ANN BASED Optimal Control of the dc-dc Converter

Suraj Bhardwaj Mtech Scholar Department of Electrical Engineering RSR Rungta College of Engineering & Technology Kurud Road, Kohka, Bhilai(C.G.),490024	Nanda R Mude Assistant Professor Department of Electrical Engineering RSR Rungta College of Engineering & Technology Kurud Road, Kohka, Bhilai(C.G.),490024

Abstract - Even while dc-dc converters are excellent power supply instruments, their nonlinear behavior means that changes to their primary parameters could compromise their stability. In order to regulate the dc-dc buck converters' output voltage, this study suggests using an Artificial Neural Network (ANN) approach. This paper offers a mathematical model and simulation of a nonlinear dc-dc buck converter. As a contrast example, we have the traditional PID controller of a dc-dc converter. A dc-dc buck converter is tested using MATLAB Simulink software under both no-load and full load situations to assess the effectiveness of the suggested method. The results demonstrated that the proposed neural controller outperformed the classic PID controller technique in terms of responsiveness and robustness, all without incurring a substantial additional expense.

.

Index Terms- Power electronics, Buck converter, PID control, neural network, modeling and analysis, Continuous conduction mode (CCM), discontinuous conduction mode (DCM), NARMA –L2

1 Introduction

To keep up with the ever-increasing demand, researchers are working on a number of innovative DC-DC converter topologies that are both highly efficient and easy to manage. Modeling and analysis, as well as enhancing both static and dynamic performance, are all part of this. As a result, dc-dc converters have evolved into useful instruments for supplying power and as regulators for various electronic systems, thanks to innovations in power electronics. There are a number of benefits to using a converter instead of a traditional linear regulator that divides voltage or current, which is wasteful because the output voltage can't be lower than the input voltage and has a low power density [1]. On the other hand, switching regulators are great at turning energy efficiently since they can operate at high frequencies with little losses and improve the converter's dynamic behavior [1]. The converters also allow for outputs that are greater than their inputs. Because the converter's characteristics impact its stability and nonlinear behavior, there are challenging processes to follow when addressing the design parameters in order to develop a mathematical model that represents the converter. Control applications involving dc-dc converters frequently make use of general-purpose PI and PID controllers. However, when the control parameters, loading circumstances, and the dc-dc buck converter itself are altered, it fails to produce satisfactory results. The precise system model is not necessary for the creation of artificial neural networks (ANN). Even if the system parameters are changed, this ANN design approach ensures stable operation. It is enough to comprehend the system's general behavior for ANN. For dc-to-dc converters, for example, ANN was developed and studied.

system model; ANNs can be used to identify parameters in system modeling, which is a control scheme requirement [2]. Cellular systems that can learn from experience, retain it, and use it later on are known as artificial neural systems [3]. The next step was to research and develop the artificial neural network (ANN) controllers that are recommended in this paper; these controllers are the best option for getting an accurate plant model. After comparing the outcomes of the various tactics, one was selected for actual implementation. One major advantage of artificial neural networks (ANN) is their versatility; they can handle both linear and non-linear functions [4]. The outcomes are displayed in computer models, which reveal distinct traits and reactions from every controller.

2 DC - DC Buck Converter Principles

By means of a switching mechanism, DC-DC converters transform one electrical voltage level into another. There are two separate modes of operation for the dc-dc Buck converter: continuous conduction mode (CCM) and discontinuous conduction mode (DCM). Both modes, with vastly different features, can be used by a converter in practice. This means that both types of operation should inform the design of the converter and its control. But in this study, only dc-dc converters that are run in CCM are taken into account.

Circuit Operation of figure 1: When the switch (d) is ON fora time duration (d*T)

Where: d = switching status (0 for OFF statue or 1 for ON statue).

T = duration of operation of switching device (TON orTOFF). The switch conducts the inductor current and the diode becomes reverse biased. This results in a positive voltage VL = Vg – Vo across the inductor.

Where: VL is inductor voltage.Vg is dc input voltage. Vo is output voltage.

This voltage causes a linear increase in the inductor current(i_L). When the switch is turned OFF, $i_L$because of the inductive energy storage, continues to flow. This current now flows through the diode, and VL = -Vo for a time duration (1-d)*T until the switch is turned on again. This converter gives an output voltage v0 smaller than the inputvoltage vg.

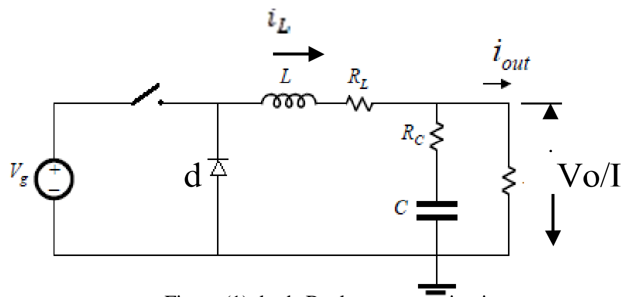

Figure (1) dc-dc Buck converter circuit

3 Mathematical Modeling and Simulation of aDC-DC Buck Converter

As RL and Rc make up the model. In other words, the input and output voltages in these topologies are not separate. But these non-isolated topologies have isolated derivations. The topology of the power supply describes the connections between the switches, output inductor, and output capacitor. The characteristics of each topology are distinct. The input and output current types, the steady-state voltage conversion ratios, and the output voltage ripple character are all examples of such features. The frequency responsiveness of the duty-cycle-to-output-voltage transfer function is another significant feature. The buck power stage, also known as a step-down power stage, is the simplest and most prevalent power stage topology. [5]

By using Kirchhoff's Voltage Law (KVL) and Kirchhoff's Current Law (KCL), System equations are obtained as shown below and these laws can be applied on the other dc – dc converter (boost, buck-boost and cuk) converters.

$$\frac{di_L}{dt} = \frac{1}{L}\left(V_g \cdot d - i_L R_L - v_o\right) \quad \ldots\ldots\ (1)$$

$$\frac{dv_C}{dt} = \frac{1}{C}\left(i_I - i_{out}\right) \quad \ldots\ldots\ldots\ldots\ (2)$$

$$v_o = v_c + R_c\ (i_L - i_{out}) \quad \ldots\ldots\ldots\ (3)$$

Where: $i_{out} = \frac{V_o}{R}$

The open loop dc-dc buck converter with simulation is shown in figure (1). Parameters used in the simulation studies are given below:

Vg = 12 volt, L = 1 µH , RL = 80 mΩ , C = 376 µF , Rc = 5 mΩ , d = 1 (duty cycle) , R = 28 Ω (load). [5]

PID Controller

One common feedback mechanism in industrial control systems is the proportional-integral-derivative (PID) controller. By calculating and then releasing a corrective action that can change the process as needed, a PID controller aims to fix the discrepancy between a measured process variable and a desired set point. Three distinct parameters—Proportional, Integral, and Derivative values—are used in the PID controller calculation procedure. The response to the present error is determined by the Proportional value, the response to the sum of the errors that have occurred recently is determined by the Integral, and the response to the rate at which the error has been changing is determined by the Derivative. A control element, like the location of a control valve or the power source of a heating element, is adjusted by weighing the total of these three steps. A PID controller can tailor its control action to meet the needs of a particular process by adjusting the three constants that make up its algorithm. One way to characterize the controller's behavior is by looking at how it handles errors, how much it deviates from the setpoint, and how much oscillation there is in the system. It should be noted that the PID algorithm is not a guarantee of optimal control when used for system control. To put it another way, first estimation is like a proportionate PID controller's behavior, but PID controllers cannot be taught and necessitate an appropriate setup. Tuning the controller means choosing the right gains for efficient control. [5] Trial and error has long been the method of choice for determining these parameters.

When the controller's performance is heavily dependent on the design engineers' expertise, manual tuning of the PID controller is an arduous, time-consuming, and hard process to do. [6]

.

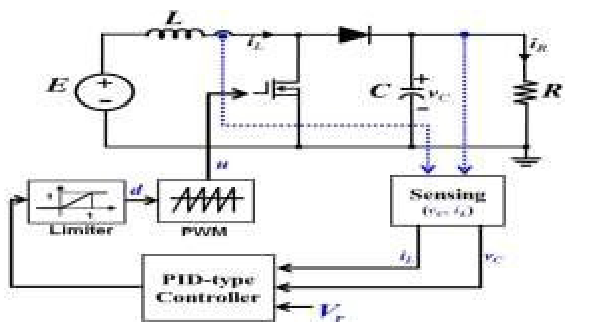

Figure 2: dc-dc buck converter with PID

The modeling of a dc-dc buck converter with PID controlleris shown in figure (2). The simulation model for a dc-dc buck converter for load change with PID controller is shown in the figure (5). By setting the proportional gain Kp to 8, Ki to 280, and Kd to 0.001. These parameters are determined by a trial and error approach.

4 NEURAL CONTROLLER

The model of an artificial neuron that closely matches a biological neuron is given by an op-amp summer like configuration shown in figure (3).

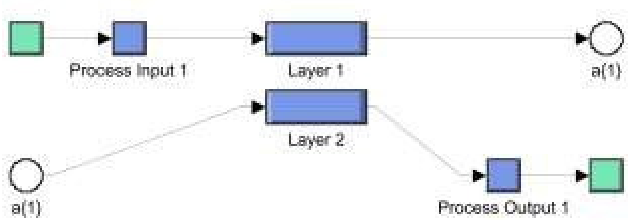

Figure 3: ANN Network

Where x1, x2, x3… are input signals, each of the input signal flows through a gain called synaptic weight. The weight can be positive (excitory) or negative (Inhibitory) corresponding, respectively, to acceleration or inhibition [7].

The summing nodes accumulate all the input weighted signals and then pass to the output through the transfer function which is usually nonlinear. The transfer function can be step or threshold type, signum type, or linear threshold type. The transfer function can also be nonlinear continuously varying type, such as sigmoid, inverse-tan, hyperbolic, or Gaussian type. The sigmoidal transfer function is most commonly used, and it is given by

$$Y = \frac{1}{1+e^{-\alpha x}} \quad \ldots\ldots\ldots\ldots (4)$$

Where α is the coefficient or gain which adjusts the slope of the function. With high gain, this function approaches a step function. The sigmoidal function is nonlinear, monotonic, differentiable, and has the largest incremental gain at zero signal, and these properties are of particular interest.

In general, neural networks can be classified as feedforwardand feedback types depending on the interconnection of theneurons.

At present, the majority of the problems use feedforward architecture, and it is of direct relevance to power electronics and motion control applications.

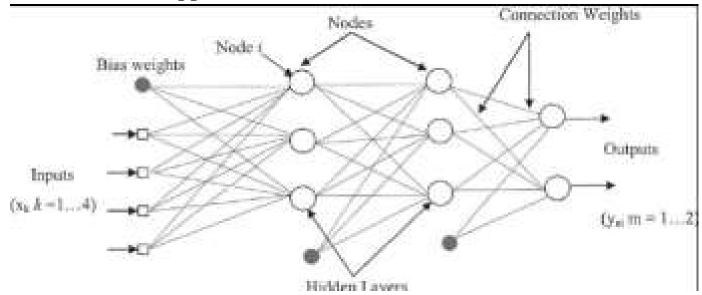

Figure 4: feedforward multiplayernetwork

A feedforward multiplayer network with two input signals and two output signals is illustrated in Figure (4). The Perceptron, first postulated by Rosenblatt in 1958, forms the basis of the topology. Each dot in a link stands for a weight, while the circles indicate neurons. A, the input layer, b, the hidden layer, and c, the output layer, make up the network's three layers. Connecting the input and output layers is the hidden layer's job. Each layer's neuron count is proportional to the quantity of impulses it receives or sends. Although the neurons in the input layer lack transfer functions, the input signals are normalized using scale factors, as demonstrated. When designing a network, it is important to take into account the number of hidden layers and the amount of neurons in each layer. As demonstrated, signals are transmitted from the input layer to the concealed layer, which subsequently transmits them to the output layer. The network may or may not have complete connectivity.

4-1 Back Propagation Training

Back-Propagation training algorithm is most commonly used in a feedforward neural networks as mentioned before. For this reason, a feedforward network is often defined as back-prop network.

In the beginning, the network is assigned random positive and negative weights. For a given input signal pattern, step by step calculations are made in the forward direction to derive the output pattern. A cost functional given by the squared difference between the net output and the desired net output for the set of input patterns is generated and this is minimized by gradient descent method altering the weights one at time starting from the output layer. The equations for the output of a single processing unit aregiven as:

$$Net^P_J = \sum_{i=1}^{N} WijXi \quad \ldots..(5)$$

$$Y^P_J = fj(Net^P_J) \quad \ldots..(6)$$

Where j is the processing unit under consideration, p is the input pattern number is the output of the i^{th} neuron connected to the j^{th} neuron, is the connection weight between the d_j^p i^{th} and j^{th} neurons. is the output of the summing node, i.e., the j^{th} neuron activation signal, N is the number of the neurons feeding the j^{th} neuron, f j is the nonlinear differentiable transfer function (usually sigmoid), and Y_j^p is the output of the corresponding neuron.For the input pattern p, the squared output error for all theoutput layer neurons of the network is given asWhere is the desired output of the j^{th} neuron in theoutput layer y_j^p, is the corresponding actual output, S is thedimension of the output vector y^p is the actual net output vector, and d^p is the corresponding desired output vector. The total squared error E for the set of P patterns is then given by

$$E_p = \frac{1}{2}(d^p - y^p)^2 = \frac{1}{2}\sum_{j=1}^{5}\left(d_j^p - y_j^p\right)^{2!} \quad \text{....(7)}$$

$$E = \frac{1}{2}\sum_{p=1}^{P} E_p = \frac{1}{2}\sum_{p=1}^{P}\sum_{j=1}^{s}\left(d_j^p - y_j^p\right)^{2!} \quad \text{... (8)}$$

The weights are changed to reduce the cost functional E in a minimum value by gradient descent method, as mentioned. The weight update equation is then given as:

$$W_{ij}(t+1) = W_{ij}(t)^{\eta}\left[\frac{\delta E_p}{\delta W_{ij}(t)}\right] \quad \text{..... (9)}$$

Where η is the learning rate, Wi j (t +1) is the new weight and Wi j (t) is the old weight. The weights are updated for all the P training patterns. Sufficient learning is achieved when the total error E summed over the patterns falls below a prescribed threshold value. The iterative process propagates the error back-propagation [7, 8, 9].

Using this equation directly can cause realization problems, because must determine the control input based on theoutput at the same time, i.e:

y(k+d)=f[y(k),y(k–1),..,y(k n+1),u(k),u(k–1),…,u(k–n+1)]+ g[y(k),…,y(k–n+1),u(k),…,u(k–n+1)] u(k+1) …… (10)

5-2 NARMA – L2 NEURAL CONTROLLER

In this work, the NARMA –L2 architecture is applied with the aid of the Neural Network Toolbox of MATLAB software. The identification can be summarized by the flowing steps:

a- The first step in using feedback linearization (or NARMA-L2 control) is to identify the system to be controlled.

Neural network is trained to represent the forward dynamics of the system. One standard model that has been used to represent general discrete-time nonlinear systems is the NARMA-L2 model [10]:

y(k +d) = N[y(k),y(k – 1),…, y(k –n+ 1), u(k),u(k– 1),…,u(k–n+ 1)] ……(15)

where u(k)is the system input, and y(k)is the system output and k ,d, n are integral number and N is the function of the output system after identification.

y^(k+d)=f[y(k),y(k–1),…,y(k–n+1),u(k–1),…,u(k–m+1)]+ g[y(k),y(k –1),…,y(k–n+1),u(k– 1) ,…,u(k–m+ 1)]u(k) ….. (11)

Where the next controller input is not contained inside the nonlinearity. The advantage of this form is that controlled input make the system output follows the reference equation (3). The resulting controller is:

$$u(k) = \frac{y_r(k+1) - f[y(k),y(k-1),..y(k-n+1)..u(k-n+1)]}{g[y(k),..y(k-n+1),u(k-n+1)]} \quad \text{.......... (12)}$$

Figure (11) is referred to block diagram of the proposed dc-dc buck converter with NARMA-L2 controller.

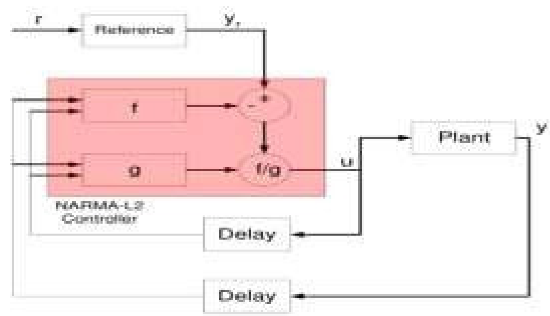

Figure 5: dc-dc buck converter with NARMA-L2 controller

5 Simulation Result

The response for output voltage and output current of

b- The next step is to make the output system follows some reference trajectory

by developing a nonlinear controller of the form:

y(k +d) = yr(k +d) …… (14)

u(k) = G[y(k),y(k – 1),…, y(k –n+ 1), yr(k +d),u(k– 1),…,u(k–m+ 1)] ….. (12)

The problem with using this controller is:

Training neural network to minimize mean square error needs to use dynamic back propagation which quite slows [11].

One solution is to use approximate models to represent the system. The controller used in this section is based on the NARMA-L2 approximate model:

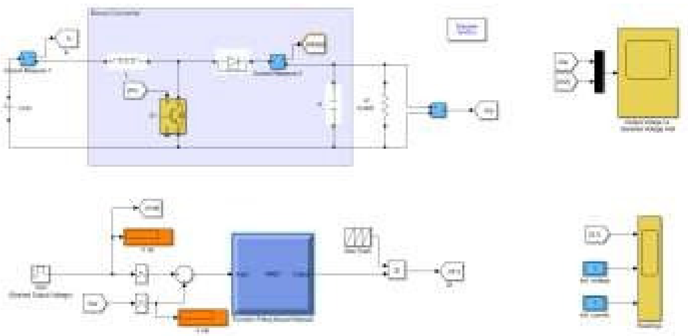

Figure 6: Modeling Of proposed system

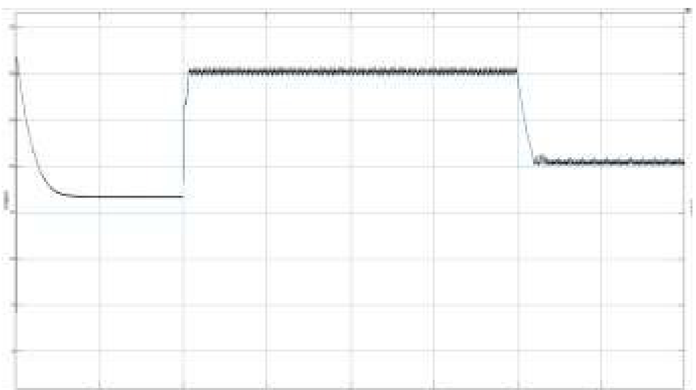

Figure 7: Vref and output Voltage

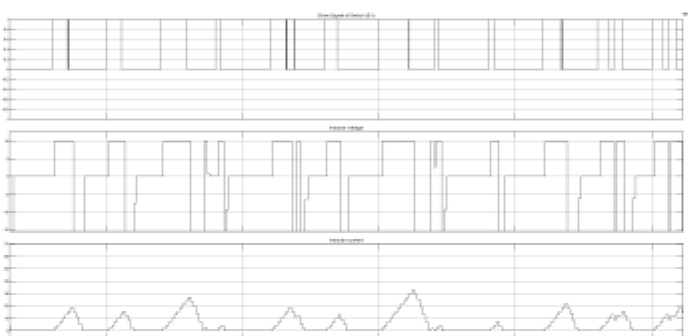

Figure 8: signal Graph of Ann network

6 Conclusion

Regardless of changes in Vg(t) and ioad(t) as well as changes in the values of the elements making up the converter circuit, it is ideal for a DC-DC converter to produce an output voltage Vo(t)=Vo that remains constant. An investigation into a neural network-controlled DC-DC voltage static converter has been conducted. When it comes to stability, the simulation findings are good. A more dynamic and responsive regulatory system is demonstrably possible. Differences between the neural network and PID correctors in terms of adjustment visibility are more pronounced. With the help of the suggested method, we can see that the ANN outperforms the PID. The system may be operated to its stability limit during steady state and still remain stable after severe disruptions thanks to the increased damping performance of the neurocontrollers. Additionally, the neural network outperforms the PID controller in terms of processing speed, resulting in a quicker rise time, elimination of overshoot, and elimination of steady state error.

7 References

[1] Muhammad, R., Power Electronics handbook, Academic Press Series in Engineering, 2001, pp. 211.

[2] Anas N. Al-Rabadi and Othman M.K. Alsmadi, Model Reduction-Based Control of the Buck Converter Using Linear Matrix Inequality and Neural Networks, Proceedings of the International MultiConference of Engineers and Computer Scientists , Vol II IMECS 2009,March 18 - 20, 2009, Hong Kong.

[3] J. M. Zurada, Artificial Neural Systems, West Publishing Company, New York, 1992.

[4] J-N. Marie-Francoise and H. Gualous, A. Berthon, dc to dc Converter with Neural Network Control for ON-Board Electrical Energy Management, UTBM-L2ES, University Franche-Comte rue Thierry MIEG, F90010 Belfort, France.

[5] Muhamad Farhan Bin Umar Baki, Modeling and Control of dc to dc Converter (Buck) ,Faculty of Electrical & Electronic Engineering University Malaysia Pahang (UMP), MAY- 2008.

[6] Ziegler, J.G., and Nichols, Optimum settlings for automatic controllers, Trans. On ASME., vol. 64, pp.759- 768 ,N.B. 1942.

[7] B. K. Bose, Expert System, Fuzzy logic, and neural network Applications in power Electronics and motion control, Proceedings of the IEEE, Vol. 82, NO. 8, 1303-1321. World Academy of Science, Engineering and Technology 462008 899 August 1994.

[8] A. G. Aissaoui, M. Abid, H. Abid, And A. Tahour, A Neural Controller for Synchronous Machine ,World Academy of Science, Engineering and Technology 46,pp 893- 900,2008.

[9] M. Chow, R. N. Sharpe and J. C. Hung, On the application and design of artificial neural networks for motor fault detection – Part I, IEEE Transaction on Industry Electronics, Vol. 40, NO. 2, 181- 188,April 1993.

[10] Becerra ,V. M., Process Control Simulation using Simulink, Sept. 2000, Updated for Matlab 6.5 – June,2003.

[11] Mehpotra ,K. and Mohan ,C. K., Elements of Artificial Neural Networks, proceeding Japan text book pp.41 ,2001.