Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 6, No.3, June : 2024

GRID INTEGRATION OF PHOTOVOLTAIC SYSTEM USING MCFNN-BASED POWER QUALITY ENHANCEMENT VIA SL-SC BOOST CONVERTER-FED CASCADE MULTILEVEL HB INVERTER

Hajitali M, M.Tech Student, Department of Electrical and Electronics Engineering, Al Falah University, Faridabad, Haryana, India.
Mati ur Rahman, Assistant Professor, Department of Electrical and Electronics Engineering, Al Falah University, Faridabad, Haryana, India.

ABSTRACT
In today's developed nations, there have been significant advancements in the production of electricity from renewable sources. With the aid of a proposed cascaded feedforward neural network (CFNN) controller-fed advanced power conversion devices like symmetrical hybrid (SH), switched inductor (SL), and switched capacitor (SC) based boost converter and reduced switch cascade multilevel inverter (CMLI), the main goal of this research is to investigate the grid integration of solar systems within allowable total harmonic distortion (THD) limits. The power conversion SH-SL-SC converter has a low switch count and a high voltage gain. The voltage and current profiles of CMLI have low harmonic content and the fewest power electronic switches. The proposed CFNN-MPPT controller works better in various weather circumstances when generating a control signal for the suggested SH-SL-SC boost converter, where it requires a substantial amount of data to function properly.

Keywords:
Photovoltaic; maximum power point tracking; cascaded feedforward neural network (CFNN) controller, particle swarm optimization; multilevel inverter; total harmonic distortion.

I- Introduction
The global conscience's revision of energy policies to combat CO2 emissions is the reason for the growing interest in renewable energies. Proposals to accomplish this aim include limiting harm to the environment and ecosystem, encouraging the development of alternative resources, and replacing fossil fuels with cleaner energy sources. Photovoltaic energy has a long lifespan, is highly reliable, and is clean. Thus, it is regarded as one of the renewable energy that is most sustainable. These systems can be placed in or close to the area where they are needed, preventing transmission losses and helping to lower CO2 emissions in cities[1].The photovoltaic module, which serves as the conversion unit in this generation system, is created by connecting a number of solar cells in series and parallel with their protection mechanisms (figure 1). In addition, the amount of energy gained is contingent upon solar radiation, cell temperature, and voltage generated within the photovoltaic module[2].

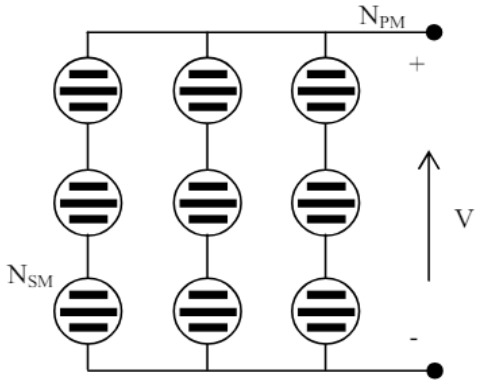


Fig. 1. PV module cell.

If PV penetration is sufficiently high, PV systems may have multiple negative grid implications. Among these include reverse power flow, distribution system overvoltage, difficulty regulating

voltage, phase imbalance, PQ problems, increased reactive power, and difficulty identifying islanding [3]. The problem of creating prediction time series data based on a stochastic (probabilistic) process with a high degree of accuracy can be formulated based on the background. One of the shortcomings of the widely used ARIMA method for forecasting is its inability to take into account the possibility of nonlinear correlations between data. Analyzing data with period patterns that change over time is likewise less likely to satisfy the stationarity assumption. Predictions are frequently less accurate as a result of this[4]. To produce more accurate findings, a more flexible and nonlinear-nonparametric CFNN model can be developed and used to predict time series data. In order to find the best architecture, the input selection process is the crux of the CFNN modeling problem.An Artificial Neural Network (ANN, sometimes simply referred to as NN alone) is a collection of simple, interconnected units known as neurons. Particularly when it comes to having several inputs and one output, every neuron is a mapping[5]. The neuron's output is determined by the total of its inputs. An activation function is the function that a neuron uses in its output[6]. The symbol for a single neuron indicates the number of arrows originating from the neuron because the single output of the neuron can be utilized as an input to some additional neurons. The term "perceptron" refers to a straightforward network architecture that consists simply of the input and output layers. The signal is transferred straight from the input layer to the output layer in a perceptron[7]. There is a linear link between input and output. Weighted sums of signals are transmitted from the input to the output, showing the direct connection between the two levels.A multi-layer neural network, or Multi-Layer Perceptron (MLP), is created by adding layers between the input and output. MLP network is also known as a Feed Forward Neural Network (FFNN) in neural network modeling. The hidden layer is an extra layer that is included in the FFNN network[8].

II- PV module modeling:

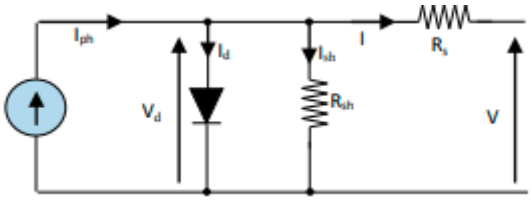


Figure-2 Equivalent circuit of solar cell

$$I= I_{ph}- I_d-I_{sh}= \{ I_{SCR} +K_i(T-T_r)\} \left(\frac{G}{1000}\right)- I_0[\exp\left(\left(\frac{q}{aKT}\right) (V+IR_s)\right)-1]- \left(\frac{V+I*R_s}{R_{sh}}\right) \quad (1)$$

I_{ph} is solar generated voltage, Iph depend on solar radiation and cell temperature .
I_0 is the reverse saturation current of diode.

$$I_o= I_{rs}\left\{\frac{T}{T_r}\right\} \exp\left[\frac{qE_{go}}{aK}\right] \left(\frac{1}{T_r}-\frac{1}{T}\right)] \quad (2)$$

Table-1 Solar cell parameters[9,10]

Sr.	Represented by	Parameters
1	V	Output voltage of solar cell
2	I	Output current of solar cell
3	I_{SCR}	short circuit current of solar cell
4	Ki	Short circuit temperature coefficient
5	T_r	Reference temperature
6	I_{rs}	Saturation current at reference temperature
7	T	Temperature (K)
8	K	Boltzmann constant (1.38*10-23 J/K),
9	Rsh	Shunt resistance of PV cell in ohm

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 6, No.3, June : 2024

N_s, is number of series solar cells in a module
N_p, is number of parallel solar cells in a module
If Ns is the number of series solar cell per module and Np is parallel solar cell per module,
The solar output current I

$$I=N_pI_{ph}- N_pI_0 [\exp ((\frac{q}{aKT})(\frac{V}{N_s}+\frac{IRs}{N_p}))-1\}-\frac{N_p}{R_{sh}}(\frac{V}{N_s}+\frac{IR_s}{N_p}) \quad (3)$$

Optimization problems are common in many real-world applications, such as those in engineering, finance, economics, transportation, and medical. Experts in these fields commonly use optimization techniques to find the best options and trade-offs that maximize the trade-offs of the optimal decisions. As an alternative, maximize profits, sales, efficiency, and other variables while minimizing costs, risks, and losses [11]. The basic idea behind a novel modified PSO approach for optimization is to create an algorithm that moves about the surrounding space of the fitness function or test function and looks for the optimal place [12]. In a neural network, a neuron (NN) is the most fundamental unit. The synaptic weight serves as the link between neurons. Figure 6 displays the five hidden tiers and the production layer of this NN. A CFFNN is one type of NN that fits this description; Figure 5 shows how one is constructed. In CFFNNs, the output nodes communicate with the hidden nodes, and the contribution nodes communicate with each other.

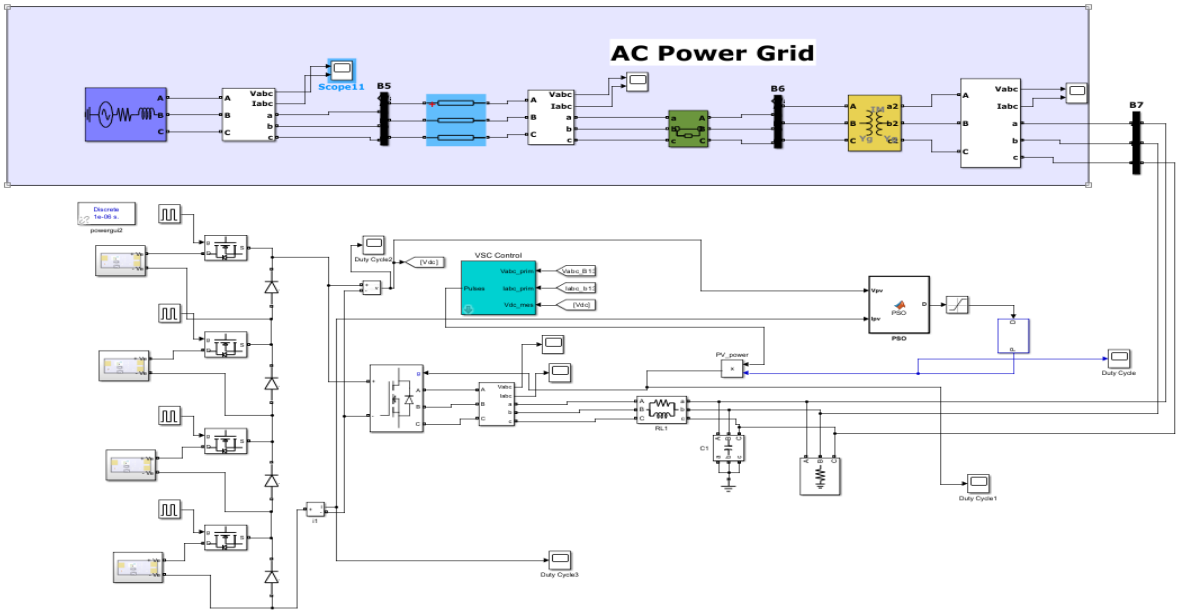
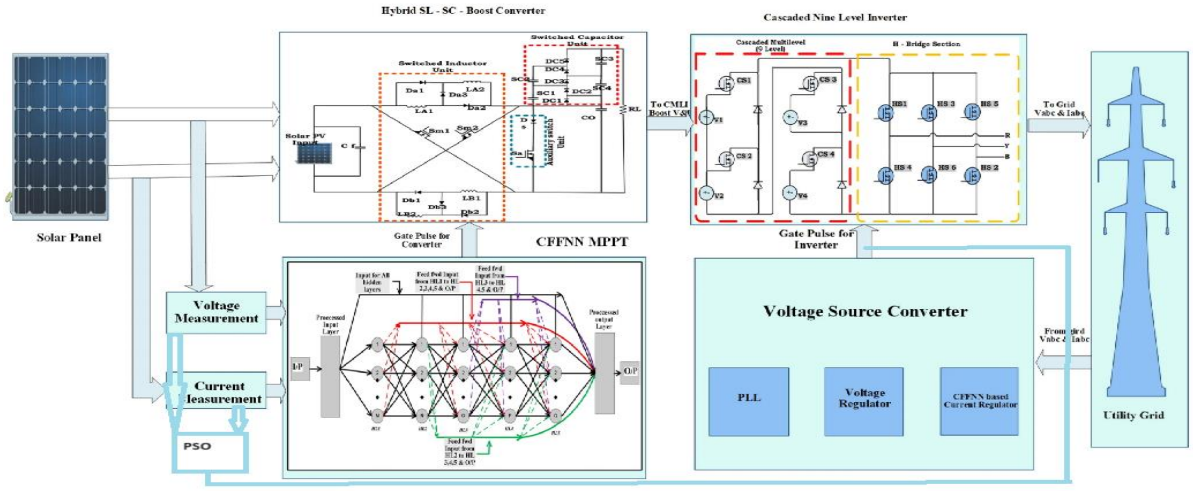


Fig-3 MATLAB Simulink of proposed grid Integration of photovoltaic system



Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 6, No.3, June : 2024

Fig- 4 Controllers used in proposed grid Integration of photovoltaic system

III- PSO Algorithm

We define the issue as a d-dimensional simple optimization problem where d dimensions are optimized [13]. The multidimensional test function characterizes the degree to which the ith particles' locations in multidimensional space align with the desired target. The velocity of the particle to be managed (expressed by the d component) determines the location of the multidimensional algorithm. Therefore, the ith particle's position may be xi(xi,0,...,xi,d), and its velocity could be vi (vi,0,..).In the novel modified pso position of ith particle is updated with equations (4) and equations (5).

$$V_{t+1}=\omega_t V_t + c_1 r_1 (g - x_t) - c_2 r_2 (P - x_t) \quad (4)$$

$$x_{t+1}= x_t + V_{t+1} \quad (5)$$

Where x_t and V_t are the current positions and current velocity of the ith particle whereas P is the best to position fitness value, g is the position obtain its best fitness value by an entire swarm, c_1, c_2 are learning constant whereas r_1, r_2 are the random number in the range of [0,1] and ω_t is the damping parameter that regulates the transition between exploration and exploitation phases in the presented algorithm [14].

Table -2 PSO prameters

PSO parameters	Magnitudes
Swarm coefficient c_1	2.2
Swarm coefficient c_2	2.2
Min weighting coefficeant w_{min}	0.5
Max weighting coefficeant w_{max}	0.8
Iteration k	105
Swarm size n	20
Lower limite of variables $\Delta d(k)min$	[-0.2, -0.22]
Upper limite of variables $\Delta d(k)max$	[0.2, 0.22]

IV- Cascaded feedforward neural network (CFNN) controller

Every layer in the network has an initial connection from the input to that layer, and then there is a subsequent link from that layer to the layers below. The relationship between the input and the output of a perceptron is direct. On the other hand, the relationship between input and output in a CFFNN is not direct. The relationship of the unseen layer is shaped by a stimulation function that is both linear and non-linear. Combining perceptron and multilayer networks can be used to create a network that connects the input and output layers. Cascade forward refers to the CFFNN that this connection architecture generates. Equation (6) presents a possible solution for the CFFNN model.

$$y = \sum_{i=1}^{n} f^i w_i^i x^i + f^0 \left(\sum_{j=1}^{k} w_j^0 f_j^h \left(\sum_{i=1}^{n} w_{ji}^h x_i\right)\right) \quad (6)$$

where f_0 is the activation function on the output layer, f_j^h is an activation function on the hidden layer,w_j^0 is the weight of the jth neuron at the output layer, w_{ji}^h is the weight of the jth neuron at the hidden layer, and w_i^i is the weight of the ith neuron at the input layer.where w_b is the weight of the bias from the input. The CFNN model often employs time series data. The outcome is present data at X_t level because neurons in the input layer delay the time series data represented by the data at X_{t-1}, X_{t-2}, X_{t-3}, and X_{t-p} levels. To increase the size of the entire network by the number of neurons in the contribution layer, the network weight must be computed and modified. The feedforward algorithm on CFNN, known as backpropagation, consists of three stages: initial weight computation, pattern error counting, and further weight calculation. The error is computed during the feedforward phase,

and then the development continues with the feedforward design (the variation in the result to the target)[15].

$$y = \sum_{i=1}^{n} f^i w_i^i x_i + f^0 \left(w^b + \sum_{j=1}^{k} w_j^0 f_j^h \left(\sum_{i=1}^{n} w_{ji}^h x_i\right)\right) \quad (7)$$

To ensure that everything is still exact, the weights must be adjusted before the computation is repeated. The operation is continued as long as no errors or iteration halts are discovered. The conjugate gradient optimization technique for changing the CFNN's weights is briefly covered in this section. Assume that is a weight vector representing the length and that the objective is to find all of the network weights as given by equation (8).

$$e = 0.5((X_t - \widehat{X_t})^2) \quad (8)$$

The dimension of the optimistic positive matrix is s*s. However, QT = Q has a definition of Q. The steps in the Conjugate Gradient Optimization process are as follows:
Step 1: Select 0 as the starting location and set k to 0.
Step 2: Utilizing equation (9) as a guide, ascertain the gradient of the starting weight.

$$g^0 = \frac{de}{dw^0} = \frac{de}{dw\ if\ w=w^0} = \left|\frac{de}{dw_1^0} \dots\dots\dots \frac{de}{dw_s^{(0)}}\right| \quad (9)$$

If $g^{(0)} = 0$ then stop and then obtained the optimal weight Ω (0). Else, $d^{(0)} = g\ (0)$
Step 3: Determine α_k using equation (10)

$$\alpha_k = \arg\min_{\alpha \geq 0}\ e(w^k + \alpha d^k) = -\frac{g^{(k)T} d^{(k)}}{d^{kT} Q_{d(k)}} \quad (10$$

Step 4: Determine $\Omega^{(k+1)}$ using equation (11)

$$\Omega^{(k+1)} = \Omega^{(k)} + \alpha_k\ d^{(k)} \quad (11)$$

Step 5: $g^{(k+1)} = \frac{de}{dW^{(k+1)}}$ if $g^{(k+1)} = 0$ stop and the optimal weight is $w^{(k+1)}$

Step 6: Determine β_k using equation (12)

$$\beta_k = \frac{g^{(k+1)} Q d^k}{d^{(k)T} Q d^k} \quad (12)$$

Step 7: Determine $d^{(k+1)}$ using equation (13)

$$d^{(k+1)} = -g^{(k+1)} + \alpha_k d^{(k)} \quad (13)$$

Step 8: k = k+1: drive to stage 3
Epoch repetition is also referred to as weight searching on CFNN in feedforward neural networks (FFNN). To start the repetition method, the program must not have satisfied the iteration termination condition before epoch k = K. The direction vector is reset after each iteration because this method can't guarantee convergence in the n steps, and the procedure is continued until the termination condition is met. In the non-linear model, Q is a nonconstant Hessian matrix that is produced for each iteration. An algorithm for eliminating Q is employed to make the method straightforward, with the

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 6, No.3, June : 2024

only sources of algorithm reliance being the function and gradient value for each iteration. One of the many formulae for substituting Qd(k) with other forms is the Hestenes-Stiefel formula, or Qd(k). Thus, the k may be expressed as in equation (14).

$$\beta_k = \frac{g^{(k+1)T}[g^{(k+1)} - g^k]}{d^{(k)T}[g^{(k+1)} - g^k]} \qquad (14)$$

The job in issue is the stimulation task between the contribution and production layers, and the stimulation function from the contribution layer to the production layer is represented by the weight in the contribution layer's stimulation function. equation (6) changes to resemble equation (7) when an additional bias is applied to the contribution layer and each neuron in the hidden layer has an activation function of fh.

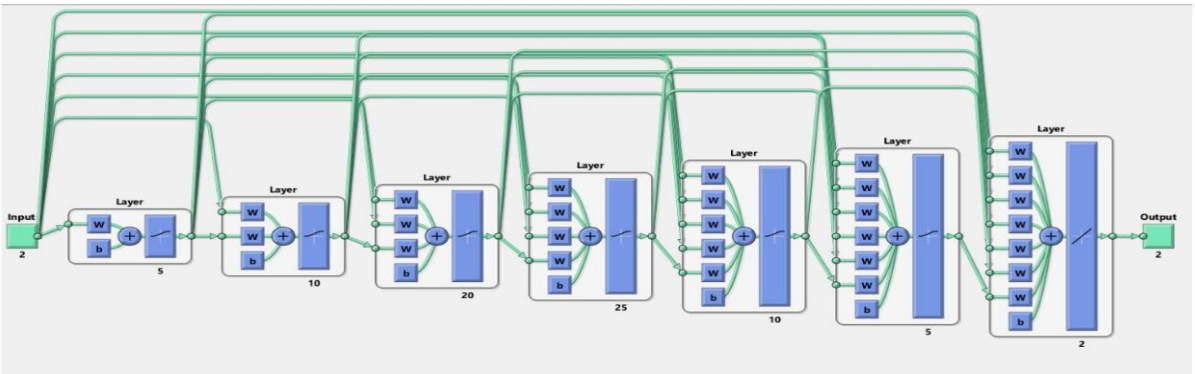


Figure 5. The architecture of CFFNN

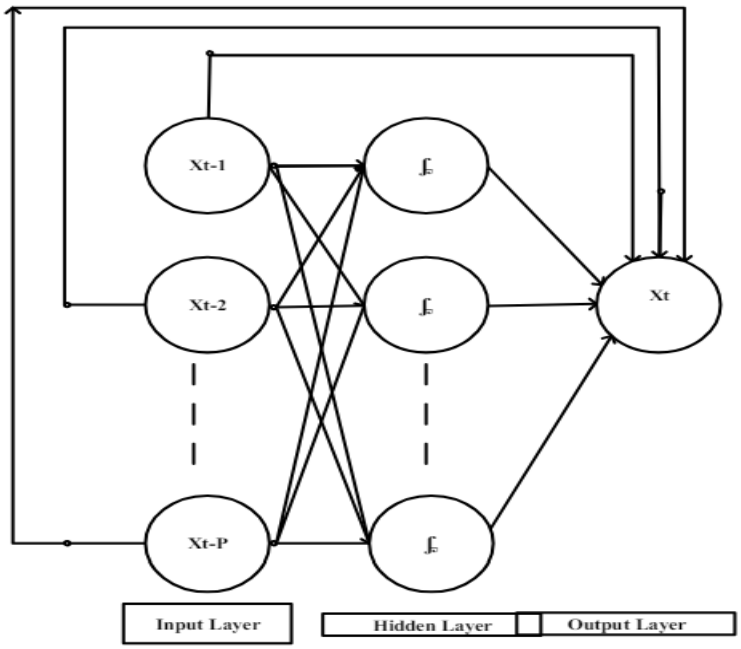


Figure 6 . CFFNN architecture – five layers

V- Result

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 6, No.3, June : 2024

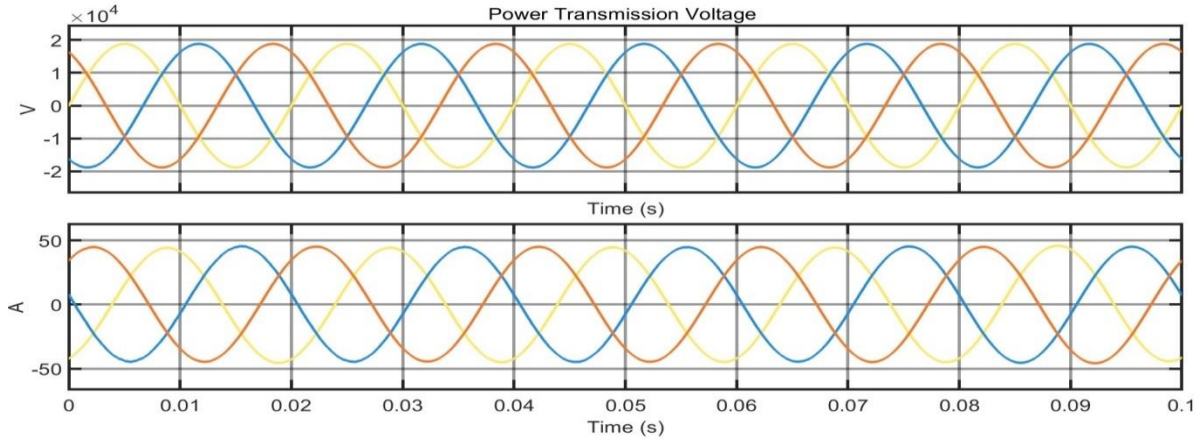


Fig-7 Grid voltage and current

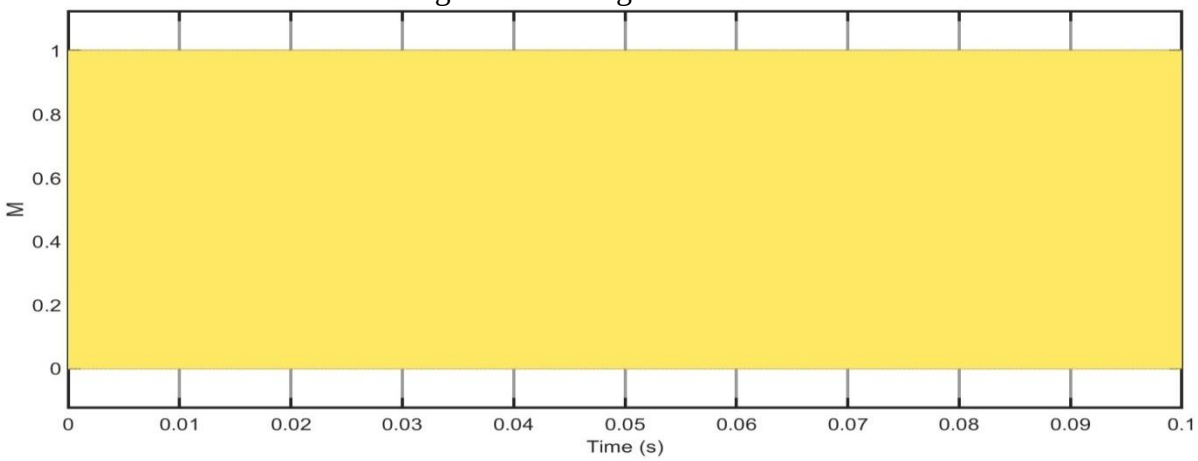


Fig- 8 Custom Neural Network generated pulse to trigger boost converter

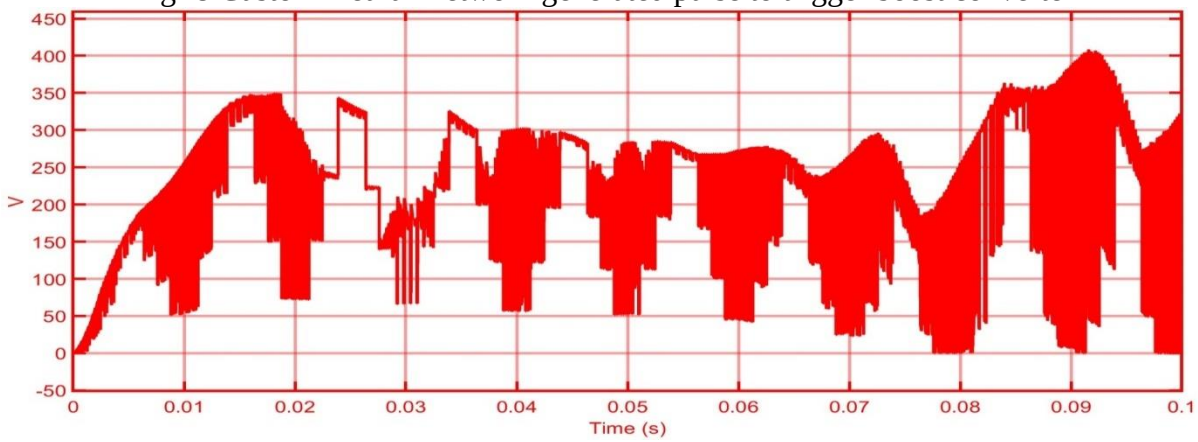


Fig- 9 Solar Voltage

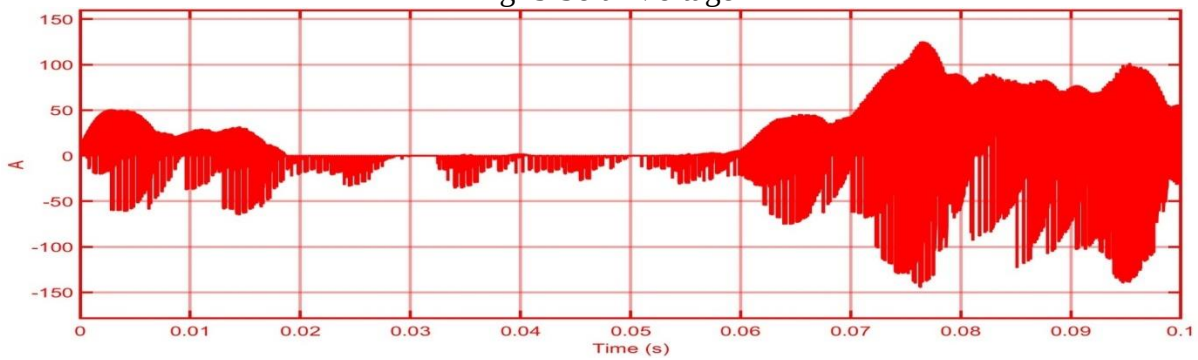


Fig-10 Solar current

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 6, No.3, June : 2024

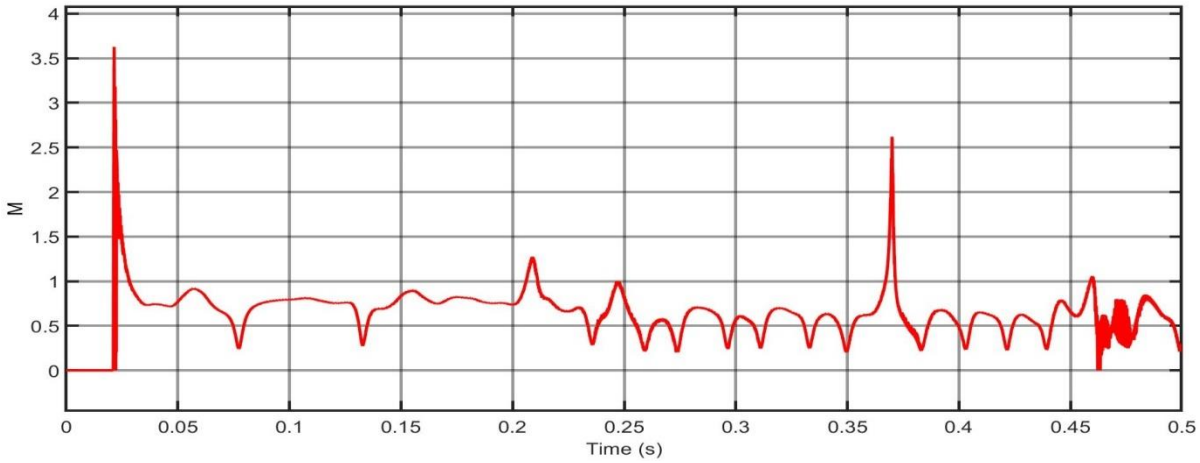


Fig-11 THD in load voltage

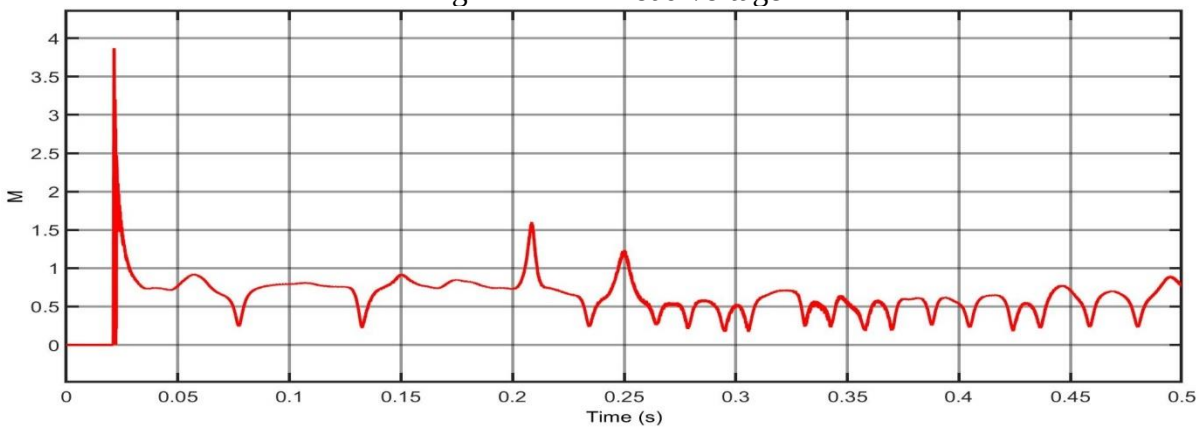


Fig-12 THD in load Current

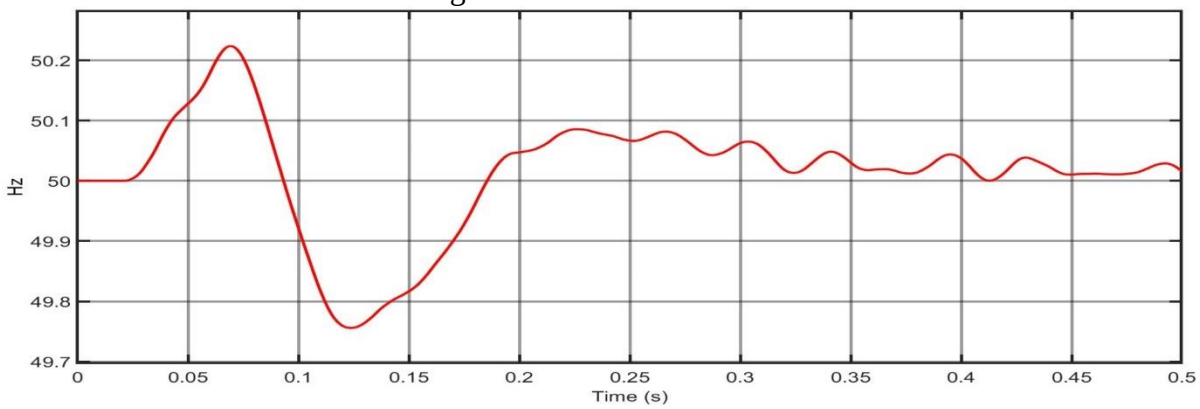


Fig-13 Grid Frequency

VI- Conclusion:

In this study, a 15 kW PV system with a reduced switching 9-Level CMLI was modeled in Matlab using the CFNN control approach. A variety of operating conditions, such as active and reactive power, power factor, load voltage, and current THDs, were evaluated for the system's performance, and the model was simulated. The integration and operation of various control algorithms with PV systems was the subject of the second section of this study report. The systems' performance indicates that the load voltage and current THD values with CFFNN are 8.87% and 8.72%, respectively. In grid-connected systems, the power factor values are 0.989 but neither of the control methods takes longer to obtain the rated voltage, current, or active power.

VII- Reference:

[1] G. Walker, “Evaluating MPPT converter topologies using a Matlab PV model”, Journal of Electrical & Electronics Engineering, Australia, Vol.21, No. 1. (2001), pp. 49-56.
[2] M.G. Villalva, J.R. Gazoli, E. Ruppert “Modeling and circuit-based simulation of photovoltaic arrays”. Brazilian Journal of Power Electronics, (2009) vol 14, nº 1 pp 35-45.
[3] A. Mellit, “Modelling and simulation of a stand-alone photovoltaic system using an adaptive artificial neural network: Proposition for a new sizing procedure” Renewable Energy 32 (2007) pp 285-313.
[4] M. G. Villalva, J.R. Gazoli, E. Ruppert “Comprehensive approach to modeling and simulation of photovoltaic arrays”. IEEE Transactions on power electronics, vol. 24, nº 5, pp. 1198-1208, May 2009.
[5] M. Vorländer, "Auralization: Fundamentals of Acoustics Modelling Simulation Algorithms and Acoustic Virtual Reality" in , Berlin:Springer, 2008.
[6] "Immersive sound: The art and science of binaural and multi-channel audio" in , New York:Routledge, 2018.
[7] H. Wierstorf, M. Geier, A. Raake and S. Spors, "A Free Database of Head Related Impulse Response Measurements in the Horizontal Plane with Multiple Distances", *Engineering Brief presented at the 130th Audio Engineering Society Convention*, 2011.
[8] R. Bomhardt, M. de la Fuente Klein and J. Fels, "A high-resolution head-related transfer function and three-dimensional ear model database", *Proceedings of Meetings on Acoustics*, vol. 29, 2017.
[9] Z. Haraszy, D. Ianchis and V. Tiponut, "Generation of the head related transfer functions using artificial neuronal networks", *13th WSEAS International Conference on Circuits*, 2009.
[10] M. Lovedee-Turner and D. Murphy, "Application of Machine Learning for the Spatial Analysis of Binaural Room Impulse Responses", *Applied Sciences*, 2018.
[11] A. Kohlrausch, S. van de Par, R. van Eijk and J. F. Juola, "Human performance in detecting audio-visual asynchrony", *The Journal of the Acoustical Society of America*, vol. 120, no. 3084, 2006.
[12] V. Melo, R. Tenenbaum and R. Musafir, "Intelligibility assessment in elementary school classrooms from binaural room impulse responses measured with a childlike dummy head", *Applied Acoustics*, no. 74, pp. 1436-1447, 2013.
[13] R. Tenenbaum, V. Melo and J. Naranjo, "Virtual reality: a new approach to validate computer modeling auralizations by using articulation indexes" in Virtual reality: technologies medical applications and challenges, New York:Nova Publishers, pp. 55-71, 2015.
[14] R. Tenenbaum, F. Taminato, V. Melo and J. Torres, "Auralization generated by modeling HRIRs with artificial neural networks and its validation using articulation tests", *Applied Acoustics*, no. 130, pp. 260-269, 2018.
[15] Zhang JP, Qi M 2005 Neural network forecasting for seasonal and trend time series, European Journal of Operational Research, 160, 501–514.