Z. Hou and P. J. Antsaklis, "Identification and Control of a Powertrain System Using Neural Networks,"
P roc o f t he S ymposium o n I mplicit a nd N onlinear S ystems , Univ of Texas at Arlington, Ft. Worth, TX,
Dec. 14-15, 1992.

Identification and Control of a Powertrain System Using Neural Networks

Zugang Hou and Panos J. Antsaklis

Department of Electrical Engineering
University of Notre Dame
Notre Dame, IN 46556, USA

Abstract

An auto powertrain transfers the torque produced by the engine through the clutch to the load, thus accelerating the vehicle to the desired speed. Good models are essential in predicting the clutch heat generation so to avoid overheating and lockup of the clutch plates. In this paper, neural networks are used to first model and then control the highly nonlinear MIMO clutch- load subsystem. The two inputs were the applied force on the clutch plate and the output torque of the torque converter; the two outputs are the angular velocities of the torque converter and of the load, beyond the gear box.

1 Introduction

The function of a powertrain system is to transfer the torque produced by the engine through the torque converter, the clutch package and the gear box to the load, accelerating the vehicle to the desired speed, see Fig. 1. The research presented in this paper concentrates on modeling and controlling a powertrain subsystem , the clutch package. The design goal is to control the system so that the maximum heat power generated in the clutch plates or the total energy dissipated during the clutch engagement period do not exceed certain values.

Due to the heavy demands on the powertrain system, the powertrain system of certain heavy duty vechicles can fail in the practical situations. The most vulnerable part is the clutch plate package. Currently, most of the encounted clutch failures are caused by overheating and lockup of the clutch plates [6]. Either the extreme high heat peak power or too much heat energy dissipated during the clutch engagement period can result in the clutch failure problems. So the problem arises: how to control the clampping force, so that there will be no peak heat power exceeding the allowable value, and minimize the heat energy dissipated in the clutch package within certain amount of engagement time (less than 1.5 second). This paper concentrates on the modeling and control of the highly nonlinear multiinput and multioutput (MIMO) clutch - load subsystem. The two inputs considered are the applied force on the clutch plate and the output torque of the torque converter; the two outputs are the angular velocities of the torque converter and of the load, beyond the gear box.

Many results have been published in powertrain system modeling and control. In Wouters's paper [19], a method to calculate energy dissipation and temperature distribution of a multidisc clutch in powershift transmissions is provided. Dundore [4] gives a method to establish a failure criterion for a wet clutch. Jania [8] analyzes the performance of friction clutches with focus on the problem of vibration. Johnson [9] develops and experimentally verifies a transient heat transfer model which could predict the temperature as a function of time. Using bond graphs, Hrovat [7] develops a linearized model of a typical powertrain system with manual transmission. Moskwa [10] [12] discusses nonlinear methods that would provide a more pleasing shift with a sliding mode controller. Cho [3] also describes a sliding mode control method to deal with uncertainties. The models and the analysis methods mentioned above are valid only to a certain extent. Due to uncertainties in powertrain systems, analysis techniques are restricted. Although some general models or partial models are available, many assumptions are made, which limit the use of these models. Due to the lack of accurate models and effective analysis methods, many problems remain unsolved. Since neural networks have the ability to deal with nonlinear dynamic system, it may be possible for neural networks to enhance the traditional modeling and control methods in powertrain nonlinear dynamic systems, and this is examined in the paper.

Section 2 deals with methods to model the powertrain subsystem. Both a mechanical model (first principal model) and a neural network model will be discussed. A neural network controller designed for the powertrain clutch - load subsystem is introduced at Section 3. Simulation results are presented.

2 Modeling the Powertrain Clutch - Load Subsystem

2.1 Mechanical Model

A typical powertrain system consists of an engine, a torque converter, a clutch package, a gear box and a load. A simple block diagram of the whole powertrain system

Proc of the Symposium on Implicit and Nonlinear Systems, Ft. North TX, Dec 14-15, 1992.

Z. Hou and P. J. Antsaklis, "Identification and Control of a Powertrain System Using Neural Networks,"
P roc o f t he S ymposium o n I mplicit a nd N onlinear S ystems , Univ of Texas at Arlington, Ft. Worth, TX,
Dec. 14-15, 1992.

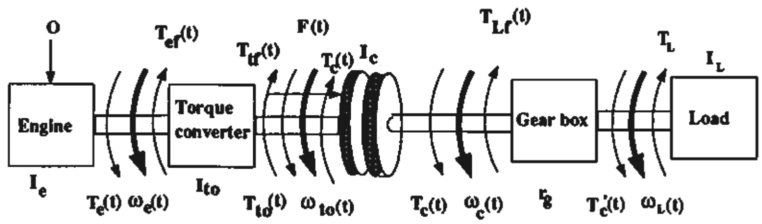

Figure 1: Block diagram of entire powertrain system

is shown in the Fig. 1.

In Fig. 1, the following quantities occur:

I_e - Moment of inertia of the engine output.
$T_e(t)$ - Output torque of the engine.
$\omega_e(t)$ - Output angular speed of the engine.
$T_{ef}(t)$ - Damping torque at the engine output side.
O - External input variables of the engine torque (throttle position, angle of ignition, etc.)
I_{to} - Moment of inertia of the torque converter at output side.
$T_{to}(t)$ - Output torque of the torque converter.
$\omega_{to}(t)$ - Output angular speed of the torque converter.
$T_{tf}(t)$ - Damping torque at the torque converter output side.
I_c - Moment of inertia of the clutch package output.
$T_c(t)$ - Output torque of the clutch package.
$T_c(t)'$ - Output torque of the gear box.
$\omega_c(t)$ - Output speed of the clutch package output.
$\omega_L(t)$ - Output speed of the load.
$T_{Lf}(t)$ - Damping torque in the clutch package - load side.
I_L - Moment of inertia of the load.
$T_L(t)$ - External load torque.
$F(t)$ - Clamping force of the clutch.
r_g - Transmission gear ratio.

Note that the inertia of the gear box is considered as part of the clutch package inertia. The gear ratio can be assumed to be one without loss of generality which implies that $\omega_c(t)$ should be equal to $\omega_L(t)$. (In the case of a ratio different from one, the load can be transformed using r_g^2.)

The engine is the power source of the whole system. The output variables are speed and torque of the load. These two variables depend on several factors. The only free variable is the engine torque. It depends on the throttle position, the engine speed, the angle of ignition, the engine temperature, the engine lubrication and other less essential quantities. For simplicity, the engine can be considered as a torque generator with a certain moment of inertia as shown in Fig. 2.

The torque converter acts as a buffer between the engine and the clutch package. Ideally, the torque converter transforms the input variables, input speed and input torque to the output variables, output speed and output torque, in a lossless fashion, i.e.

$$T_{ti}(t)\omega_{ti}(t) = T_{to}(to)\omega_{to}(t). \tag{1}$$

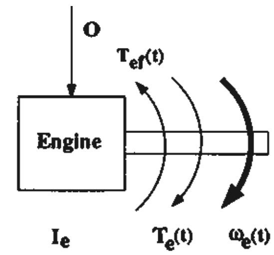

Figure 2: Simplified engine model.

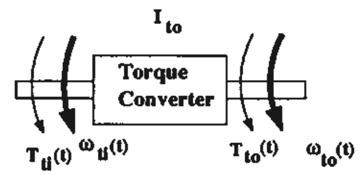

Figure 3: Important variables for the description of the torque converter

In reality, there is some energy loss in the torque converter, i.e.

$$T_{ti}(t)\omega_{ti}(t) = T_{to}(to)\omega_{to}(t) + P(\omega_{ti}(t)/\omega_{to}(t)), \tag{2}$$

where $P(\omega_{ti}(t)/\omega_{to}(t))$ is the dissipated power in the torque converter. The block diagram of the torque converter is shown in Fig. 3

The clutch package includes the clutch plates and the gear box. During the engagement of the clutch, the input torque from the torque converter output can be transferred to the load shaft through the clutch package.

Since the maximum available torque at the engine output strongly depends on the engine speed, and the engine speed is limited within the range : $\omega_{min} \leq \omega_e(t) \leq \omega_{max}$, a transmission system is used which provides a wide speed range for the load by choosing different transmission ratios.

The load consists of a moment of inertia due to the body of the vehicle and an optional constant external torque, which represents an external force acting on the vehicle.

The whole powertrain system can be treated as three cascaded subsystems. i.e. engine - torque converter, torque converter - clutch package, clutch package - load. These three subsystems are coupled by the torque converter and the clutch package. The coupling through the clutch package is created by friction, creating the transmitted clutch torque $T_c(t)$. For now, the friction coefficient μ is assumed to be constant. The coupling created by the torque converter is of a more difficult nature. The input and output of the torque converter are loosely connected through fluid coupling.

Next, the equation model is set up using the laws of circular motion.

The engine - torque converter subsystem is described

Z. Hou and P. J. Antsaklis, "Identification and Control of a Powertrain System Using Neural Networks," P roc o f t he S ymposium o n I mplicit a nd N onlinear S ystems , Univ of Texas at Arlington, Ft. Worth, TX, Dec. 14-15, 1992.

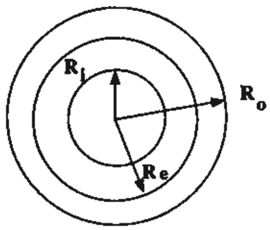

Figure 4: The clutch plate geometry

by

$$I_e \frac{d\omega_e(t)}{dt} = T_e(t) - T_{ti}(t) - T_{ef}(t). \tag{3}$$

where $T_{ti}(t)$ is the input torque of the torque converter.

The relationship for the torque converter - clutch subsystem is given by

$$I_{to} \frac{d\omega_{to}(t)}{dt} = T_{to}(t) - T_c(t) - T_{tf}(t). \tag{4}$$

The clutch - load subsystem is characterized by

$$I_c \frac{d\omega_L(t)}{dt} = T_c(t) - T_{Lf}(t) - T_L(t). \tag{5}$$

There is a relationship between the damping torque and the angular speed. A first order approximation of the damping torques can be expressed as follows:

$$T_{ef}(t) = K_e \omega_e(t), \tag{6}$$

$$T_{tf}(t) = K_{tf} \omega_{to}(t), \tag{7}$$

$$T_{Lf}(t) = K_L \omega_L(t), \tag{8}$$

where,

K_e : Damping torque coefficient of engine - torque converter subsystem.

K_{tf} : Damping torque coefficient of torque converter - clutch package subsystem.

K_L: Damping torque coefficient of clutch package - load subsystem.

The above three damping coefficients are considered to be constants, which is only true for the neighborhood around an operating point. Otherwise all three damping coefficients are functions of the corresponding speed.

The torque $T_c(t)$ transmitted by the clutch is a function of the effective overall friction coefficient μ, the clamping force $F(t)$ and the effective radius R_e

$$T_c(t) = \mu F(t) R_e. \tag{9}$$

Where the effective radius R_e is given by the following equation and is illustrated in Fig. 4:

$$R_e = \frac{2(R_0^3 - R_i^3)}{3(R_0^2 - R_i^2)}. \tag{10}$$

Note that the clutch-load subsystem acts as two dynamic systems with some internal dynamic relations before engagement; these two systems act as a new system with a combined moment of inertia after engagement.

Then the model of the clutch - load subsystem is given by:

$$\begin{cases} \frac{d\omega_{to}(t)}{dt} = \frac{T_{to}(t)}{I_{to}} - \frac{K_{tf}}{I_{to}}\omega_{to}(t) - \frac{T_c(t)}{I_{to}}, & \\ \frac{d\omega_L(t)}{dt} = \frac{T_c(t)}{I_L} - \frac{K_{Lf}}{I_L}\omega_L(t) - \frac{T_L(t)}{I_L}, & \text{if } \omega_{to}(t) > \omega_L(t). \\ \frac{d\omega_L(t)}{dt} = \frac{T_c(t)}{I_L+I_{to}} - \frac{K_{Lf}}{I_L+I_{to}}\omega_L(t) - \frac{T_L(t)}{I_L+I_{to}}, & \text{if } \omega_{to}(t) = \omega_L(t). \end{cases} \tag{11}$$

This model consists of two sets of differential equations. Due to the assumptions in the above derivation, if these differential equations are looked at separately, they are linear systems. But due to the change of the system structure, the parameters of the system (the moment of inertia, I_{to}, I_L and $I_L + I_{to}$)are different before the slip speed reaches zero and after that. This changing of the system structure leads to the nonlinear characteristics of the system. It can be seen that this is a simplified model since it is built under the assumptions that the friction coefficient is constant during the engagment, the drive shaft is a rigid body and that a first order approximation of the damping torques is used. For this paper, this simplified nonlinear model will be used to generate the input/output data.

2.2 Modeling Using Neural Networks

Modeling the input - output behavior of nonlinear plants using neural networks has gained ground in recent years. This is mainly because neural networks can also deal with cases where mathematical models are either very poor or nonexistent and because neural networks can easily be updated via learning procedures to reflect changes in the physical plant. Feed forward multilayer neural networks are the most common architectures with hyperbolic tangent or sigmoidal functions being the most common activation functions in the neural units. The success of the modeling procedure is based on the fact that such networks can approximate with arbitrary accuracy any nonlinear static map under certain mild assumptions. Note that neural networks in modeling and control of dynamical systems have been the topic of the Special Issues in IEEE Control Systems Magazine Antsaklis[1]. A brief review of this research can be found in Antsaklis[2].

Modeling, or neural identification, is seen as the process of constructing a model of a dynamic system based on available system information namely input/output data. The model parameters are adjusted based on system output error, at least over the input range of interest. The system to be identified is seen as a black box with bounded input/bounded output stability. As the identification process begins, the plant output and the neural network model output are compared and an error term is produced. Based on this error, the parameters (neural network weights) of the neural network model are adjusted according to the gradient methods and the back propagation algorithm.

The neural network is trained using a random signal which is uniformly distributed over certain ranges. Since

Z. Hou and P. J. Antsaklis, "Identification and Control of a Powertrain System Using Neural Networks,"
P roc o f t he S ymposium o n I mplicit a nd N onlinear S ystems , Univ of Texas at Arlington, Ft. Worth, TX,
Dec. 14-15, 1992.

this method provides rich excitation to the plant, the neural network can capture most of the relevant properties of the plant over the input range of interest. So random input is used throughout system identification simulations conducted here. After training is completed, a test signal, such as an irregular sinusoid wave, a step input etc., is used to test the trained neural network.

The equations of a neural network identifier can be described as follows. The same equations can be used to describe a neural network controller.

The equations are based on a two hidden layer neural network structure.

The following notation will be used:

N - The number of neurons on the output layer.
n_2 - The number of neurons on the second hidden layer.
n_1 - The number of neurons on the first hidden layer.
I_n - The number of neurons on the input layer.
m - Indicates m^{th} output layer, here the output layer.
$y_j^m[k+1]$ - The j^{th} neuron output on the output layer at time $k+1$.
x_i^{m-1} - The i^{th} neuron output on the second hidden layer.
R_l^{m-2} - The l^{th} neuron output on the first hidden layer.
$U_s[k]$ - The s^{th} neuron output on the input layer at time k.
w_{ij}^m - The connection weights from the second hidden layer to the output layer.
w_{li}^{m-1} - The connection weights from the first hidden layer to the second hidden layer.
w_{sl}^{m-2} - The connection weights from the input layer to the first hidden layer.

The output of the neural network, that is of the output layer is:

$$y_j^m = f_o[\sum_{i=1}^{n_2} w_{ij}^m x_i^{m-1}], \tag{12}$$

where $f_o(x) = x$, $j = 1, 2, \cdots, N$.

The output of the second hidden layer is:

$$x_i^{m-1} = f_2[\sum_{l=1}^{n_1} w_{li}^{m-1} R_l^{m-2}], \tag{13}$$

where $f_2 = \tanh(x)$, $i = 1, 2, \cdots, n_2$

The output of the first hidden layer is:

$$R_l^{m-2} = f_1[\sum_{s=1}^{I_n} w_{sl}^{m-2} U_s], \tag{14}$$

where $f_1(x) = \tanh(x)$, $l = 1, 2, \cdots, n_1$.

So, by combining these equations, the following expression is obtained for a general neural network identifier or a neural network controller:

$$y_j^m[k+1] = f_o[\sum_{i=1}^{n_2} w_{ij}^m f_2[\sum_{l=1}^{n_1} w_{li}^{m-1} f_1[\sum_{s=1}^{I_n} w_{sl}^{m-2} U_s(k)]]], \tag{15}$$

where

$f_o(x) = x$, $f_1(x) = \tanh(x)$, $f_2(x) = \tanh(x)$, $j = 1, 2, \cdots, N$.

In general, y_j^m represents the output vector of the neural network identifier or the output vector of the neural network controller.

On the basis of the Delta Rule (a steepest descent algorithm with a constant step size), Rumelhart developed the back propagation algorithm. The multilayer cost function C is described as follows:

$$C = \sum_{p=1}^{n_p} C_p = \sum_{p=1}^{n_p} \frac{1}{2} \sum_{j=1}^{N(o)} [d_{pj} - y_{pj}]^2 \tag{16}$$

where $N(o)$ is the number of output neurons, d_{jp} is the desired output of the j^{th} output neuron for the p^{th} input pattern and y_{pj} is the output of the j^{th} output neuron for the p^{th} input pattern. The output of the j^{th} neuron for the m^{th} layer is given by the equation:

$$y_{pj}^m = f(\sum_{i=1}^{N(m-1)} w_{ij}^m y_{pi}^{m-1}) = f(net_{pj}^m) \tag{17}$$

Where $N(m-1)$ is the number of neurons in the m^{th} layer and w_{ij}^m is the weight from the output of the i^{th} neuron in the $(m-1)^{th}$ layer to the j^{th} neuron in the m^{th} layer. Note that when the $(m-1)^{th}$ layer is the input layer, then the $y_{pi}^{m-1} = u_{pi}$ where u_{pi} is the i^{th} input of the p^{th} pattern. The nonlinear hyperbolic tangent function is given by

$$f(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}} \tag{18}$$

The back propagation algorithm to update the weights of the neural network is summarized as follows:

$$w_{ij}^m(k+1) = w_{ij}^m(k) - \eta \frac{\partial C}{\partial w_{ij}^m} + \alpha \Delta w_{ij}^m(k)$$

$$= w_{ij}^m(k) + \eta \sum_{p=1}^{n_p} \delta_{pj}^m y_{pi}^{m-1} + \alpha \Delta w_{ij}^m(k) \tag{19}$$

where

$$\delta_{pj}^m = \begin{cases} \sum_{k=1}^{N(m+1)} \delta_{pk}^{m+1} w_{jk}^{m+1} (1 - (y_{pj}^m)^2) \\ \quad \text{if } m^{th} \text{ layer is the hidden layer} \\ (d_{pj} - y_{pj}^m)(1 - (y_{pj}^m)^2) \\ \quad \text{if } m^{th} \text{ layer is the output layer} \end{cases}$$

In the above equations, $\alpha \Delta w_{ij}^m(k)$ is the momentum term. The gradient tends to change along an average decent direction. This can reduce oscillations observed when the training step size is large. The coefficient of the momentum term, α, is usually between 0 and 1.

Here a nonlinear plant is considered as a black box and the only information which is generally available is

Z. Hou and P. J. Antsaklis, "Identification and Control of a Powertrain System Using Neural Networks,"
P roc o f t he S ymposium o n I mplicit a nd N onlinear S ystems , Univ of Texas at Arlington, Ft. Worth, TX,
Dec. 14-15, 1992.

the plant input/output data. Certainly, if additional information about the plant is known, this may make the identification process easier. Obviously, the less information needed to conduct the identification, the more general the method is.

The mechanical nonlinear model derived in section 2.1 has two inputs $T_{to}(t)$, $T_c(t)$ and two outputs $\omega_{to}(t)$ and $\omega_L(t)$. The clutch torque is expressed as $T_c(t) = \mu R_e F(t)$. It is assumed that μ is constant, so the clutch torque only changes by varying the clamping force $F(t)$. The effective radius of the clutch plates and load torque T_L are also assumed constant. The nonlinearity of the system has been discussed in section 2.1. To design a neural network controller, a neural network model of the system will be built first, based on the input/output data of the mechanical model. The input torques $T_{to}(t)$ and $T_c(t)$ will accelerate the load to the desired speed after engagement starts. The damping torques exist at the torque converter-clutch side and at the clutch-load side. The load torque is applied to the system and remains constant.

The parameters and the initial values of the variables are assigned as follows:

$k_{tf} = 0.02$	$\omega_{to}(0) = 2300$
$\mu = 0.05$	$T_{to}(0) = 2.433$
$K_{Lf} = 0.02$	$\omega_L(0^-) = 0$
$I_{to} = 0.01$	$F(0^-) = 0$
$I_L = 0.01$	$F(t) = 179$
$T_L = 1$	$R_e = 0.2$

The values of the parameters are not obtained from the actual system but are chosen based on expected transient responses of the actual system. The friction coefficient μ is about 0.05 for wet clutches. The effective radius of clutch is usually 0.2ft. The damping torque at the torque converter-clutch side is small compared to the one at the clutch-load side. The moment of inertia of the torque converter and clutch package is small compared to that of the load. The torque converter output torque should be large enough to drive the load and the clamping force should be large enough to make clutch engagement possible. The angular velocity of the torque converter output $\omega_{to}(t)$ and the load speed $\omega_L(t)$ should be chosen to be within the operating range which is typically between -2500 and 2500 rpm. The engagement period should be less than 1.5 seconds which is the typical time needed for full engagement. For identification of the real powertrain system the data should be obtained from actual experiments.

With the initial values of the system, at time $t = 0$, a clamping force is applied to the clutch and the engagement process begins. The input/output data of the system are generated from the mathematical model and are used to train the neural network model.

Based on the input/output data, a neural network is used to identify the dynamics of the system. The neural network model is expressed as follow:

$$\hat{y}(k+1) = f(u(k), u(k-1),$$
$$\cdots u(k-d), y(k), y(k-1) \cdots y(k-d)) \qquad (20)$$

where d is the estimated delay needed for the neural identification model. The input $u(k) = [T_{to}(k), F(t)]$ and the output $y = [\omega_{to(k)}, \omega_L(k)]$. The delay d needed depends on the simulation experiment. $\hat{y}(k+1)$ is the neural network plant output at k+1. The function f is a hyperbolic tangent function as in the equation (18). In the neural network simulation, if $d = 1$, the output of the neural network does not converge to the system output within a certain amount of iterations (100,000). If $d = 2$, good results can be achieved. Experiments have shown that if $d > 2$, the improvement in convergence time is small. Therefore, in the following simulations, $d = 2$.

In order to model this MIMO system, a neural network with two hidden layers (20, 10) is chosen. The backpropagation training algorithm is used. Two random training signals of $T_{to}(t)$ and $F(t)$ are applied to the neural network and the plant. Their values are taken to be uniformly distributed within the range $[0,3]$ and $[0,250]$ respectively. (Note the relationship $T_c(t) = \mu ReF(t)$.) We can not overstress the importance of the choice of the training signals. Note that the two inputs $T_{to}(k)$ and $F(t)$ cannot be random signals if the actual physical system is used to generate the data, since the mechanical constraints of the system do not allow severe fluctuation of these two variables, especially $T_{to}(t)$. Different training signals must then be applied, such as small step inputs, ramp inputs or other signals sufficiently rich in frequency. In general, when a neural network is trained by random inputs, good results are obtained but training time is relatively longer. If a neural network is trained by particular signals, then it may only work well as a model for input signals whose properties are close to that of the training signal; the training time for convergence, however is relatively short. Therefore, for a system which does not need a large variety of input signals, one can use specific signals for training purposes. This applies to the powertrain system under investigation. In fact one could originally train the neural network model using random inputs (off line by using an available model) and then continue training using the physical system and only determine the inputs (Off or on line to take into consideration information not present in the model used in the simulations). In this section, random signals are used. If the real mechanical constraints are added to the system model, then a specific set of practical training signals should be chosen to train the system.

During training, the identification step size is $\eta = 0.4$, which is the learning rate of the back propagation algorithm. The coefficient of momentum term $\alpha = 0.2$.

Training results with 100,000 iterations and 490,000 iterations are shown in Fig. 5 and Fig. 6 respectively. The test signal $T_{to} = 2.43$ and $F(t) = 179$ were applied at time $t = 0$. From Fig. 5, it can be seen that the output of the neural network tracks the output of the plants with large errors. This means that the neural network model has not captured the dynamics of the system completely. So more training iterations are needed. After more iterations, it can be seen from Fig. 6, the modeling error of the two outputs of the neural network model has improved.

Z. Hou and P. J. Antsaklis, "Identification and Control of a Powertrain System Using Neural Networks," Proc of the Symposium on Implicit and Nonlinear Systems, Univ of Texas at Arlington, Ft. Worth, TX, Dec. 14-15, 1992.

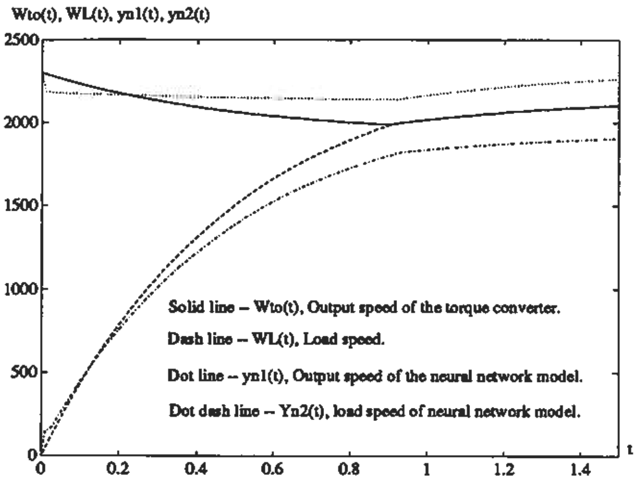

Figure 5: The identification model of the clutch-load subsystem. The speeds of the system and neural network outputs are shown after 100,000 iterations with random input training.

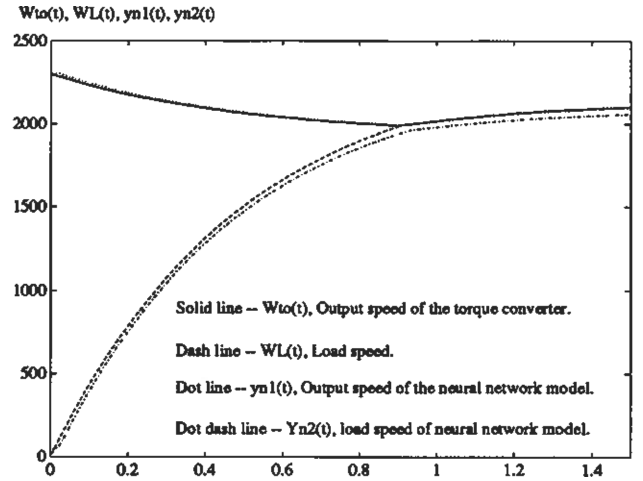

Figure 6: Identification model of the clutch-load subsystem. The speeds of the system and neural network outputs are shown after 490,000 iterations with random input training.

If the training goes on, it is expected that the error will improve further.

The simulations conducted above (also in [6] and else where) show the potential of neural network modeling of nonlinear dynamical system such as the powertrain clutch-load subsystem. It should be noted that when the physical model is available in the practical industrial applications, the off and on line methods discussed above should be used together.

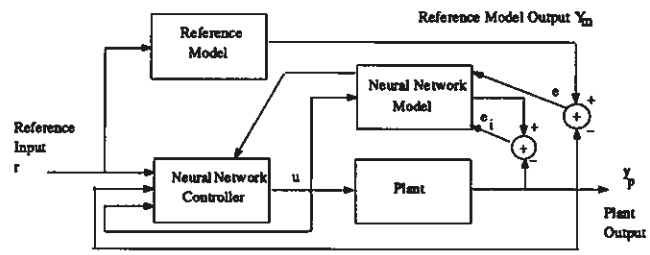

Figure 7: Indirect control configuration

3 Control of the Powertrain Subsystem using Neural Networks

The control structure used here is shown in Fig. 7. Other control configurations are of course possible; see [6] and [2] among others. In this structure, which is often called indirect control, the plant is first identified by a neural model. After identification, the weights of the neural plant are fixed. Given a reference signal r, the output of the plant and the output of the reference model are different at the beginning. The differences of the two outputs, the output error e, is back propagated to the neural controller though the neural plant. The purpose of the neural plant is to provide a path for the output error to reach the neural controller. The neural controller's weights are adjusted based on the output error. The basic algorithm used is back propagation. After training, the values for the weights of the neural controller have converged. The output of the plant should now track the output of the reference model well.

The structure and the size of neural networks, in terms of the number of hidden layers and the number of neurons on each hidden layers, are decided by experiments.

For this specific problem, to control the powertrain subsystem certain control strategy has to be chosen. To meet the design specifications we need to:

•Confine the maximum heat peak power below certain allowable level.

•Minimize the heat energy dissipated during the engagement period.

•Complete the engagement within 1.5 second.

The control objective is to design a neural network controller to control the clutch - load subsystem output ($\omega_{to}(t)$ and $\omega_L(t)$, so that these two outputs can follow the desired trajectories predesigned by engineers. The main controller input is the the error signal (The differences between the desired trajectories and the actual trajectories of the plant.) The controller output is the clamping force $F(t)$. To find the desired plant output trajectories under

Z. Hou and P. J. Antsaklis, "Identification and Control of a Powertrain System Using Neural Networks," P roc o f t he S ymposium o n I mplicit a nd N onlinear S ystems , Univ of Texas at Arlington, Ft. Worth, TX, Dec. 14-15, 1992.

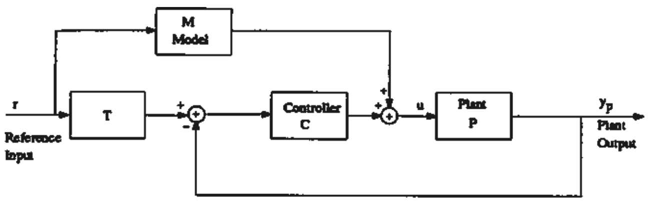

Figure 8: Configuration of the M type control

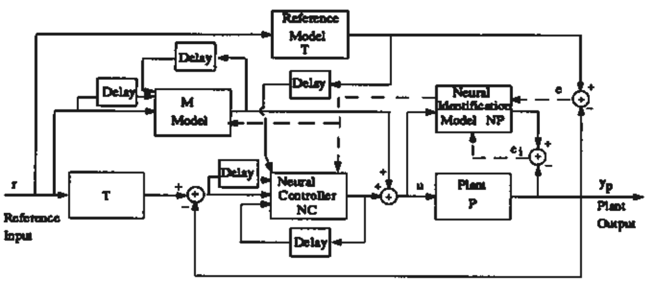

Figure 9: Configuration of the M type control via a Neural network

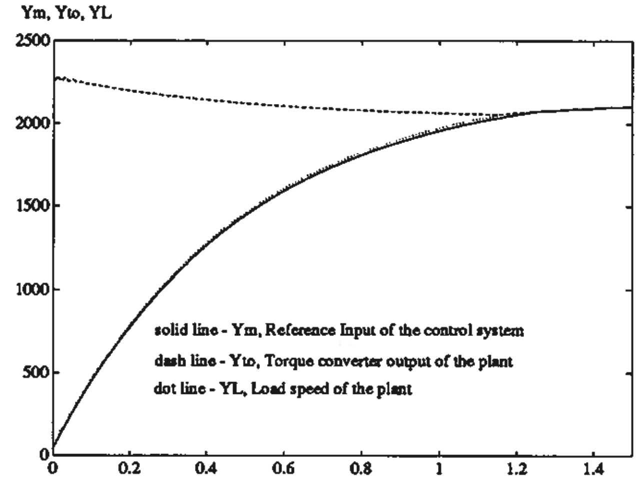

Figure 10: The controlled plant output, and the reference model output are shown after 200,000 iterations using random input training.

the given technical conditions is a problem of choosing an optimal set of parameters of the system inputs. (corresponding to open loop control.) An approach that can be used to solve this problem is described in Peek and Antsaklis [11]; other approaches are also possible. Note that the difficulties in determining the output trajectories that satisfy the constraints stem from the system nonlinearitis but mostly are due to uncertainties; so a procedure such as [11] may be more approprate here than optimal control methods. When the desired output trajectories are decided then neural network control system can be trained as a model following system. So any desired output trajectories fed as a reference inputs to the control system will lead to the powertrain system output to follow those trajectories. The configuration is shown in Fig. 8. In Fig. 8, controller C safeguards stability while M causes the response of the system to be T $(T = PM)$.

The neural controller system configuration is shown in Fig. 9. Here C is a neural network controller, M is a neural network auxiliary controller, T is a desired transfer function, and P is an unknown nonlinear plant.

The plant model, which is used to generate input output data, is the same as in equation (11).

The neural network controller is:

$$\hat{y}(k+1) = \hat{f}(u(k), u(k-1),$$

$$\cdots u(k-d), \hat{y}(k), \hat{y}(k-1), \cdots, \hat{y}(k-d)) \qquad (21)$$

$\hat{y}(k+1)$ is the neural network controller output and at k+1. $u(k)$ is the input to the neural network controller at time k. (The M-model has the same structure as the neural network controller.) Since there are two neural networks in the control system, the training procedure has to be carefully chosen. First the neural plant has to be trained with a random input signal. During the next step the controller is being trained with the weights of the neural plant model being fixed.

The neural networks for C and M are the same with two hidden layers (20,10) and two delays. The sizes and the delay are chosen based the experiments conducted in the previous work [6]. During controller training, C and M are trained simultaneously. The training step size is $\eta = 0.01$ and $\alpha = 0$. The reference input training signal is a random variable uniformly distributed within [0, 3600](Scale factors were used in the training process.). This reference input is used for the load speed of the powertrain system. Since the torque converter output speed is closely related to the load speed while the clamping force only indirectly affects the torque converter output speed, for reference input only choose one of the quantities. Here, the torque converter output torque is set constant at $T_{to} = 2.43$. During the simulation, the output error is back propagated through the neural plant to the controller C and M. After 200,000 iterations, the training process is stopped and the desired load speed curve is fed to the reference input. The result is shown in the Fig. 10. The reference input and the outputs of the control system trajectories are very close. The control action, clamping force F is shown in Fig. 11. The vibrations at the beginning of the clamping force may indicate that more training iterations should be conducted if this phenomena can not be tolerated.

The shape of the clamping force curve is in between the step input and the ramp type input which are commonly used in industry, derived mostly by empirical considerations. See Fig. 12. Several different curves have been used as the reference inputs and the system tracks well. See Fig. 13.

The controller configurations in this section have relaxed the constraint for the plants, (the delayed plant

Z. Hou and P. J. Antsaklis, "Identification and Control of a Powertrain System Using Neural Networks,"
P roc o f t he S ymposium o n I mplicit a nd N onlinear S ystems , Univ of Texas at Arlington, Ft. Worth, TX,
Dec. 14-15, 1992.

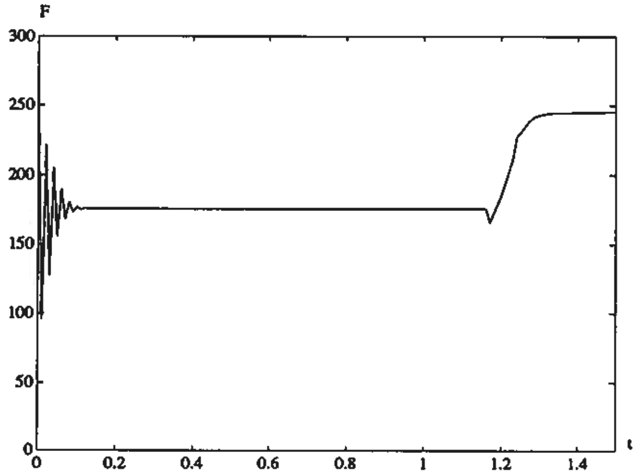

Figure 11: The clamping force curve

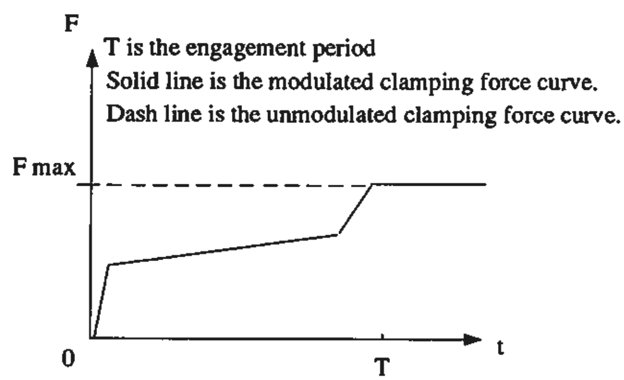

Figure 12: Currently used clamping force modulation curves.

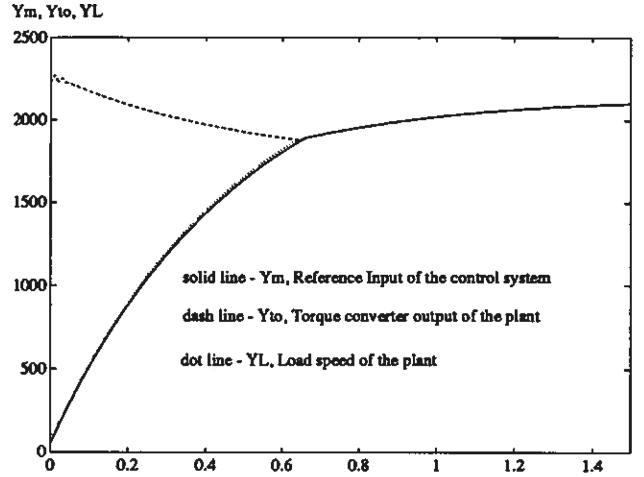

Figure 13: The controlled plant output, and the reference model output are shown after 200,000 iterations using random input training. Different reference input trajectory is used here from in Fig.10.

input terms and the delayed plant output terms must be separable as described in [13]), which can be controlled by this type of neural network controller. During training, the training step size η plays an important role. The experiments show that large values of η can make the whole control system unstable, while smaller values of η can lead to lengthy convergence times. The designer should watch the training process carefully.

From the simulations in this section, it may be noted that the training process is off line, thus, it is not possible to realize adaptive control in the classical sense. To accommodate parameter changes in an unknown plant, the whole training process must be implemented on line. This would require the plant identification and controller training to be performed simultaneously. This would greatly broaden the application of neural networks and is left for future research efforts.

4 Conclusion

Neural network models were used to model and control a nonlinear powertrain clutch-load subsystem. The results show that neural networks can be effective in this type of problems. Further simulations and testing on the actual vehicle need to be carried out for the few validation of the procedures.

References

[1] Antsaklis, P. J., Ed. Special Issues on "Neural Networks in Control Systems," *IEEE Control System Magazine*, April 1990 and April 1992.

Z. Hou and P. J. Antsaklis, "Identification and Control of a Powertrain System Using Neural Networks," P roc o f t he S ymposium o n I mplicit a nd N onlinear S ystems , Univ of Texas at Arlington, Ft. Worth, TX, Dec. 14-15, 1992.

[2] Antsaklis, P. J., Neural Networks for the Intelligent Control of High Autonomy Systems. Technical Report # 92-9-1 Dept. of Electrical Engineering, U. of Notre Dame, Sept. 1992.

[3] Cho, D., "Sliding Mode Control of Automotive Powertrains with Uncertain Actuator Models," Dept. of Mechanical and Aerospace Engineering, Princeton University, Princeton, NY

[4] Dundore, M. W. and Schreider, R. C., " Clutch Energy- A Criteria of Thermal Failure," SAE National Combined Farm Construction and Industrial Machinery, Power Plant, and Transportation Meetings Milwaukee, Wis. Sept. 9-12, 1968

[5] Hopfield, J. J., "Neural Networks and Physical Systems with Emergent Collective Computational abilities," *Proc. of the National Academy of Science*, U/ S. A., vol. 79, pp 2554-2558, April 1982

[6] Hou, Zugang, "Analysis of Auto Powertrain Dynamics and Modeling Using Neural Networks". Master Thesis, Dept. of Electrical Engineering, U. of Notre Dame, 1992

[7] Hrovat, D., Tobler, W.E. and Tsangarides, M.C., "Bond Graph Modeling of Dominant Dynamics of Automotive Power Trains," Research Ford Motor Company, Dearborn, Michigan.

[8] Jania, Z. J., "Friction-Clutch Transmissions," Automotive Research Dept. Ford Motor Co. Dearborn, Michigan.

[9] Johnson, R. W., " Computer Modeling of the heat transfer in a Powershif Transmission Clutch under Slippage," *Master Thesis*, Kansas State University, 1987

[10] Moskwa, J. and Hedrick, J., "Nonlinear Algorithms for Automotive Engine Control," *IEEE Control System Magazine*, vol. 10, pp 88-93, April 1990

[11] Peek, Mark D. and Antsaklis, Panos J., " Parameter Learning for Performance Adaptation," *IEEE Control System Magazine,*, pp3-11, Dec. 1990

[12] Moskwa, J. and Hedrick, J., "Automotive Engine Modeling for Real Time Control Application," *1987 American Control Conference*, vol. 1, pp. 341-346, 1987.

[13] Narendra, K., "Identification and Control of Dynamical systems Using Neural Networks," *IEEE Trans. Neural Networks*, vol. 1. NO. pp. 4-27, 1 March 1990

[14] Nguyen, H. D. and Widrow, B., "Neural Networks for Self-Learning Control Systems," *IEEE Control Systems Magazine*, vol. 4 pp. 18-23, April, 1990.

[15] Patrikar, A. and Provence, J., "Learning Control of Dynamical Systems using Neural Networks," *proc. of 28th Annual Allerton Conference on Communication, Control, and Computing*, pp.1-10. Oct. 3-5, 1990.

Dynamical Systems using Neural Networks," *proc. of 28th Annual Allerton Conference on Communication, Control, and Computing*, pp.1-10. Oct. 3-5, 1990.

[16] Rumelhart, D. E.; Hinton, G.J. and Williams, R., "Learning Internal Representations by Error Propagation," Rumelhart, D. E., McClelland J. L., eds, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol.1. Foundation, pp 318-362, MIT Press, 1986.

[17] Sartori, M., "Neural Networks in Control System," the fifth chapter, Ph.D. dissertation. U. of Notre Name, 1991.

[18] Werbos, P., "Backpropagation Through Time: What It Doed and How to Do it," *Proc. IEEE*, vol. 78, No. 10, pp. 1550-1560, 1990

[19] Wouters, M., "A New Method to Calculate Energy dissipation and Temperature Distribution in Multidisc Clutches in Powershift Transmissions," International Off-Highway & Power Plant Congress and Exposition, Milwaukee, Wisconsin, Sept. 9-12, 1991