CHAPTER 4

INTELLIGENT CONTROLLERS FOR SEPOSCLC

Voltage stability plays a crucial role in the performance quality of DC-DC converters. It has been seen in the previous chapter that the performance of boost converters using conventional PI and PID controllers was sluggish. So it would be necessary to use such devices that would ensure voltage stability.

In 1965, LotfiZadeh formalized the Fuzzy set theory [Zadeh 1965] and in 1973 brought it into the context of control systems [Zadeh 1973]. According to LotfiZadeh, Fuzzy logic control systems are ‘higher machine intelligence quotient’ [Cox 1992]. Artificial Neural Network (ANN) is generally acknowledged as an innovation offering an options approach to solve intricate and poorly characterized problems. It is a gross generalization of real biological networks and copies fairly the learning procedure of a human mind. It is capable of handling complex and large system with man-given interrelated parameters. It seems to ignore, abridge, surplus input data that is of importance and concentrate on the more significant input. A trained ANN can be used to approximate an arbitrary input-output mapping of the system.

Considering the advantages of Fuzzy logic and ANN, both have been used in this research, with the addition of sliding controller to ANN in order to improve the time response specifications and to regulate the DC output voltage of the SEPOSCLC converter.

4.1 Fuzzy Logic Controller

Fuzzy logic is based on a mathematical model which deals with uncertainty and imprecise, ambiguous, noisy, or missing input information. It
compares an analog input signal to a predetermined logic variable membership function or fuzzy sets that correspond to values in the range 0 and 1. Fuzzy logic is implemented in a control system to emulate human-like decisions in control. FL techniques have been widely used in a wide range of engineering applications because of their heuristic nature associated with simplicity and effectiveness for both linear and nonlinear systems [Veerachary, M. 2002][ Diaz, N.L. and Soriano, 2007][ El Khateb et.al 2014] [ So, W.C et.al, 1996]. It can be implemented in software, hardware or a combination of both. Fuzzy sets are considered as the brain of the fuzzy control system which in turn is responsible for converting analog input values into a scale of 0 to 1.

Over the past few decades, fuzzy logic controllers have been successfully used in many industrial applications.

Some of the essential characteristics of FL are listed below [R. J. Machado 1992]:

- Exact reasoning is viewed as a limiting case of approximate reasoning.
- Everything is a function of degree.
- Knowledge has inferred a compilation of elastic or, equivalently, the fuzzy constraint on a compilation of variables.
- The interface is viewed as a process of propagation of elastic constraints.
- Any logic system can be fuzzified.

There are two main characteristics of fuzzy systems that enable them to perform better in specific applications.
Fuzzy systems are appropriate for vague or rough reasoning, particularly for a system with a mathematical model that is difficult to derive.

Fuzzy logic allows decision making with estimated values under incomplete or uncertain information.

### 4.1.1 Fuzzy Logic Controller Structure

The general structure of a fuzzy controller with its main parts is shown in Figure 4.1. The fuzzy logic controller consists of four components: fuzzification interface, knowledge base, interface engine, and defuzzification interface [Y. Bai 2007].

![General structure of fuzzy interface system](image)

**Figure 4.1** General structure of fuzzy interface system [Y. Bai 2007].

The fuzzifier transforms crisp inputs into the fuzzy subsets to obtain degrees of memberships and represents them with linguistic variables, which in turn are used to activate rules. [Guo, L et.al, 2009] [Gupta et.al, 1997]. The fuzzy inference engine process those data to obtain the desired output. The defuzzifier converts those fuzzy outputs to crisp variables to complete the desired fuzzy logic control objectives. Every fuzzy set can be represented by its membership function. In practice, the membership functions can have a number of different shapes depending on the application. They can be triangular, trapezoidal, Gaussian, bell-shaped, sigmoidal or S-curve. Some
popular waveforms of the membership functions are shown in Figure 4.2 [Y. Bai 2007, M. Mobaied 2008].

Figure 4.2 Different graphs of membership functions (a) monotonic (b) trapezoidal (c) Gaussian. (d) Triangular [M. Mobaied 2008].

Trapezoidal or triangular membership functions are usually used in systems that require significant dynamic variation in a short period of time. A Gaussian or S-curve membership function is always selected if the system requires very high control accuracy [Y. Bai 2007].

4.1.2 Fuzzy Interface Procedure

There are several techniques for performing the fuzzy interface process, such as the Mamdani method, Takagi-Sugeno_Kang (TSK) method, Tsukamot method, and Larsen method. The Mamdani and TSK methods are the most commonly used methods in fuzzy controllers [Y. Bai 2007, M. Mobaied 2008]. However, the Mamdani method is usually more popular for most control engineering applications because this method is computationally more efficient and has better interpolative properties than the other interface methods [N. S. D'Souza 2006].
4.1.2.1 **Mamdani method**

The Mamdani method is the most widely used fuzzy interface strategy. It was proposed in 1975, by Professor Ebrahim Mamdani of London University who built one of the fuzzy systems to control a combination of steam engine and boiler. In Mamdani’s method, a set of fuzzy if-then rules was applied, and these were supplied by experienced human operators [I. Iancu 2012].

Mamdani fuzzy logic uses the linguistic variables to describe the process states and uses these variables as inputs to control rules [Viswanathan, K et.al, 2005]. The terms of the linguistic variables are fuzzy sets with a certain shape. Mamdani fuzzy logic usually uses the trapezoidal or triangular fuzzy set, but other shapes are possible. The fuzzy interface process of the Mamdani method can be formed in four steps [I. Iancu 2012]:

- Fuzzification
- Rule evaluation
- Aggregation of the rule outputs
- Defuzzification

To clarify the fuzzy interface working process of the Mamdani method, a simple two-input one-output problem that includes three rules is examined:

Rule 1: **IF X is A3 OR Y is B1 THEN z is C1**

Rule 2: **IF X is A2 AND Y is B2 THEN z is C2**

Rule 3: **IF X is A1 THEN z is C3**
Step 1: Fuzzification

Fuzzification is the first step in fuzzy logic processing, in which the crisp quantities are converted to fuzzy inputs. The transformation process of fuzzy values is represented by the membership function [M.Mobaied 2008, I. Iancu 2012]. There might be some error, when measuring the voltage, current, temperature, solar insolation, etc., in most real applications. This will result in vague data. Therefore, the vagueness can be represented by the membership functions. Figure 4.3 shows the fuzzification stage of the example. Firstly, take the crisp inputs, $x_1$ and $y_1$, and determine the degree to which these inputs belong to each of the appropriate fuzzy sets.

Step 2: Rule evaluation

In this step, the fuzzified inputs are applied to the antecedents of the fuzzy rule, then applying the fuzzy operator to fuzzy rule that has multiple antecedents to resolve the antecedents to a single number between 0 and 1. The fuzzy operator (AND or OR) is used to obtain this signal number. In order to evaluate the conjunction of the rule antecedents, the AND fuzzy operation intersection is applied.

Step 3: Aggregation of the rule outputs

After finding the output of each rule, the aggregation process is applied to unify all the rules outputs to become one output. The input to the aggregation process is the combination of truncated or scaled consequent membership function, and the output is the combined output fuzzy set.

Step 4: Defuzzification

The Defuzzification is the last step in the process of the fuzzy inference, and its reverse of the fuzzification process. The output of the fuzzy
inference process result from the combination of the control inputs, and it is still linguistic variable. However, the final output of the fuzzy control has to be a crisp number. Hence the defuzzification process is needed to convert the fuzzy output back to the crisp output to be available for real application controllers [Y. Bai 2006, M. S. El-Moghany 2011].

There are several defuzzification techniques which are used, but there is one systemic procedure for selecting a good defuzzification strategy. Selection of the defuzzification technique relies on the system or the application characteristics [C.-C. Lee 1990].

The following three defuzzification techniques are the most commonly used techniques:

- Mean of maximum method (MOM)
- Height method (HM)
- Centre of gravity method (COG)

The center of gravity method is the most widely used defuzzification technique in real applications. The advantage of COG method is the defuzzified values tend to move smoothly around the output fuzzy region [Y. Bai 2006]. The COG technique is a basic general defuzzification technique that computes the value of the abscissa of the center of gravity of the area below the membership. [Perry, A.G e.al, 2006][Ofoli, A.R. and Rubaai, A, 2006][Mattavelli, P et.al, 1995][Guesmi, K et.al, 2008]. Hence in this analysis center of gravity method is proposed.

4.1.3 **Fuzzy Logic Controller in Converter proposed in this research**

Duty ratio for the switch in the converter is controlled by the fuzzy logic controller for constant DC voltage control. The difference between the
reference voltage ($V^*$) and actual voltage ($V$) is stated as voltage error which is fed as one of the inputs to the Fuzzy controller. Change in error is fed another input to the FLC. There are several ways to define the result of a rule; this research work implies a max-min method of inference. Mamdani type fuzzy is proposed with the two inputs such as voltage error ($e$) and change in error ($e_c$) to produces $D$ as output.

$$E = V^* - V$$ \hspace{1cm} (4.1)

Both inputs and outputs variables have five degrees such as NB-negative big, NS-negative small, Z-zero, PS-Positive Small, and PB-Positive Big [Koprda, S 2015]. Defuzzification is the mathematical procedure to convert fuzzy values into crisp values. Many methods of defuzzification are available. In this study, centroid method of defuzzification is selected [M. Hassan 2012]. Table 4.1 shows the fuzzy rules. Figure 4.3 & Figure 4.4 shows the input membership functions & Output membership functions respectively.
Figure 4.4 Output membership functions

Table 4.1 Fuzzy Rules

<table>
<thead>
<tr>
<th>Change in error</th>
<th>NB</th>
<th>NS</th>
<th>Z</th>
<th>PS</th>
<th>PB</th>
</tr>
</thead>
<tbody>
<tr>
<td>NB</td>
<td>NB</td>
<td>NS</td>
<td>NS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NS</td>
<td>NB</td>
<td>NS</td>
<td>NS</td>
<td>PS</td>
<td>PB</td>
</tr>
<tr>
<td>Z</td>
<td>NB</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
<td>PB</td>
</tr>
<tr>
<td>PS</td>
<td>NB</td>
<td>NS</td>
<td>PS</td>
<td>PS</td>
<td>PB</td>
</tr>
<tr>
<td>PB</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
<td>PS</td>
<td>PB</td>
</tr>
</tbody>
</table>

The Simulation model of Fuzzy logic controller is shown in Figure 4.5.

Figure 4.5 Simulation model of FLC controller
Figure 4.5 shows the simulation model of FLC for SEPOSCLC control. Two inputs error and change in error are given to FLC to produce \( D \).

4.2 Simulation results

In this analysis, SEPOSCLC converter is analyzed with constant and variable input voltage in the range of 11V to 16V. For a constant input voltage analysis, input voltage is kept as 11V, 12V, 13V, 14V, 15V and 16V. The performance of FLC controlled SEPOSCLC converter at 12V constant input is shown in Figure 4.6.

![Figure 4.6 Output voltage of FLC controlled SEPOSCLC](image)

![Figure 4.7 Capacitor voltage of FLC controlled SEPOSCLC](image)
From the Figure 4.6 it is noted that output voltage oscillates at the time of starting and settles after 0.03sec. Minimisation of capacitor voltage ripple can be noted from Figure 4.7. It eliminates overshoot at the time of starting. For a variable input voltage analysis, the input voltage is varied from 11V to 13V, 14V, 15V and 16V. The performance of FLC controlled SEPOSCLC converter for variable input at 11V to 13V is shown in Figure 4.8.

![Variable input Voltage for FLC controlled SEPOSCLC converter](image1)

**Figure 4.8** Variable input Voltage for FLC controlled SEPOSCLC converter

![Output Voltage of FLC controlled SEPOSCLC converter for variable input of 11V to 13V](image2)

**Figure 4.9** Output Voltage of FLC controlled SEPOSCLC converter for variable input of 11V to 13V

From the Figure 4.8 & Figure 4.9, it is noted that the output voltage is maintained is at 72V with the help of FLC controller even when there is a change in input voltage. During the transition of the input voltage at a time of 0.05sec output voltage is oscillated for the period of 0.05sec to 0.06sec and settles after that.
The performance of FLC controlled SEPOSCLC converter for variable input at 14V to 12V is shown in Figure 4.10.

**Figure 4.10** Variable input Voltage for FLC controlled SEPOSCLC converter

From the Figure 4.10 & Figure 4.11, it is noted that the output voltage is maintained is at 72V with the help of FLC controller even when there is a change in input voltage. During the transition of the input voltage at a time of 0.05sec output voltage is oscillated for the period of 0.05sec to 0.06sec and settles after that.

**Figure 4.11** Output Voltage of FLC controlled SEPOSCLC converter for variable input of 14V to 12V
The performance of SEPOSLC converter using a FLC controller in the aspects of peak overshoot, output voltage ripple and settling time are presented in Table 4.2.

**Table 4.2** Performance of SEPOSLC converter using FLC controller

<table>
<thead>
<tr>
<th>Input voltage (Volt)</th>
<th>Peak overshoot (%)</th>
<th>Output Voltage ripple (%)</th>
<th>Settling time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>0</td>
<td>3.89</td>
<td>0.02945</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>3.88</td>
<td>0.02935</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>3.81</td>
<td>0.0218</td>
</tr>
<tr>
<td>14</td>
<td>0</td>
<td>4.4</td>
<td>0.0183</td>
</tr>
<tr>
<td>15</td>
<td>0</td>
<td>3.125</td>
<td>0.01635</td>
</tr>
<tr>
<td>16</td>
<td>0</td>
<td>3.6</td>
<td>0.0153</td>
</tr>
</tbody>
</table>

From the Table 4.2, it is noted that voltage ripple and settling time reduces when the input voltage increases. Peak overshoot is eliminated for all input voltages even for change in input voltage there is no overshoot. The graphical representation of SEPOSLC with FLC is shown in Figure 4.12 to Figure 4.14. Comparison of SEPOSLC with PI and FLC controller is shown in Figure 4.15 to Figure 4.17.

![Figure 4.12 Peak overshoot](image-url)
Figure 4.13 Voltage ripple

Figure 4.14 Settling time
Figure 4.15 Peak overshoot

Figure 4.16 Voltage ripple
4.3 Artificial Neural Network

ANN is an artificial network that imitates the human biological neural networks performance, extensively used in modelling multifaceted relationships between inputs and outputs in nonlinear systems [Eskander, M.N., 2002, Gnanasaravananan, A. and Rajaram, M., 2012, Jafari, M. and Malekjamshidi, 2011]. ANN can be stated as parallel distributed information executing system containing inputs and as a minimum one hidden layer and one output layer. These layers have executing elements named neurons interrelated together (see Figure 4.18).
Figure 4.18 General Structure of a four inputs ANN [Bishop, C. M. 1995]

The main advantages of using the artificial neural network ANN controller are:

1. A neural network can complete tasks that a linear program cannot i.e. awfully appropriate for non-linear systems.
2. When an element of the neural network fails, it can continue without any problem by their parallel nature.
3. A neural network learns and does not need to be reprogrammed.
4. It can be applied in different applications.

The main drawbacks of implementing the artificial neural network ANN controller are:

1. The neural network needs the training to operate.
2. The architecture of a neural network is sometimes complicated in design.
3. Requires high processing time for large neural networks.
An ANN is developed, with the one or more inputs which decide the output of the task. [Langer, N et.al, 2014][ Lin, B.R 1995][ Luo, X et.al, 2000]. The point by point ANN structure and information will be discussed in subtle elements in the accompanying segments.

4.3.1 Collecting Data

In designing of an ANN is the first step to collect historical data on the problem that is being solved using the network. In case of DC-DC converter control reference voltage and actual voltage and their corresponding duty ratio are required to in order to train the network. [Malekjamshidi, Z et.al, 2011]. This attained data is stated as training points.

4.3.2 Selecting Network Structure

As stated before, neural networks contain at least two layers (one hidden layer and another output layer). The input data is given to the hidden layers via weighted connections where the output data is estimated. The performance of the network is controlled by the number of hidden layers and the number of neurons in each layer.

As of not long ago, there are no rules for choosing an approach to pick the quantity of neurons alongside the number of shrouded layers for an offered issue to give the best execution. And it is still a trial and error plan system.

The ANN developed in this thesis has two inputs with two layers (one hidden layer and one output layer) the hidden layer has ten neurons with tan-sigmoid activation function, and the output layer has only one neuron with pure linear activation function which is the duty ratio. Figure 4.19 shows the structure discussed below.
An input vector \( x = (x_1, x_2, x_3 \ldots x_n)^t \) is applied to the input layer of the network. The net input of the hidden ‘j’ unit is:

\[
net^h_j = \sum_{i=1}^{n} W_{ji}x_i + \theta^h_j
\]  
(4.2)
Where $W_{ji}^h$ is the weight on the connection from $i^{th}$ input unit $\theta_j^h$ for $j = 1,2,...,N_h$ represents the bias for hidden layer neurons. Now output of the neurons in the hidden layer may be written as

$$Y_j^h = f \left( \sum_{i=1}^{n} W_{ji} x_i + \theta_j^h \right) \quad (4.3)$$

And the net input to the neurons in the output layer becomes

$$net_k^0 = \sum_{j=1}^{N_h} W_{kj} y_j^h + \theta_k^0 \quad (4.4)$$

Where $\theta_k^0$ represents the bias for neurons in the output layer. Finally, the output of the neurons in the output layer is

$$Y_k^0 = f \left( \sum_{j=1}^{N_h} W_{kj} y_j^h + \theta_k^0 \right) \quad (4.5)$$

With the above equations in the forward direction, the error propagation rule is used in the following steps.

**Step 1:** Construct the network and initialize the synaptic weights and random values.

**Step 2:** Apply an input vector to the network and find the corresponding output value.

**Step 3:** Compare the actual outputs with the desired outputs and find the error.

**Step 4:** Determine the amount by which each weight is to be changed and make necessary corrections.

**Step 5:** Repeat the steps until an acceptable error is attained.
4.3.3 Training the Network

The collected training points are passed into the designed network in order to teach it how to perform when different points than the training points are inserted into it. Computer software is usually used to do this process.

4.3.4 Testing the Network

Some of the composed Test points are reserved as test points. The purpose of test points is to test the performance of the designed ANN after its training is completed as these test points will be new to it and thus judge whether it gives perfect results or not.

4.4 Artificial Neural Networking SEPOSLC

4.4.1 Training the Neurons

- The acquired input is processed with the scalar variable ‘p.’ thus the input gets incremented for each feedback cycle.
- The processed data is then delayed for the specified duration by the delay unit. The delay is created by the presence of multiplexer in delay unit, which multiplexes the processed input again.
- In the weight unit, the delayed input is added up with the weight value, and dot product function of weight and bias is obtained. The dot product unit output is also multiplexed.
- The overall net sum of the output is applied to saturation unit, to check whether the linearized output meets the saturation condition or not.
- The obtained output is again processed with the scalar variable ‘p.’ thus the output gets corrected and get incremented for each feedback cycle. It is then given as output from ANN controller.
- The ANN controller output is then relationally compared with the repeating sequence signal.
- The compared output is given as gate signal to \( S_2 \). The compared and inverted output is provided as a gate signal to \( S_1 \).

**Figure 4.21** Flow chart Artificial Neural Network controller

Flow chart of ANN controller is shown in Figure 4.21. The inputs given to the ANN controller are the actual output voltage obtained from the SEPOSLC and rate limited reference voltage. These two inputs are multiplexed, and the selected input is passed to ANN controller.

Inside the ANN controller, the linear designing is implemented. The following steps are performed by the ANN controller. The acquired input is processed with the scalar variable ‘p.’ thus the input gets incremented for each feedback cycle.
The processed input is then delayed for the certain duration by the delay unit. The delay is created by the presence of multiplexer in delay unit, which multiplexes the processed input again.

In the weight unit, the delayed input is added up with weight value, and dot product function of weight and bias is obtained. The dot product unit output is also multiplexed. The overall net sum of the output is applied to saturation unit, to check whether the linearized output meets the saturation condition or not.

The obtained output is again processed with the scalar variable ‘p.’ thus the output gets corrected and get incremented for each feedback cycle. It is then given as output from ANN controller. The ANN controller output is then relationally compared with the repeating sequence signal. The compared output is provided as a gate signal to $S_2$. The compared and inverted output is given as gate signal to $S_1$.

Simulation model of ANN controller is shown in Figure 4.22.

Simulation model of ANN controller

4.4.2 Simulation Results

In this analysis, SEPOSLC converter is analyzed with constant and variable input voltage in the range of 11 V to 16 V. The essential parameters and its associated values for the SEPOSLC converter are given in Table 4.3. For a constant input voltage analysis, input voltage is kept as 11 V, 12 V, 13 V, 14 V, 15 V and 16 V. The performance of ANN controlled SEPOSLC converter at 12V constant input is shown in Figure 4.23.
Figure 4.23 Output voltage of ANN controlled SEPOSCLC

Figure 4.24 Capacitor voltage of ANN controlled SEPOSCLC

From the Figure 4.24, it is noted that output voltage reaches the set value of 72V in 0.007sec. The peak overshoot is eliminated at the time of starting. From the Figure 4.24, It is clear that capacitor voltage ripple is minimised very much compare to PI and FLC. For a variable input voltage analysis, the input voltage is varied from 11V to 13 V, 14V, 15V and 16V. The performance of ANN controlled SEPOSCLC converter for variable input at 11V to 13V is shown in Figure 4.25.
From the Figure 4.25, it is noted that the output voltage is maintained is at 72V with the help of ANN controller even when there is a change in input voltage. During the transition of the input voltage at a time of 0.05sec output voltage is oscillation is very less.

The performance of ANN controlled SEPOSLC converter for variable input at 14V to 12V is shown in Figure 4.27.
Figure 4.27 Variable input Voltage for ANN controlled SEPOSLC converter

Figure 4.28 Output Voltage of ANN controlled SEPOSLC converter for variable input of 14V to 12V

From the Figure 4.28, it is noted that the output voltage is maintained is at 72V with the help of ANN controller even when there is a change in input voltage. During the transition of the input voltage at a time of 0.05sec output voltage is oscillation is very less.

The performance of SEPOSLC converter using an ANN controller in the aspect of peak overshoot, voltage ripple and settling time are presented in Table 4.3.
Table 4.3 Performance of SEPOSLC converter using ANN controller

<table>
<thead>
<tr>
<th>Input voltage (Volt)</th>
<th>Peak overshoot (%)</th>
<th>Voltage ripple (%)</th>
<th>Settling time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>0</td>
<td>0.833</td>
<td>0.0072</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>0.9722</td>
<td>0.0072</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>1.11</td>
<td>0.0072</td>
</tr>
<tr>
<td>14</td>
<td>0</td>
<td>1.223</td>
<td>0.0072</td>
</tr>
<tr>
<td>15</td>
<td>0</td>
<td>1.355</td>
<td>0.0072</td>
</tr>
<tr>
<td>16</td>
<td>0</td>
<td>1.477</td>
<td>0.0072</td>
</tr>
</tbody>
</table>

From the Table 4.3, it is noted that voltage ripple increases when the input voltage increases. Meantime peak overshoot almost zero and settling time minimal value when the input voltage increases. The graphical representation of SEPOSLC with ANN is shown in Figure 4.29 to Figure 4.31.

![Graph](image)

**Figure 4.29** Peak overshoot
From the results, it is noted that peak overshoot and settling time are very less when the input voltage is greater than 13V. Voltage ripple decreases considerably for input voltage lesser than 13V.
4.5 Sliding Mode Controller

Sliding Mode Control does not need a disturbance waveform characterization to execute the control law. The simplest control law of SMC is $u=-M.\text{sgn}(\sigma)$. The main benefit of Sliding Mode Control (SMC) is the heftiness to unidentified disturbances. [Vidal-Idiarte, E et.al, 2004]. Necessary information of the disturbance is limited to the disturbance margin.

Conventional SMC was constrained by a discontinuous control law. Contingent upon the high-frequency switching, plant dynamics may or may not be a problem to contend with. There are methods to limit and eliminate the high-frequency switching associated with traditional SMC. The effective gains of the error compensator can be increased by using a sliding mode controller to tune the observer for duty ratio control.[Sira-Ramirez, H. Et.al, 2001]. It gives the robust performance for a drive. This control approach is nonlinear where the drive response is forced to "slide" along a predefined trajectory in a phase plane by a switching algorithm despite parameter variations or load disturbances [Wai, R.J., 2000 and Martin Ansbjerg 2004].

4.5.1 SMC Graphical Illustration

For an analysis consider a sliding mode controller (SMC) simple second-order undamped linear system with a variable plant gain, K. The SMC controller comprises two switches with the option of negative or positive feedback as shown in the figure below. (Figure 4.32 & Figure 4.33)
In either the negative or positive feedback case, the system can be shown to be unstable. However, when switched between the two states, not only can stability be achieved but the system can be made robust against variations in $K$.

Consider first the case of negative feedback, i.e. switch 1 closed. In this case,

$$X_1 = R - C$$

or

$$R = X_1 - C$$
Where \( X_1 \) = loop error

Differentiating this expression gives

\[
\frac{d}{dt}(R - X_1) = \frac{dC}{dt} = -X_2
\]  \hspace{1cm} (4.6)

\[
\frac{dX_1}{dt} = X_2
\]

To satisfy the loop relation, we can also write

\[
\frac{dX_2}{dt} = KX_1
\]  \hspace{1cm} (4.7)

Combining these equations gives

\[
\frac{d^2X_1}{dt^2} + KX_1 = 0
\]  \hspace{1cm} (4.8)

The general solution to this equation is

\[
X_1 = A \sin(\sqrt{K}t + \theta)
\]  \hspace{1cm} (4.9)

Differentiation of \( X_1 \) in (4.9) produces \( X_2 \) as in (4.10)

\[
X_2 = \frac{dX_1}{dt} = \sqrt{K} \ A \cos (\sqrt{K}t + \theta)
\]  \hspace{1cm} (4.10)

Combining these equations gives

\[
\frac{X_1^2}{A^2} + \frac{X_2^2}{(\sqrt{K}A)^2} = 1
\]  \hspace{1cm} (4.11)

This is the equation of an ellipse as shown below
Similarly, in the positive feedback mode, (switch 2 closed) the equations become:

\[ \frac{dX_1}{dt} = X_2 \]  \hspace{1cm} (4.12)

\[ \frac{dX_2}{dt} = KX_1 \]  \hspace{1cm} (4.13)

Combining these equations gives:

\[ \frac{d^2 X_1}{dt^2} - KX_1 = 0 \]  \hspace{1cm} (4.14)

The general solution to this equation is:

\[ X_1 = B_1 e^{\sqrt{K}t} + B_2 e^{-\sqrt{K}t} \]  \hspace{1cm} (4.15)

Differentiation of \( X_1 \) in (4.15) produces \( X_2 \) as in (4.16)

\[ X_2 = \frac{dX_1}{dt} = \sqrt{K} B_1 e^{\sqrt{K}t} - \sqrt{K} B_2 e^{-\sqrt{K}t} \]  \hspace{1cm} (4.16)
Squaring and combining these equations gives:

$$\frac{X_1^2}{4B_1B_2} - \frac{X_2^2}{4B_1B_2} = 1$$

(4.17)

This equation describes a set of hyperbolas as shown in the Figure 4.35

The straight line asymptote equations are obtained by setting $B_1B_2 = 0$ which gives:

$$4KX_1^2 - X_2^2 = 4KB_1B_2 = 0$$

(4.18)

The system can be switched front and back between these two modes. The superposition of the two-phase plane diagrams results in the figure shown below (Figure 4.36)
Consider that system at $t = 0$ is in negative feedback mode at point $X_10$. It moves along the ellipse until the positive feedback mode is conjured at point $B$. It will subsequently (ideally) move along $B_0$ to settle at 0 at steady state, where $X_1$ and $X_1$ are zero. Let us describe a straight line reference trajectory by the equation:

$$
\sigma = \sqrt{KX_1 + X_2}
$$

(4.19)

Where $C < \sqrt{K}$ so that the line slope is lower than and beyond the range of the variation in $K$.

$$
\sigma = CX_1 + X_2 = 0
$$

(4.20)

Observed that the -ve and +ve feedback ellipses and hyperbolas cross the reference trajectory in conflicting directions. This causes in a
zig-zag deviation about the reference trajectory until steady state is reached (as the operating condition is switched front and back between -ve and +ve feedback).

4.6 Total sliding mode control system

Total Sliding mode controller is the combination of the computed voltage controller and sliding mode controller; it is one of the effective nonlinear robust control approaches since it provides system dynamics with an invariance property to uncertainties once the system dynamics are controlled in the sliding mode.

Total sliding mode law

\[ u_{sm}(t) = u_{eq}(t) + u_{vs}(t) \]  \hspace{1cm} (4.21)

Control law for \( U_{eq} \):

\[ u_{eq}(t) = B_n^{-1}[-A_n \dot{\theta}(t) + \dot{\theta}_d(t) - k_1 \dot{\theta}_e(t) - k_2 \theta_e(t)] \] \hspace{1cm} (4.22)

Control law for \( U_{vs} \):

\[ u_{vs}(t) = B_n^{-1}W \text{sgn}[s(t)] \] \hspace{1cm} (4.23)

Where \( s(t) \) is the output of sliding surface, which is defined as follow:

\[ s(t) = \dot{\theta}_e(t) + k_1 \theta_e(t) + k_2 \int_0^t \theta_e(\tau) d\tau \]

In this analysis \( K_1 = 1 \).
4.6.1 **SMC in SEPOSCLC converter**

DC voltage controller is a sliding mode controller. The sliding surface is designed to control the duty ratio of the converter. The duty ratio tuned by the ANN is applied to this controller, and the sliding surface enforces converter to produce the required voltage.

\[ S_v = e_v + K_v \int e_v \, dt \]  

(4.24)

Where \( d \) is the duty ratio produced by the ANN

\( K_v \) is the controller design constant.

Since the output of ANN is again fine tuned using SMC, the ANN-SMC controller produces improved performance compare to all other controllers. Simulation model of SMC controller is shown in Figure 4.37.

![Figure 4.37 Simulation model of ANN SMC controller](image)

4.6.2 **Simulation Results**

In this analysis, SEPOSCLC converter is analyzed with constant and variable input voltage in the range of 11V to 16V. The essential parameters and its associated values for the SEPOSCLC converter are given in Table 4.4. For a constant input voltage analysis, input voltage is kept as 11V, 12V,
13V, 14V, 15V and 16V. The performance of ANN-SMC controlled SEPOSLC converter at 12V constant input is shown in Figure 4.38.

**Figure 4.38** Output voltage of ANN-SMC controlled SEPOSLC converter

**Figure 4.39** Capacitor voltage of ANN-SMC controlled SEPOSLC converter

From the Figure 4.38, it is noted that output voltage reaches the set value of 72V in 0.007sec. The peak overshoot is eliminated at the time of starting. Inductor current oscillation at time of starting is very less compare to PI and FLC. Figure 4.39 shows that capacitor voltage ripple is very less compare to all other controllers. For a variable input voltage analysis, the input voltage is varied from 11V to 13V, 14V, 15V and 16V. The performance of ANN-SMC controlled SEPOSLC converter for variable input at 11V to 13V is shown in Figure 4.40.
Figure 4.40 Variable input Voltage for ANN-SMC controlled SEPOSLC converter

From the Figure 4.41, it is noted that the output voltage is maintained at 72V with the help of ANN-SMC controller even when there is a change in input voltage. During the transition of the input voltage at a time of 0.05sec output voltage is not oscillated.

The performance of ANN-SMC controlled SEPOSLC converter for variable input at 14V to 12V is shown in Figure 4.42.
Figure 4.42 Variable input Voltage for ANN-SMC controlled SEPOSCLC converter

Figure 4.43 Output Voltage of ANN-SMC controlled SEPOSCLC converter for variable input of 14V to 12V

From the Figure 4.43, it is noted that the output voltage is maintained is at 72V with the help of ANN-SMC controller even when there is a change in input voltage. During the transition of the input voltage at a time of 0.05sec output voltage is not oscillated.

The performance of SEPOSCLC converter using a ANN-SMC controller in the aspect of peak overshoot, voltage ripple and settling time are presented in Table 4.4.
Table 4.4 Performance of SEPOSCLC converter using ANN-SMC controller

<table>
<thead>
<tr>
<th>Input voltage(Volt)</th>
<th>Peak overshoot (%)</th>
<th>Voltage ripple (%)</th>
<th>Settling time(sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>0</td>
<td>0.029</td>
<td>0.007</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>0.029</td>
<td>0.007006</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>0.02916</td>
<td>0.007002</td>
</tr>
<tr>
<td>14</td>
<td>0</td>
<td>0.02916</td>
<td>0.006996</td>
</tr>
<tr>
<td>15</td>
<td>0</td>
<td>0.02916</td>
<td>0.007003</td>
</tr>
<tr>
<td>16</td>
<td>0</td>
<td>0.02916</td>
<td>0.006996</td>
</tr>
</tbody>
</table>

From the table 4.4, it is noted that peak overshoot is eliminated. Voltage ripple and settling time are almost constant for wide input voltage range. The graphical representation of SEPOSCLC with ANN-SMC controller is shown in Figure 4.44 to Figure 4.46. Comparison of ANN and ANN-SMC is shown in Figure 4.47 to Figure 4.48.

![Graph](image-url)  
**Figure 4.44** Peak overshoot
Figure 4.45 Voltage ripple

Figure 4.46 Settling time
Figure 4.47 Peak overshoot

Figure 4.48 Voltage ripple
Comparison of all controllers in case of change in input voltage is shown in Figure 4.50.

From the Figure 4.50 it is noted that ANN-SMC controller eliminates output voltage oscillation in case of change in input voltage performs better than all other controllers.

The Figure 4.50 analyses the performance of line regulation of a converter similarly load regulation of converter using various controllers are analysed with variable load is shown in Figure 4.51. For an analysis load is increased at time of 0.05 seconds.
Figure 4.50 Comparison of all controllers with change in input voltage
From the Figure 4.51 it is noted that during change in load PI and FLC based SEPOSLC converter output voltage is varied and remains in reduced voltage. ANN controller produces output voltage with the very minimum change.
during change in load. Whereas ANN-SMC based converter produces oscillation free output voltage compare to all other controllers.

Comparison of all controllers for 14V input is presented in Table 4.5.

<table>
<thead>
<tr>
<th>Controller</th>
<th>Peak overshoot (%)</th>
<th>Voltage ripple (%)</th>
<th>Settling time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PI</td>
<td>18.35</td>
<td>0.0972</td>
<td>0.04</td>
</tr>
<tr>
<td>FLC</td>
<td>0</td>
<td>4.4</td>
<td>0.0183</td>
</tr>
<tr>
<td>ANN</td>
<td>0</td>
<td>1.223</td>
<td>0.0072</td>
</tr>
<tr>
<td>ANN-SMC</td>
<td>0</td>
<td>0.02916</td>
<td>0.006996</td>
</tr>
</tbody>
</table>

From the Table 4.5, it is noted that ANN controller produces better performance compare to PI and FLC-based SEPOS LC in the aspects of overshoot and voltage ripple. Whereas ANN-SMC results in better performance than PI, FLC and ANN in the aspects of overshoot, voltage ripple and settling time. The graphical representation of various controllers is shown in Figure 4.52 to Figure 4.54

![Graph of peak overshoot](image)
4.7 Summary

In this chapter, intelligent controllers such as a Fuzzy logic controller, ANN, and ANN-SMC controller are used to control SEPOSLC. Compared to all other controllers ANN-SMC produces voltage ripple which is...
comparatively negligible. Voltage ripple is low for a wide range of input voltage especially in voltage ripple is less for the dynamic case that is in the case of a change in input voltage. Hence ANN-SMC controller is suitable for constant output voltage with frequent variable input voltage applications.