

Development and Real-Time Implementation of Recurrent Neural Networks for AFR Prediction and Control

Ivan Arsie, Cesare Pianese and Marco Sorrentino

University of Salerno

Copyright © 2008 SAE International

ABSTRACT

The paper focuses on the experimental identification and validation of recurrent neural networks (RNN) for real-time prediction and control of air-fuel ratio (AFR) in spark-ignited engines. Suited training procedures and experimental tests are proposed to improve RNN precision and generalization in predicting both forward and inverse AFR dynamics for a wide range of operating scenarios. The reference engine has been tested by means of an integrated system of hardware and software tools for engine test automation and control strategies prototyping. The comparison between RNNs simulation and experimental trajectories showed the high accuracy and generalization capabilities guaranteed by RNNs in reproducing forward and inverse AFR dynamics.

Then, a fast and easy-to-handle procedure was set-up to verify the potentialities of the inverse RNN to perform feed-forward control of AFR. Preliminary experimental tests indicate how the inverse RNN controller performance are comparable and in some cases even better than those guaranteed by the commercial ECU the reference engine is equipped with. Therefore RNN-based control of AFR emerges as a high potential alternative to reduce calibration efforts and to improve control performance as compared to the currently adopted techniques.

INTRODUCTION

Air-fuel ratio is a key challenge and keeps being an open problem for the engine control community. Since the eighties, the transition from carburetor to electronically controlled injection system motivated the researcher to concentrate on this topic. Proper control of air-fuel ratio is greatly beneficial to the three way catalyst performance in both steady and transient operations. Therefore this control task plays a fundamental role in limiting exhaust emissions in SI, GDI and lean-burn engines.

Despite the considerable efforts made, the more stringent environmental regulations, imposed throughout the world, make the achievement of satisfactory

conversion efficiency still an ambitious goal. Furthermore, the engine control system (ECS) designers have to deal with the on board diagnostics (OBD) requirements, introduced since 1996 in US and later in Europe. This represents a more critical goal in the field of automotive control, since it requires to continuously monitor all powertrain components to prevent critical faults with respect to exhaust emissions.

AFR control currently relies on a mean value engine model representation. The outstanding works of Aquino [1] and Hendricks and Sorenson [2] represent the foundation of the actual AFR controller implemented on mass market vehicles. Such a controller estimates in a feedforward way the air flow rate at the injector location, providing the right amount of fuel to be delivered with the appropriate time dependence. The implementation of this control within a feedback compensation loop allows further correction of the injection time based on the measurements performed by the oxygen sensor.

Despite their accuracy, mean value models have some significant limitations, such as the high experimental burden requested for parameters identification and the intrinsic non-adaptive features. To overcome the latter problem, adaptive methodologies have been proposed in order to estimate the states and tune the parameters making use of real-time measurements (e.g. observers, Kalman filters) [3–5] or robust control methodologies (e.g. H_∞ control) [6]. On the other hand, the AFR signal delay represents a very critical issue to be overcome to improve the performance of the closed-loop control strategies [4, 7, 8].

A promising solution for approaching these problems is given by Neural Networks based black-box models. They have high mapping capabilities and guarantee a good generalization even with a reduced set of identification data. Moreover, the possibility of implementing adaptive training procedures allows to take into account the influence of exogenous effects on control performance [9–11].

Some examples of NN models for automotive application have been proposed in the literature for engine control [8, 12], diagnostics [13,14]. Neural networks have also

been proven useful for modeling nonlinear dynamic systems introducing feedback connections in a recursive computational structure (i.e. RNN). In the engine modeling and control fields, of particular interest are the contribution of [15–17].

Main contributions of the present paper are numerical/experimental methodologies aimed at developing accurate RNN models of AFR dynamics. Well-known problems associated with network training, such as data overfitting and overtraining, are addressed. Moreover, an error-based procedure previously performed by the authors [18] allowed identifying and removing the measuring delay from the AFR signal. Such an approach contributes, on one hand, to enhance RNN identification and, on the other, to make RNN suitable to perform feedback control of AFR. Then a neural network control architecture is developed, made up of two RNNs: an inverse one, acting as fuel injection controller, and a forward one, used as virtual AFR sensor [19] to provide the feedback signal. Finally, real-time implementation of the proposed controller is performed to verify control performance with respect to commercial ECU.

POTENTIALITIES OF RNN FOR AFR ESTIMATION

The actual state of the art of AFR control relies on both feedforward and feedback control strategies. The base injection pulse is computed as function of desired AFR, engine speed, manifold pressure, throttle opening and feedback from the exhaust oxygen sensor. Furthermore, a number of factors are introduced to account for changes in ambient conditions, battery voltage and engine thermal state. During transient operation, in order to compensate for the different dynamic response of air and fuel path due to the wall wetting phenomenon, the base injected fuel is compensated by a feedforward controller. This latter is usually based on a mean value model (MVM) according to the approach originally proposed by Aquino [1]. The MVM parameters are computed accounting for the influence of engine operation and thermal state on fuel wall wetting and engine breathing, respectively. The output of the MVM-based feedforward controller is then further corrected by a closed loop control task and a long term fuel trim. The former is based on a PI controller aimed at keeping the AFR measured by the exhaust oxygen sensor as close as possible to the desired value, as addressed by the catalyst control strategies [20]. Similarly to the MVM parameters, the PI gains are stored in look-up tables to account for their dependence on engine operating conditions. Long term fuel trim is intended to compensate for imbalance and/or offset in the AFR measurement due to exogenous effects, such as canister purge events, engine wearing, air leaks in the intake manifolds, etc. [21].

The control strategy described above has been well established in the automotive industry for long time, thanks to its good performance during both steady and transient conditions. Nevertheless, the more stringent regulations imposed in the last years (i.e. OBD and Low Emission Vehicles) have pushed car manufacturers towards further developing the actual control strategies implemented on commercial ECU. On the other hand, design burden has increased and the reduction of calibration efforts, required to develop the parameter maps and to improve the feedback control performance, has become a stringent need.

Recurrent Neural Network, whose modeling features are presented in the following section, have significant potential to face the issues associated with AFR control. The authors themselves and other contributions [17, 22] showed how an inverse controller made of two RNNs, simulating both forward and inverse intake manifold dynamics, is suitable to perform the feedforward control task. Such architectures could be developed making use of only one highly-informative data-set [23], thus reducing the calibration effort with respect to Aquino-based approaches. Moreover, the opportunity of adaptively modifying network parameters allows accounting for other exogenous effects, such as change in environmental conditions, construction tolerances and engine wear.

Besides their high potential, when embedded in the framework of pure neural-network controller, RNN AFR estimators can also be suitable in virtual sensing applications, such as the prediction of AFR in cold-start phases. RNN training during cold-start can be performed on the test-bench off-line, by pre-heating the lambda sensor before turning on the engine. Moreover, proper post-processing of training data enables to predict AFR excursions without the delay between injection (at intake port) and measuring (in the exhaust) events, thus being suitable in the framework of sliding-mode closed-loop control tasks [7, 20]. In such an application the feedback provided by a UEGO lambda sensor should be used to adaptively modify the RNN estimator to take into account exogenous effects. Finally, RNN-based estimators are well suited for diagnosis of injection/air intake system and lambda sensor failures [24]. Differently from control applications, in this case it is envisioned that the AFR prediction includes the measuring delay.

RECURRENT NEURAL NETWORK FOR AFR PREDICTION

The RNNs are derived from the static multi layer perceptron feed forward (MLPFF) networks by adding feedback connections among the neurons, thus introducing a dynamic effect in the computational system. Therefore, due to the non-linear mapping features of the MLPFFs, the RNNs are suitable for black-box non-linear dynamic modeling [9, 11].

Depending upon the feedback typology, which can either involve all the neurons or only the ones located in the output and input layers, RNNs are classified into global, local or external recurrent neural networks [9]. The latter is the class used in this work with one hidden layer of nodes. The sketch of Fig. 1 gives the schematic representation of the following general form:

$$\hat{y}(t+1, \theta) = F[\hat{y}(t, \theta), \dots, \hat{y}(t-n+1, \theta), u(t), \dots, u(t-m+1), x(t), \dots, x(t-m+1)] \quad (1)$$

where $\hat{y}$, u and x are the output, control and external input variables, respectively; while θ is the vector of parameters (weights). The indices n and m are set once the lag spaces of both external input x and feedback variable $\hat{y}$ have been fixed. As an example, in Fig. 1 the lag space values are set equal to 2. The terms in square brackets of Eq. (5) compose the RNN regressors vector, which is representative of the past process dynamics. F is the non-linear mapping operation performed by the neural network. The RNN described above is known in the literature as nonlinear output error model (NOE) [10].

Network training is performed by minimizing a cost function estimated as function of the mean squared error (MSE). A weight regularization term is added to improve model generalization:

$$E(\theta) = \frac{1}{2N} \sum_{t=1}^N (\hat{y}(t|\theta) - y(t))^2 + \frac{1}{2N} \cdot \theta^T \cdot \alpha \cdot \theta \quad (2)$$

where α is a scalar and represents the weight decay. The above function minimization can be conducted in either a batch or an adaptive way. The former is usually preferred at the initial development stage, whereas the latter is greatly beneficial to online RNN implementation as it allows to adapt network weights to the exogenous variations of the controlled/simulated system.

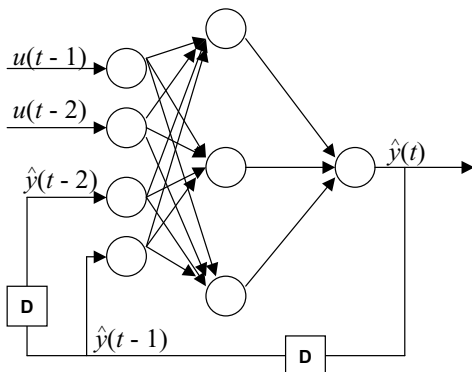


Fig. 1 – Structure of NOE RNN with one input variable, one output, one hidden layer and two output delays D .

For the current application, a Forward RNN Model (FRNNM) of AFR dynamics can be developed

considering that the dynamic processes affecting AFR transient response depend on both air and fuel dynamics. Therefore, the output, control and external input variables of the FRNNM virtual sensor are, respectively:

$$\begin{aligned} \hat{y} &= AFR \\ u &= t_{inj} \\ x &= [rpm, p_{man}] \end{aligned} \quad (3)$$

Therefore, the FRNNM resulting from Eq. (1) will be:

$$\begin{aligned} A\hat{F}R(t, \theta) = F[& A\hat{F}R(t-1, \theta_1), \dots, A\hat{F}R(t-n, \theta_1), t_{inj}(t-1), \\ & \dots, t_{inj}(t-m), rpm(t-1), \dots, rpm(t-m), \\ & \dots, p_{man}(t-1), \dots, p_{man}(t-m)] \end{aligned} \quad (4)$$

It is worth to note that the output feedbacks in the regressors vector (i.e. $A\hat{F}R$ in Eq. 4) are simulated by the network itself, thus the FRNNM does not require any AFR measurement as feedback to perform the online estimation. This represents a very appealing feature that makes the NOE structure a suitable solution for AFR virtual sensing when the oxygen sensor does not guarantee an accurate measurement, as it happens during cold start phases.

RECURRENT NEURAL NETWORK ARCHITECTURE FOR AFR CONTROL

Neural Network based control systems are classified into two main typologies: a) Direct Control Systems and b) Indirect Control Systems [10]. In the first case a) the neural network is used as controller and its output directly determines the control actions. In the second case b), the NN output is processed by an optimization algorithm to determine the optimal control actions. For a detailed analysis of the main neural control architectures, the reader is addressed to a previous paper [22] and to the selected bibliography [10, 25].

In this paper, an RNN architecture of class a) was selected, namely the Inverse Model Neural Controller (IMNC) [22, 25]. Usually, in the IMNC structure the control actions are computed by an Inverse RNN Model as function of the feedbacks measured by the sensors and the external past inputs [10, 22, 25]. In the current application, the feedbacks are provided by the FRNNM virtual sensor presented above, which will thus replace the lambda sensor. This solution is particularly suitable for AFR control applications since it allows removing the delay due to engine cycle, transport phenomena and sensor response, as it will be discussed more deeply in the next sections.

Fig. 2 shows a block diagram of the closed-loop neural controller resulting from the integration between IRNNM and virtual sensor (i.e. FRNNM). The output values predicted by the FRNNM are fed as feedbacks to the