RBF architecture consists of three layers, namely the input, the hidden and the output layer, as shown in Fig. 1. The input layer distributes data from the N input variables to the L hidden layer nodes. All hidden nodes correspond to a center vector representing the center of the RBF in the input space. In this context, one can see that the hidden layer performs a nonlinear transformation of the input space to a new space of usually higher dimensionality. The input $\mu_l(\mathbf{u}_k)$ to the l-th hidden node (called activity) is a distance metric between the k-th input vector $\mathbf{u}_k$ and the hidden node center vector $\mathbf{c}_l$. The distance metric employed in this work is the Euclidean distance.

$$\mu_l(\mathbf{u}_k) = \|\mathbf{u}_k - \mathbf{c}_l\|_2 = \sqrt{\sum_{i=1}^{N}\left(u_{i,k} - c_{i,l}\right)^2},\ l = 1, 2, \ldots, L,\ \ k = 1, 2, \ldots, K \quad (1)$$

where K is the number of available training samples, $\mathbf{u}_k = \left[u_{1,k}, u_{2,k}, \ldots, u_{N,k}\right]^T$ is the input data vector and $\mathbf{c}_l = \left[c_{1,l}, c_{2,l}, \ldots, c_{N,l}\right]^T$ is the node center vector, respectively. The activation function

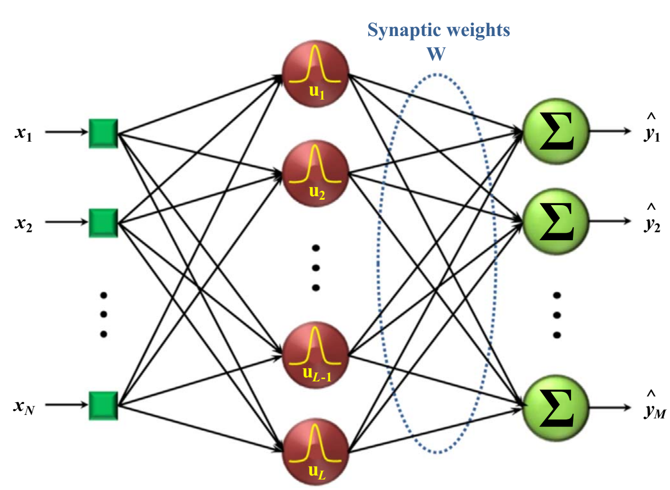

Fig. 1. Typical structure of an RBFNN with Gaussian basis functions.

Please cite this article as: Stogiannos M, et al. Model predictive control for systems with fast dynamics using inverse neural models. ISA Transactions (2017), http://dx.doi.org/10.1016/j.isatra.2017.09.016