

An ANFIS-PID Based Boost Converter Control Scheme

John Darvill, Alin Tisan, Marcian Cirstea
Anglia Ruskin University, Cambridge, UK

john.darvill@student.anglia.ac.uk , alin.tisan@anglia.ac.uk, marcian.cirstea@anglia.ac.uk

Abstract- The PID algorithm has proven to be a popular and widely used control method, due to its relative simplicity and robustness. Despite this the linear nature of the algorithm means it doesn't provide optimal control to non-linear systems. This paper presents a method of improving the performance of the PID controller using an ANFIS based controller to provide gain scheduling. This control scheme is applied to a Boost Converter circuit and simulated within the PSIM modelling environment. The simulation results indicate that using the ANFIS controller provides a fast system response with minimal errors even under dynamic operating conditions. The ANFIS controller is also shown to simplify the design flow in comparison to the popular Fuzzy-PID gain scheduling method.

I. INTRODUCTION

The first use of the classical PID controller is widely attributed to Elmer Sperry in 1911 in his work on automating ships steering. The first known publication on the topic was published eleven years later by Minorksy [1]. Despite being a century old, the classical PID controller remains a popular control solution due to the algorithms flexibility and robustness. In the modern era a number of modelling packages are available which offer quick and easy design and tuning of the PID controller helping to sustain its use.

Despite its popularity and advantages, the PID algorithm is not an ideal solution for all control solutions. As the algorithm provides a control signal which is essentially proportional to the error input, the PID is fundamentally linear. This means that sub optimal control is provided for non-linear systems. This has led to the need for improvements in control systems to be able to handle non-linearity. This is particularly prevalent in the more complex plants which are often present in modern systems.

A number of different approaches have been taken in attempting to develop control systems which are more suitable for non-linear plants. There are now two distinct sets of controller designs for non-linear systems – adaptive control and gain scheduling. In an adaptive controller the control law is adaptive to the altering dynamics of the plant. As the plant transitions into different operating conditions, the controller responds by updating its control parameters for that state. One of the most commonly used types of adaptive control is sliding mode control. In this scheme the controller provides a discontinuous control signal which forces the system or “slide” along a section of its normal operating parameters. As a popular method of non-linear control, sliding mode control has been shown to be a good solution for the control of power converters [2][3]. Sliding mode control has been shown to be a robust method of control for

non-linear systems. However its implementation is a complex task which often requires quite detailed knowledge of the underlying plant.

Gain scheduling is another popular method employed for non-linear control. In a gain scheduling controller a number of linear controllers are employed depending on the state of the system. Each of the separate linear controllers is able to give optimal control for a different operating state of the system. In a real system which employs a gain scheduling controller there will be just one linear controller, commonly a PID controller, which is adapted to the systems operating characteristics by altering the gain of the controller. This approach is an attractive option for power converters where the PID parameters can be altered according to the load, as well as for disturbances. An approach which has proven to be popular is the use of a Fuzzy logic controller to provide the gain-scheduling for the PID, such as in [4] and [5]. This approach has also been successfully adopted in power converter systems, such as in [6] and [7]. These papers demonstrate the improved performance which can be achieved in non-linear systems when this type of non-linear controller is introduced.

The main attraction of the fuzzy logic controller in the gain scheduling scheme is its ability to make decisions. However this is offset by the need for expert level of knowledge in the system to be able to design a controller with suitable performance. This is in contrast to the other popular family of artificial intelligence (AI) based controllers - Neural Networks. In systems which employ the neural network there is an inherent ability for the system to learn and be trained, simplifying the design process. However there is an inability of such systems to make decisions which limits how useful they can be in control solutions. In order to best exploit the capabilities of both types of controller, it is possible to utilize a neural network which is trained to perform as a fuzzy logic controller. One system which has been shown to achieve the best of best types of AI controller is the Adaptive Neuro-Fuzzy Inference System (ANFIS) [8]. The ANFIS algorithm is an adaptive network which has a similar training scheme to the neural network whilst offers equivalent performance to a fuzzy logic inference system. This makes it an ideal tool for use in a gain scheduling PID based control system for power converter systems.

The need for a reduction in carbon emissions has led to an increasing interest in renewable energy sources as a means for energy production. Renewable energy is particularly

suitable to distributed generation (DG), whereby locally produced energy is used to power a small local area. This is in contrast to the traditional method of having a small number of power plants providing power to a wide area. In this paper a solar photovoltaic (PV) system is introduced which is suitable for DG systems, consisting of a PV cell, a MPPT tracking algorithm and a boost converter to regulate the output power. A popular and emerging research theme for DG systems is the so called demand side management [9], which is concerned with intelligently monitoring the load. An example of this is incorporated in the boost controller through the use of the ANFIS-PID gain scheduling controller. The boost converter is a highly non-linear plant which is controlled using the ANFIS-PID methodology to produce a stable 48VDC output. The ANFIS-PID monitors the load demand of the converter and adapts the PID controller parameters accordingly.

The rest of this paper is organized as follows: firstly the control problem is presented, then the ANFIS network is introduced, next the system model is dealt with in greater detail before simulations are presented and discussed and finally conclusions end the paper.

II. CONTROL PROBLEM

The main goal of the system presented in this paper is to take the power produced by a PV cell and convert this into a stable 48VDC output. The PV cell selected for this system is the BP3230 which is capable of producing 230W, with current of 8A at maximum power. The output of the solar cell is applied firstly to a pre-regulator, which is used to apply a MPPT algorithm, and finally to a boost converter to provide the stable output. The main interest of this paper is in the novel control of the boost converter which features an ANFIS-PID control solution. A diagram of the full circuit is given in Fig 1 showing the PV cell, MPPT circuit, the boost converter and the ANFIS-PID controller.

A. Mathematical Model of the Boost Converter

The performance of the boost converter circuit, shown in Fig 1 is influenced heavily by the current through the inductor. When this inductor doesn't fully discharge its stored energy during a switching period, the converters behaviour is considerably simpler to model. This mode of operation is known as continuous conduction. The other mode of operation occurs when the inductor is allowed to fully discharge for some period, also known as discontinuous conduction. In this mode of operation, the equations governing the behavior of the circuit become much more complex. Given this it is therefore preferential for the converter to remain in continuous conduction mode. The average output of the converter is controlled through the duty cycle of the switch, which can be defined by either the ratio of the input voltage to the output voltage or of the period that the switch is closed to the total switching period as shown in Eq (1). In this equation T_s denotes the total switching period of the inductor whilst D represents the duty cycle of the

switch. This equation shows how the output can be easily controlled by adjusting the duty ratio of the switch when in continuous conduction mode.

In order to model the inductor in the boost converter there are two equations which must be understood – one for the ripple current and one for the voltage across the inductor. The

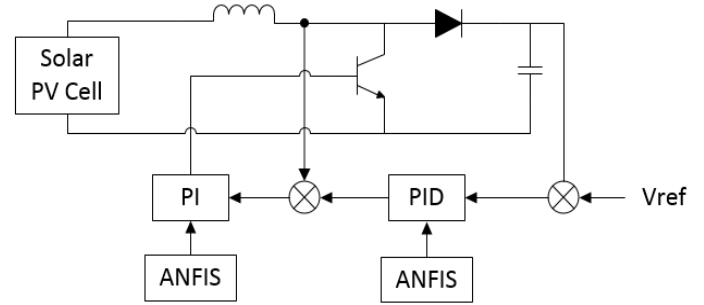


Fig 1 Diagram of the System

$$\frac{V_{out}}{V_{in}} = \frac{T_s}{T_{off}} = \frac{1}{1-D} \quad (1)$$

$$D = 1 - \frac{V_{in}}{V_{out}}$$

equation used to model the voltage across the inductor is a simple function of inductance and change in current, as shown in Eq.(2). The ripple current of the boost converter is important in the modeling of the circuit as controlling this helps to keep the converter in the continuous conduction mode of operation. The equation for modeling the ripple current is shown in Eq. (3) and minimizing this value helps to increase the operating range of the circuit.

$$V_L(t) = L \frac{di}{dt} \quad (2)$$

$$\Delta I_L = \frac{V_{in_{min}} D}{f_s L} \quad (3)$$

Another important defining feature of the boost converter is the output capacitor as this has a large effect on the quality of the DC output. The capacitor is important as it smoothes out the fluctuations which are introduced into the circuit through the switching action, meaning it has a large influence on the output voltage ripple of the circuit. The overall ripple of the circuit can be modeled by the equation shown in Eq. (4), with f_s being the switching frequency.

$$\Delta V_{out} = \frac{I_{out_{max}} D}{f_s C} \quad (4)$$

B. PID Controller

The PID controller is a widely used controller, firstly introduced in the early parts of the 20th century. This controller consists of three separate branches, a proportional branch, an integral branch and a derivative branch. The three

terms of the controller serve to respond to different elements of the error signal, with the proportional part acting immediately on the current error, the integral adding a contribution equivalent to the history of all errors and the derivative action predicting the future contribution of the errors. The addition of the integral and derivative action to the basic proportional element enhances the performance of the controller by reducing steady state errors and the rise/fall times respectively. Eq. (5) shows the basic PID algorithm, which is used in the controller, with e being the error of the output.

$$K_p \cdot e + \frac{1}{T_r} \cdot \int e + T_d \cdot e \quad (5)$$

The simple PID algorithm offers robust control from a relatively simple algorithm, which has contributed its continued popularity. The control output of the PID is proportional to the error input, meaning that the PID is not ideally suited to the control of complex non-linear systems. However the algorithm dynamics can be altered through adaption of the gains of the three elements – K_p , T_r and T_d . In the gain scheduling scheme which is to be applied in this paper these three gain are adapted according to the operating parameters of the converter. This popular approach allows for the PID to achieve a more optimal level of control of the non-linear system.

The boost converter circuit is a non-linear second order system. Whilst the ANFIS –PID is designed to counteract the non-linearity of the system, it is possible to further simplify the system response. If the current through the inductor is more closely controlled then this has the effect of reducing the system response to a first order type. In doing this the system becomes easier to control and a better response can be achieved. In the system shown in Fig 1, the inner PI controller is used to more closely control the current through the inductor, thus reducing the control complexity of the output and helping to further improve the system output.

C. Maximum Power Point Tracking

The characteristics of PV cells are influenced by both the solar radiation and the temperature. In order to utilize the maximum available power from the PV cell a MPPT algorithm is included in the system. There are a number of different algorithms available for MPPT with two of the most widely used being the Perturb and Observe (P&O) and Incremental Conductance (INC). Both of these techniques are examples of “hill-climbing” algorithms. In both cases the algorithm works by slightly increasing or decreasing the operating voltage of the PV cell. By observing whether this causes an increase or decrease in the power it is possible to track the MPPT. These techniques are popular as they are relatively simple to implement, exhibit good efficiency and are technology independent. A review of these algorithms applied to grid-tied PV systems is conducted in [10], showing that the INC algorithm yields the best results. A variation of the INC algorithm introduced in [11], which features adaptive step sizes has been used in this application. Using this

variation of the INC algorithm is shown to give reductions in oscillations in steady state operation whilst also benefitting from improved dynamic response. The flow for the variable step size INC algorithm used is shown in Fig 2, where $S(k)$ is the variable step size which is calculated as in Eq. (6) with N being a scaling coefficient which is tuned for the device.

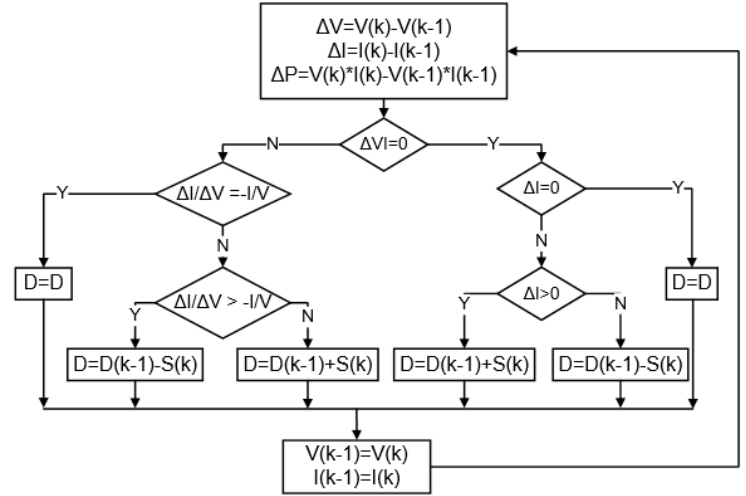


Fig 2 Incremental Conductance Algorithm with Variable Step Size

$$S(k) = N * \left| \frac{\delta P}{\delta V} \right| \quad (6)$$

III. ANFIS ALGORITHM

Whilst Fuzzy logic and Neural Networks are both well established AI techniques which have been widely applied to the field of control, both have drawbacks. In the case of fuzzy logic, whilst it is excellent at making decisions, this is the inability to learn. Conversely, neural networks don't share this ability to make decisions but do possess the ability to learn. Combining the benefits of both types of system, new hybrid Neuro-Fuzzy systems have emerged. One popular example of such a system is the ANFIS architecture. The ANFIS architecture was originally introduced in 1993 by J-S Jang [8] and has since been used in a variety of control systems such as [12] and [13].

The ANFIS architecture is comprised of a five layer feed forward neural network, as shown in shown in Fig 3. This approach means that the learning capabilities of the Neural Network are preserved but the desirable decision making abilities of the fuzzy inference system are added. When fully trained, this network exhibits behaviour which is analogous to a Sugeno type fuzzy inference system. The Sugeno inference engine is a universal approximator which is capable of approximating non-linear functions. The ability of the Sugeno model to approximate is only limited by the number of rules. With a limitless number of rules, the system will be capable of approximating even the most complex non-linear function with great accuracy. As the ANFIS network is analogous to the Sugeno system, this non-linear function approximation

capability is inherited. When coupled with the relative ease of training, thanks to its learning ability, this makes the ANFIS an attractive option for the control of non-linear plant such as switch mode DC-DC converters.

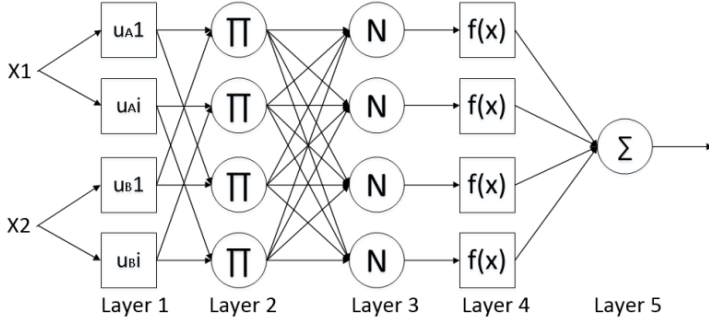


Fig 3 ANFIS Architecture

In the rest of this section the five layers which comprise the ANFIS network are discussed in more detail.

A. Layer 1 - Fuzzification

The first stage of the ANFIS model is to convert the crisp input values into fuzzy number sets in much the same way that a fuzzy logic system would. The output of the membership functions is given in Eq (7) where μ_A and μ_B represents the membership groups (e.g. small, large, big etc).

$$\begin{aligned} O_{1,i} &= \mu_{A_i}(x_1) \\ O_{1,i} &= \mu_{B_i}(x_2) \end{aligned} \quad (7)$$

B. Layer 2 and 3 – Firing Strength

Once the fuzzification of the inputs has been performed, the next stage in the system is to calculate the firing strength of the rule. This is done in two stages - firstly the values of μ_A and μ_B are multiplied together as shown in Eq (8). This is performed in stage two; stage three then performs normalization of the multiplied values, putting the values in a predefined range as shown in Eq. (9).

$$O_{2,i} = w_i = \mu_{A_i}(x_1)\mu_{B_i}(x_2) \quad (8)$$

$$O_{3,i} = \bar{w}_i = \frac{w_i}{w_1 + w_2} \quad (9)$$

C. Layer 4 – Consequence Parameters

The next stage in the ANFIS is to calculate the consequence parameters. In a traditional fuzzy system, consisting of IF THEN rules, this part of the algorithm is equivalent to the THEN part of rule. The output of this layer is shown in Eq. (10). The output of this stage is effectively the same as the output rule for Takagi-Sugeno type fuzzy inference engine.

$$O_{4,i} = \bar{w}_i f_i = \bar{w}_i (px_1 + qx_2 + r) \quad (10)$$

D. Layer 5 – De-fuzzification

The final stage of the ANFIS algorithm is to convert the fuzzy logic sets into a crisp output which can be used by the external PID controller. This stage takes the simple form of a summation of all the rule outputs, meaning the output of this stage, and the whole ANFIS algorithm, is as given in Eq. (11).

$$O_5 = \frac{w_i f_i}{w_i} \quad (11)$$

E. Training the ANFIS

In order for the ANFIS to work in any given application it is necessary to firstly perform training on the network. This training stage is much the same as the training stage of a Neural Network. The ANFIS is presented with a set of training data, which consists of a number of inputs and the expected outputs. Training is then completed to minimize the error between the actual output and the expected output. There are two adaptive nodes which are trained in the ANFIS – the fuzzy logic sets in layer one and the consequent parameters in the fourth layer. This training is typically supervised by a hybrid training algorithm which features a forward pass and a backward pass phase. The forward pass phase of this training the node outputs are fed forward until layer four. In this stage the consequent parameters in layer four are then tuned using the least squares estimation method. The least squares method is designed to minimize the sum of the squared error of the system output. In the back pass phase of the hybrid algorithm the membership sets in layer one are tuned. In this part of the training the error signals are propagated backwards from the output and the membership parameters are optimized using the gradient descent algorithm. The gradient descent algorithm, which is also typically employed in neural network training, finds the minimum error by moving the membership parameters a distance which is proportional to the functions gradient at the given point. This movement is performed in every training iteration until the output error is sufficiently minimized.

The training of ANFIS based systems can be easily achieved using the Matlab modeling tool. The Neruo-Fuzzy plug-in allows for automation of the training using the hybrid learning algorithm. This plug-in also allows for careful selection of the membership functions, with options available for the number and type of membership functions. Once the training stage has been completed it is also easily possible to validate the data using this tool.

IV. SYSTEM MODELING

A. Boost Converter Circuit Model

In order to optimize the performance of the boost circuit the values of the output capacitor and the inductor must be carefully selected. The inductor used in this circuit is chosen as 22uH which gives a ripple current of 550mA according to Eq. (3). The output capacitor is selected as 100uF which gives a maximum ripple voltage of 25 mV according to Eq. (4). The boost converter circuit will be modeled using the PSim simulation environment. PSim is spice based simulation

tool which is targeted at the design of power electronics and control systems.

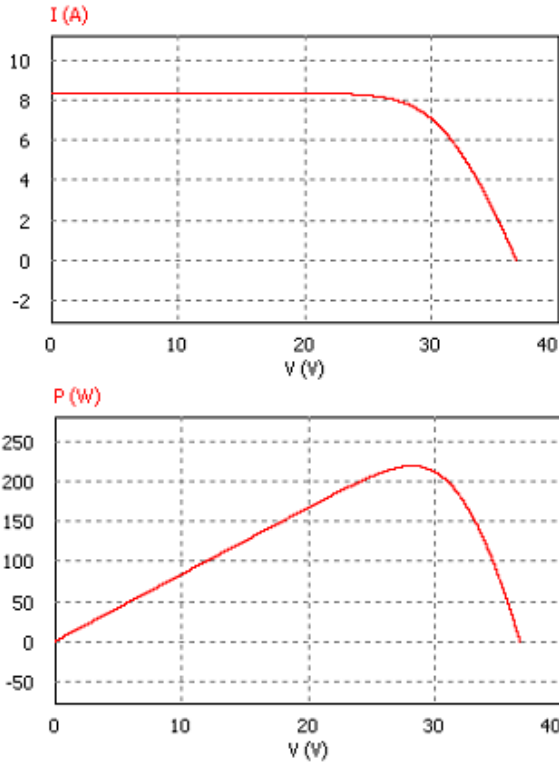


Fig 4 BP3230N IV Curve obtained in PSim

B. Solar PV Cell and MPPT Tracking

The modelling of solar photovoltaic cells can prove to be a complex issue. However the use of the PSim software suite allows for this task to be simplified greatly. PSim supplies a generic solar cell modelling tool which can be used to create a custom model of any given cell using a few parameters from the data sheet, such as rated power, series resistance, shunt resistance and short circuit current. This tool also allows for the generation of power curves at any operating point to allow for the model to be validated before it is imported into the PSim schematic for use in simulations. The solar PV cell model developed for this simulation and the obtained IV curves at MPP are shown in Fig 4.

The implementation of the variable step size INC algorithm is also achieved using the PSim software suite. Whilst this task could have been achieved using Matlab, the use of PSim at this stage allows for closer consideration of a potential final hardware target. The PSim tool offers compilation and simulation of functions written using C code. This means that the code can be developed and simulated at the system modelling stage and easily imported to a microcontroller hardware target, reducing development time.

C. Tuning the PID Controller

Prior to the ANFIS being trained it is first necessary to create the training data. In order to gather this data the PID controllers need to be tuned under a number of operating

scenarios. There are a number of different methods available for the tuning of PID controllers, from traditional methods

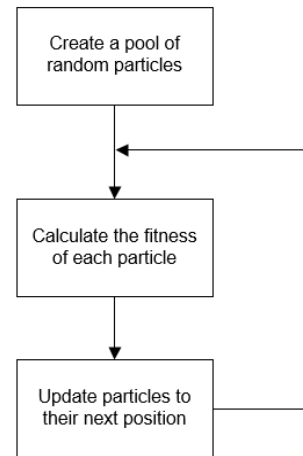


Fig 5 PSO Algorithm Flow

like Ziegler-Nichols to modern approaches using artificial intelligence (AI) algorithms and online training solutions.

The training algorithm selected here is known as Particle Swarm Optimization (PSO) and is an AI based approach which mimics the behaviour of flocking birds. This algorithm has proven to be a popular solution to the tuning of PID controllers [14][15] and benefits from being both effective and relatively simple to implement.

The main flow of the PSO algorithm is shown in Fig 5, illustrating how the algorithm is used to move potential solutions in the problem search space. A crucial component of the PSO algorithm is the particle, which consists of three separate parts – the current solution, the velocity and the personal best fit. The current solution stores the data which is to be applied to the problem, whilst the personal best fit stores the value of the particle which has yielded the best result so far. The velocity is a key parameter in controlling how the problem space is searched as it controls how fast and in which direction the particle is accelerated. This value is initially set to a random value but is iteratively updated. Each update has the effect of pulling the particle towards either a global or personal best fit. The equation for updating the velocity of the particle is shown in Eq. (12), with V_n being current velocity, X_n being the current solution, GX_n the global best fit, PX_n the personal best fit and w a constant known as inertia weight.

$$V_{N+1} = \omega \cdot V_N + (C_1 \cdot \text{Rand} \cdot (GX_N - X_N)) + (C_2 \cdot \text{Rand} \cdot (PX_N - X_N)) \quad (12)$$

The social and cognitive coefficients, shown by C_1 and C_2 respectively, have the effect of pulling the particle towards either its personal best fit (cognitive) or the global best fit (social). The values of these two therefore have the effect of promoting convergence to either the personal or global best fit. To achieve optimal performance of the algorithm the

cognitive value is linearly decreased whilst the social value is linearly increased.

Using this algorithm the PI(D) controllers are tuned, firstly the inner current PI controller and then the voltage PID. The two PI(D) controllers are both trained under a number of different steady state loads and with a number of load transients to give a more complete set of training data. The optimized gain values which are found during the tuning are then used to form the training data of the gain scheduling ANFIS controller. The PSO algorithm is modeled in the popular Matlab simulation tool using co-simulation with the boost converter developed in PSim through the Sim-Coupler plug-in.

The ANFIS training is also completed using Matlab once the PI(D) tuning has been completed. The ANFIS is designed to have two inputs – with one being the load current and one being changes in load current. There will be seven trapezoidal membership functions for each of these two inputs as this has shown to give a good approximation of the training data. The training is automated in Matlab using the Neuro-Fuzzy plug-in, which uses the hybrid learning algorithm previously discussed to tune the performance of the ANFIS controller. The fully trained ANFIS model is also simple to create in Matlab, whilst the Sim-Coupler module again allows for interaction with the PI(D) controllers and boost circuitry which is modeled in PSim.

V. SIMULATION RESULTS

In order to demonstrate the performance of the ANIFS-PID controlled boost converter PV system a number of simulations are carried out to ascertain the system performance, under both steady state and dynamic conditions. There are a total of four simulations which are carried out to illustrate the performance of the system. The first set of simulations carried out is intended to measure the steady state performance of the system under both heavy (4A) and light loads(1A). The results of these tests are shown in Fig 6 and Fig 7, showing that in both instances there is no steady state error. The ripple voltage for the light load is measured at just

4mV whilst the ripple voltage for the heavy load is 16mV.

The next simulation is carried out to show the dynamic performance of the system when there is a reduction in output load demand. The system performance with a one amp decrease in current is shown in Fig 5. This simulation demonstrates that under these dynamic operating conditions the maximum absolute error is measured at just 257 mV, which equates to an output error of just 0.48%.

The final simulation shows the dynamic performance of the system when there is a step increase in output load demand. Fig 5 shows the output of the system with a with a one amp step increase in current. This shows that under these dynamic conditions there is a maximum absolute error of just 230 mV, which equates to an output error of just 0.53% compared to the reference voltage of 48VDC.

The results of the simulations carried out and shown in figures Fig 6 to Fig 5 are also summarized and presented in TABLE I below.

TABLE I
SUMMAR OF SIMULATION RESULTS

| Steady State Tests | |
|---------------------------|------|
| 4A Load | |
| Steady State Error (%) | 0 |
| Peak to Peak Ripple (mV) | 16 |
| 1A Load | |
| Steady State Error (%) | 0 |
| Peak to Peak Ripple (mV) | 4 |
| Dynamic Test | |
| 1A Step Decrease in Load | |
| Max. Error (mV) | 230 |
| Max. Error (%) | 0.48 |
| 1 A Step Increase in Load | |
| Max. Error (mV) | 257 |
| Max. Error (%) | 0.53 |

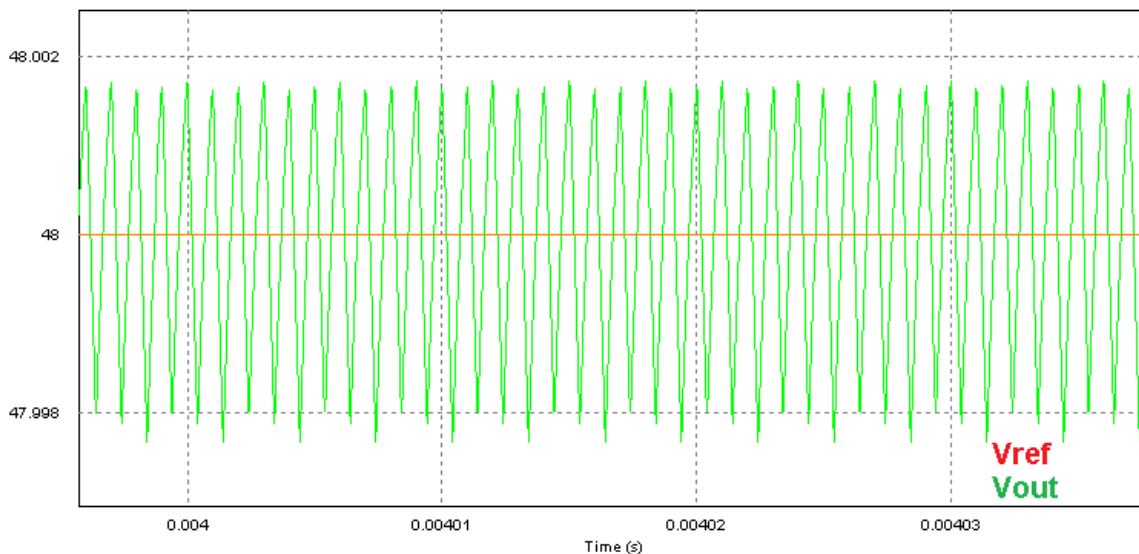


Fig 6 Boost Converter Output with 1A Load

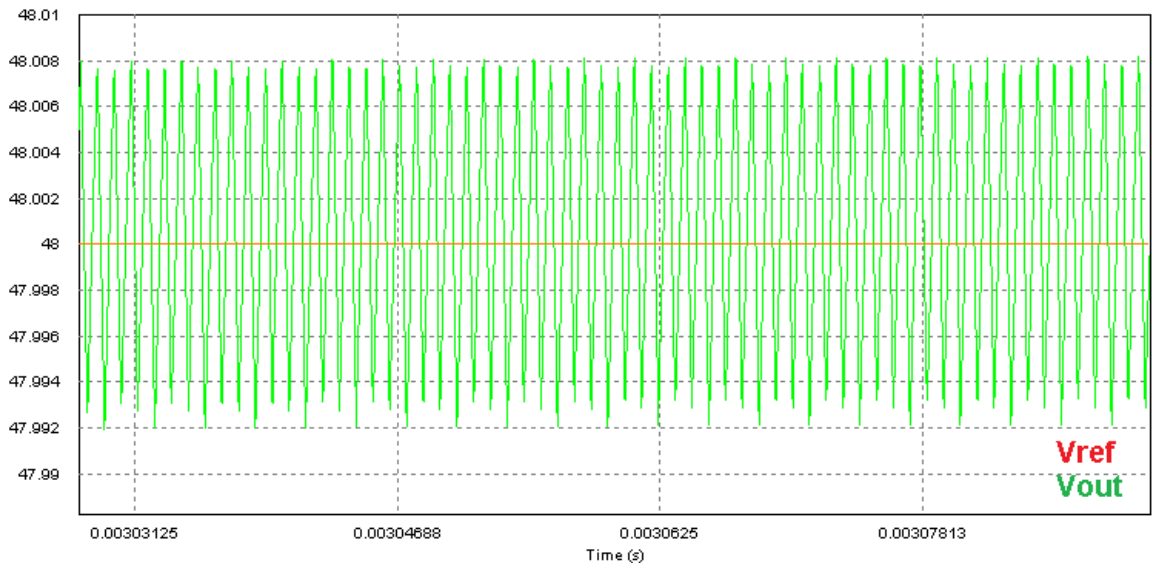


Fig 7 Boost Converter output with a 4A load

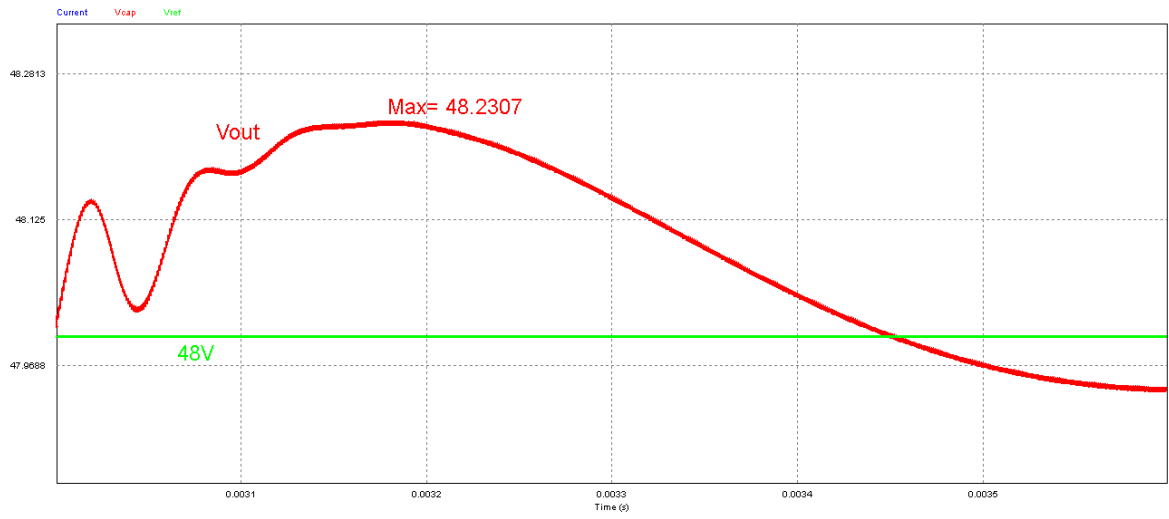


Fig 8 Boost converter output with a 1A step decrease in current

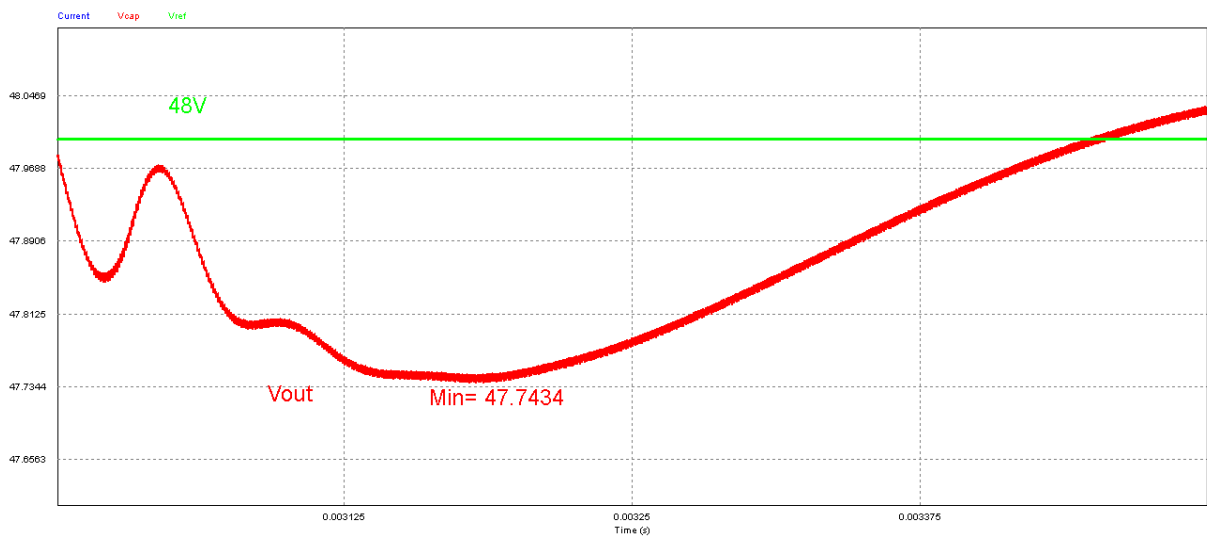


Fig 9 Boost converter output with a 1A step increase in current

VI. CONCLUSIONS

The simulation results of the new gain scheduling technique in the developed solar PV application indicate that this methodology can achieve a good level of control for a non-linear system. The boost converter to which the ANFIS-PID controller is applied exhibits excellent behaviour under both steady state and dynamic operating conditions. In the steady state operation the power converter exhibits no steady state error whilst under heavy loads the peak to peak ripple voltage is just 16mV. Under dynamic operating conditions the maximum absolute error observed is less than 0.6 %. This performance characteristic can be especially important in the PV cell fed boost converter application to which this is applied. This level of voltage deviation allows for the system to be fed into an inverter without any instability issues being introduced, meaning it can be tied to the grid.

As well as being capable of delivering a good level of control on a non-linear plant, the design flow for the controller is simplified by implementing the PSO algorithm in the training phase. The PSO algorithm is well known, efficient and simple to implement, meaning that the training data can be gathered easily. Leveraging this algorithm in this stage also means that the tuning of the PID is an automated process. The availability of the Matlab Neuro-Fuzzy tool makes the design flow more desirable still, as this means that ANFIS training can be achieved simply. This shows that this methodology can offer a fast modelling and prototyping solution.

REFERENCES

- [1] N. Minorsky, "Directional stability of auto-matically steered bodies." *J. Amer. SOC. of Naval Engineers*, Vol. 34, pp. 280-309, May 1922
- [2] L. Martinez-Salamero, G. Garcia, M.Orellana, C.Lahore and B.Estivals, "Start-Up Control and Voltage Regulation in a Boost Converter Under Sliding-Mode Operation." *IEEE Transactions on Industrial Electronics*, Vol. 60, Iss. 10, pp. 4637-4649, 2012
- [3] S. Oucheria and L. Guo, "PWM-Based Adaptive Sliding-Mode Control for Boost DC-DC Converters." *IEEE Transactions on Industrial Electronics*, Vol. 60, Iss. 8, pp. 3291-3294, 2012
- [4] A. Rubaai, M.J. Castro-Sitiriche and A.R Ofoli, "DSP-Based Laboratory Implementation of Hybrid Fuzzy-PID Controller Using Genetic Optimization for High-Performance Motor Drives." *IEEE Transactions on Industry Applications*, Vol. 60, Iss. 6, pp. 1977 – 1986, 2008
- [5] H. Yu, T. Chen and C. Liu, "Adaptive Fuzzy Logic Proportional-Integral-Derivative Control for a Miniature Autofocus Voice Coil Motor Actuator with Retaining Force" *IEEE Transactions on Magnetics*, Vol. 50, Iss. 11, 2014
- [6] M. Elshaer, A.Mohamed and O.A. Mohammed, "Smart Optimal Control of DC-DC Boost Converter for Intelligent PV Systems" *16th International Conference on Intelligent System Application to Power Systems*, pp. 1-6, 2011
- [7] V.Vindhya and V. Reddy, "PID-Fuzzy Logic Hybrid Controller for a Digitally Controlled DC-DC Converter." *International Conference on Green Computing, Communication and Conservation of Energy*, pp. 362-366, 2013
- [8] J-S, Jang, "ANFIS: Adaptive-Network-Based Fuzzy Inference System." *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 23, Iss 3, pp. 665-685, 1993
- [9] P. Palensky and D. Dietrich, "Demand Side Management: Demand Response, Intelligent Energy Systems and Smart Loads." *IEEE Transactions on Industrial Informatics*, Vol. 7, Iss. 3, pp 381 – 388, 2011
- [10] R.A. Mastromauro, M. liserre and A. Dell'Aquila, "Control Issues in Single-Stage Photovoltaic Systems: MPPT, Current and Voltage Control", *IEEE Transactions on Industrial Informatics*, Vol. 8, Iss. 2, pp. 241 – 254, 2012
- [11] F. Liu, S. Duan, F. Liu, B. Liu and Y. Kang, "A Variable Step-Size INC MPPT Method for PV Systems", *IEEE Transactions on Industrial Electronics*, Vol. 55, Iss. 7, pp 2622 – 2628, 2008
- [12] H. Abu-Rub, A. Iqbal, S. Moin Ahmed, F.Z. Peng, Y. Li and G. Baoming, "Quasi-Z-Source Inverter-Based Photovoltaic Generation System with Maximum Power Point Tracking Control using ANFIS" *IEEE Transactions on Sustainable Energy*, vol. 4, iss. 1, pp 11-20, 2013.
- [13] P. Garcia, C.A. Garica, L.M. Fernandez, F. Lorens and F.Jurado. "ANFIS-Based Control of a Grid Connected Hybrid System Integrating Renewable Energies, Hydrogen and Batteries." *IEEE Transactions on Industrial Informatics*, vol. 10, iss. 2, pp 1107-1117, 2013
- [14] X. Li, M.Chen and Y.Tsutomu, "A method of searching PID controller's optimized coefficients for Buck converter using particle swarm optimization." *IEEE Electric Power and Energy Conference*, pp.238, 2013
- [15] J. Femmy Nirmal and D.Jeraldin Auxillia, "Adaptive PSO based tuning of PID controller for an Automatic Voltage Regulator system." *International Conference on Circuits, Power and Computing Technologies*, pp.661, March 2013