



Missouri University of Science and Technology
Scholars' Mine

Electrical and Computer Engineering Faculty
Research & Creative Works

Electrical and Computer Engineering

01 Jan 2006

Comparison of Two Optimal Control Strategies for a Grid Independent Photovoltaic System

Richard L. Welch

Ganesh K. Venayagamoorthy
Missouri University of Science and Technology

Follow this and additional works at: https://scholarsmine.mst.edu/ele_comeng_facwork

 Part of the [Electrical and Computer Engineering Commons](#)

Recommended Citation

R. L. Welch and G. K. Venayagamoorthy, "Comparison of Two Optimal Control Strategies for a Grid Independent Photovoltaic System," *Conference Record of the 41st IAS Annual Meeting of the Industry Applications Conference, 2006*, Institute of Electrical and Electronics Engineers (IEEE), Jan 2006. The definitive version is available at <https://doi.org/10.1109/IAS.2006.256673>

This Article - Conference proceedings is brought to you for free and open access by Scholars' Mine. It has been accepted for inclusion in Electrical and Computer Engineering Faculty Research & Creative Works by an authorized administrator of Scholars' Mine. This work is protected by U. S. Copyright Law. Unauthorized use including reproduction for redistribution requires the permission of the copyright holder. For more information, please contact scholarsmine@mst.edu.

Comparison of Two Optimal Control Strategies for a Grid Independent Photovoltaic System

Richard L. Welch, *Student Member*, IEEE and Ganesh K. Venayagamoorthy, *Senior Member*, IEEE
Real-Time Power and Intelligent Systems Laboratory
Department of Electrical and Computer Engineering
University of Missouri-Rolla
Rolla, MO 65409, USA
rwelch@ieee.org and gkumar@ieee.org

Abstract—This paper presents two optimal control strategies for a grid independent photovoltaic system consisting of a PV collector array, a storage battery, and loads (critical and non-critical loads). The first strategy is based on Action Dependent Heuristic Dynamic Programming (ADHDP), a model-free adaptive critic design (ACD) technique which optimizes the control performance based on a utility function. ADHDP critic network is used in a PV system simulation study to train an action neural network (optimal neurocontroller) to provide optimal control for varying PV system output energy and loadings. The second optimal control strategy is based on a fuzzy logic controller with its membership functions optimized using the particle swarm optimization. The emphasis of the optimal controllers is primarily to supply the critical base load at all times, thus requiring sufficient stored energy during times of less or no solar insolation. Simulation results are presented to compare the performance of the proposed optimal controllers with the conventional priority control scheme. Results show that the ADHDP based controller performs better than the optimized fuzzy controller, and the optimized fuzzy controller performs better than the standard PV-priority controller.

Keywords—adaptive critic designs, battery storage, energy dispatch and management, fuzzy logic, neural networks, optimal control, particle swarm optimization, photovoltaic system, solar

I. INTRODUCTION

With the continuing rise in the prices of fossil fuels and falling costs of alternative energy sources such as solar and wind power, alternative energy sources are an intriguing way to reduce energy costs for heating, cooling, and meeting the general electrical needs of a residence or a facility. Many alternative energy sources are available, such as wind, solar (both direct heating of water and electrical generation via photovoltaic (PV) arrays), and hydroelectric sources. Of these sources, PV arrays are preferred because of low maintenance and high availability, as compared to wind or hydroelectric, and long life.

The price of photovoltaic (PV) panels has fallen dramatically over the past 30 years [1] as improvements in technology and fabrication have been made. The large increase in utility rates over the last few years is making the price of alternative energy even more appealing. Despite the

fall in prices of PV systems, they are still quite expensive. The payback time for a typical PV system can be 30 years or more, depending on the size of the installation, type of equipment used, and the solar radiation available. Fortunately, the life of the PV arrays themselves is around 30 years. And since they have very few or no moving parts, maintenance requirements are very low.

It is possible to reduce the overall costs of the PV system with an efficient control scheme determining when and how much of the electrical loads are to be supplied. This will allow for more efficient use of the PV system components, and thus enable the designer to design a system with smaller and less costly PV arrays and batteries while still allowing the PV system to provide adequate coverage to the base (or critical) load.

Traditionally, the control scheme that is used for PV systems is usually called a “PV-priority” control scheme [2]. In this control scheme, the controller attempts to power the entire load (both critical and non-critical loads). If there is any excess electrical energy, it charges the battery. When there is insufficient PV energy to power the loads, then it will draw energy from the battery to meet as much as possible of the load demand.

In order to improve upon the PV-priority scheme, an optimal controller can be designed such that the non-critical load is only powered when there is a sufficient amount of energy from the PV arrays. In this way, an optimal controller can conserve battery energy during times of reduced solar radiation so that there will be energy available to power the critical load whenever required. An example of a critical load would be the refrigeration of vaccines and medication in remote locations without access to a reliable electrical grid.

Alternative approaches to PV controllers using Q-learning, dynamic programming and fuzzy logic have been previously reported [2, 3, 4]. In this paper, two optimal PV controller strategies are presented. The first uses an Adaptive Critic Designs (ACDs) [5] approach, while the second is based on fuzzy logic, optimized using Particle Swarm Optimization (PSO) [6]. The objectives of optimal control are threefold: 1) to maximize or fully dispatch the required energy to the critical loads at all times, 2) to dispatch energy to charge the

The support from the National Science Foundation under the CAREER grant ECS # 0348221 is gratefully acknowledged by the authors.

battery to enable it to power the critical load when the collector cannot meet the critical load demand, and 3) to dispatch energy to the non-critical loads while not compromising on the first two objectives.

Adaptive critic designs are based on the combined concepts of approximate dynamic programming and reinforcement learning. Neural networks are used to implement adaptive critic architectures. The Action Dependent Heuristic Dynamic Programming (ADHDP) approach of the ACD family is used for the ACD optimal PV controller design [7, 8].

The fuzzy logic PV controller presented in this paper is the Mamdani fuzzy logic controller. This type of controller contains a fuzzification phase, an inference engine, and a defuzzification phase. The PSO algorithm is used to optimize the membership functions of the fuzzy logic controller so that its performance increased.

Section II of this paper presents the PV model studied in this paper. Section III describes the standard PV-priority controller. Section IV describes the ADHDP optimal controller design. Section V presents the optimized fuzzy logic controller. Section VI presents the evaluation and comparison of the standard PV-priority controller, ADHDP optimal PV controller and the optimal fuzzy controller. These controllers are compared against each other using Typical Meteorological Year (TMY) data from Caribou, Maine [9]. Finally, the conclusion is given in Section VII.

II. PHOTOVOLTAIC SYSTEM MODEL

A PV system is simulated for this study. For this simulation, a model of each component of the PV system is designed and built in Matlab. The complete PV system consists of the PV array (solar cells), maximum power point tracker, battery, critical and non-critical loads, battery charge controller, inverter, and controller. Since the emphasis is on optimizing the controller performance, it is assumed that the efficiency of some of the components (inverter, battery charge controller, and maximum power point tracker) is 100%. The critical load consists of loads that should never be dropped (such as refrigeration and/or radio communication), and the non-critical load contains items which are non-essential (e.g. television).

The solar cells are simulated with 11% efficiency. Normally, the efficiency of PV panels range from 6% to up to 30% (with the high efficiency panels being used primarily in space applications because of their light weight and their ability to operate in higher radiation environments). A rough equivalent to the PV arrays simulated in this paper is an array of 8 Kyocera KC200GT panels. These panels are over 16% efficient and each panel outputs 200W during optimal conditions [10]. The minimum charge for the battery is taken to be 30% (this is consistent with standard deep cycle lead-acid batteries).

The PV system arrangement is shown in Fig. 1. In this diagram, energy flow is in the direction of the arrows. During this simulation, if the energy from the PV array is ever greater than the sum of the loads (both critical and non-critical) and there is enough energy to completely charge the battery, then

optimal control is not used. Instead, all loads are powered and the battery is completely charged. This case occurs infrequently.

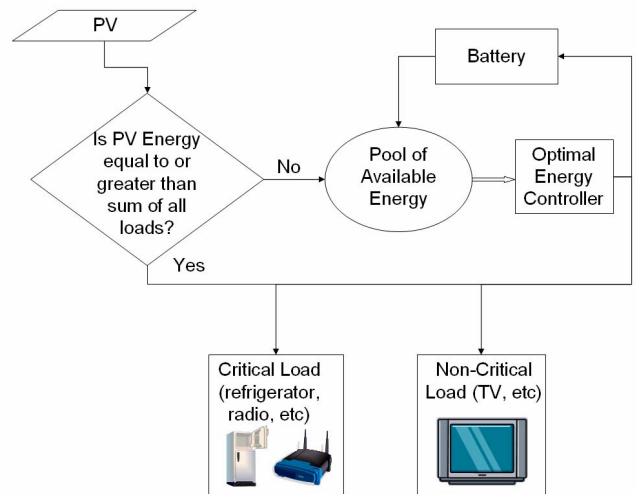


Figure 1. PV system model (energy flow shown by arrow direction), shown with optimal controller bypass (for cases with over abundance of energy).

III. PV-PRIORITY CONTROLLER

The standard controller called the “PV-Priority” controller is a simple controller which always tries to meet the load demand (the critical and then the non-critical) before charging the battery. At any one time, if there is not enough energy from the PV array to supply the loads, then the balance is drawn from the battery. If instead there is an excess, then whatever is left over after supplying the loads is dispatched to the battery. In this way, the controller will attempt to power all loads and charge the battery as best it can, without any considerations given to the time varying states of the system.

This controller works well when there is sufficient PV energy. However, when there is not sufficient PV energy, then the battery will not be fully recharged and the loads will be dropped. The weather and user loads are stochastic in nature; therefore there is no one definitive model at all times. Thus, it makes sense to look at intelligent model-free learning methods of controlling such a system.

IV. ADHDP OPTIMAL CONTROLLER

Intelligent controllers based on adaptive critic design can be well suited to areas without abundant sunlight. Adaptive critic designs (ACDs) utilize neural networks and are capable of optimization over time in conditions of noise and uncertainty. A family of ACDs was proposed by Werbos [5] as a new optimization technique, combining the concepts of reinforcement learning and approximate dynamic programming. With ACDs, for a given series of control actions that must be taken sequentially (and not knowing the effect of these actions until the end of the sequence), it is possible to design an optimal controller using the traditional supervised learning neural network.

The adaptive critic method determines an optimal control policy for a system by adapting two neural networks: an *action* network and a *critic* network. The action network is responsible for controlling the system actions, while the critic network is responsible for critiquing the action network over time to optimize it. The critic network learns to optimize the action network by approximating the Hamilton-Jacobi-Bellman equation associated with optimal control theory.

This process starts with a non-optimal or sub-optimal, arbitrarily chosen control by the action network. The critic network then guides the action network toward an optimal solution at each successive adaptation. During the adaptations, neither of the networks needs any “information” of an optimal trajectory, only the desired cost needs to be known. Furthermore, this method determines optimal control policy for the entire range of initial conditions. Unlike other neural-controllers, it needs no external training [7].

The design ladder of ACDs includes three basic implementations, in the order of increasing power and complexity. These include: Heuristic Dynamic Programming (HDP), Dual Heuristic Programming (DHP) and Globalized Dual Heuristic Programming (GDHP). The interrelationships between members of the ACD family have been generalized and explained in [8]. In this paper, an Action Dependent HDP (ADHDP) approach is chosen for the design of a PV optimal controller. Action dependent adaptive critic designs do not need system models to develop the optimal control policy (action network output). A block diagram of the ADHDP PV controller (action network) is shown below in Fig. 2.

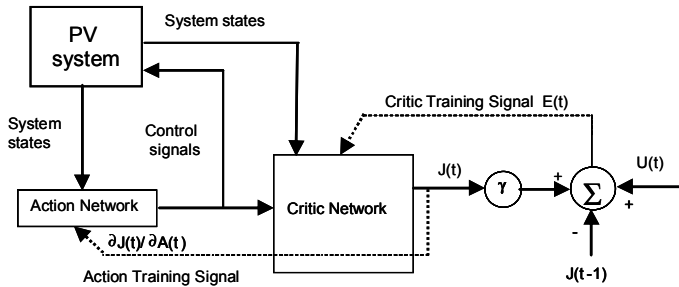


Figure 2. Structure of the ADHDP based optimal PV controller design.

For this controller design, the utility function $U(t)$ in Fig. 2 is used to guide the critic network in training the action network and is given in (1).

$$U(t) = (30 / 23) * \text{abs}(1 - (ECL / CL)) + (15 / 23) * \text{abs}(1 - (EB / ((MBC - CBC) + M * MBC))) + (13 / 23) * \text{abs}(1 - (ENCL / (NCL + M * MNCL))) \quad (1)$$

Where:

$ECL =$ Energy dispatched to the critical load

$CL =$ Critical load

$EB =$ Energy dispatched to the battery

$MBC =$ Maximum battery charge

$CBC =$ Current battery charge

$ENCL =$ Energy dispatched to the non-critical load

$NCL =$ Non critical load

$MNCL =$ Maximum non-critical load

$M =$ Multiplier (used to ensure divisor is non-zero; for this experiment, a value of 0.1 was used)

In this case, the optimal ADHDP PV controller is developed to optimally supply energy to certain loads and/or charge the battery [11]. In this way, if there is a lack of solar energy available later on, then the battery charge can be used to power the loads later.

The ADHDP controller takes as inputs the following signals:

- Solar energy from the PV array as a fraction of total possible energy from the PV array
- Critical load as a fraction of maximum critical load
- Non-critical load as a fraction of maximum non-critical load
- Current battery charge as a fraction of total charge.

The ADHDP controller outputs are the following:

- Energy dispatched to the critical load
- Energy dispatched to the non-critical load
- Energy dispatched to the battery (which can be positive or negative, depending on whether the battery is being charged or being used as a source)

Additionally, the action network’s outputs are checked to ensure that no more energy is dispatched than is available at the inputs. This is accomplished by performing the following series of actions immediately after obtaining the outputs from the action network:

- Verify that the energy dispatched to each of the loads does not exceed the load demand and isn’t negative. Also ensure that the energy to the battery is not higher than the energy collected by the PV arrays.
- Verify that the battery is not being overcharged or over depleted.
- The outputs, including the energy dispatched to the battery if it is being charged, are scaled by the ratio of energy inputs to outputs.
- Another round of checks is made on the outputs in order to be certain that they are not greater than the load or less than zero.

Finally, any difference in energy inputs and outputs is added to the energy dispatched to the battery, in case this balance is changed with the scaling or previous boundary checks. Also during this step, the energy to the battery is checked to make sure that the battery is not overcharged. More details on the ADHDP PV controller are provided in [11].

V. OPTIMAL FUZZY LOGIC CONTROLLER

The second optimal PV controller studied is a Mamdani type fuzzy controller. The inputs and outputs of this controller are very similar to the ADHDP controller discussed earlier.

Fuzzy controller inputs are the following:

- Solar energy from the PV array as a fraction of total possible energy from the PV array
- Current battery charge as a fraction of total charge
- Combined load as a fraction of maximum total load.

The fuzzy controller outputs are identical to the ADHDP PV controller outputs.

A. Fuzzy Logic Controller

In any fuzzy logic system, the system takes a value and first passes it through a fuzzification process. Then it is processed by an inference engine (or fuzzy rule set). Finally, it goes through a defuzzification process. This process is described in more detail in the following subsections, and an overall block diagram for the fuzzy logic controller is shown below in Fig. 3.

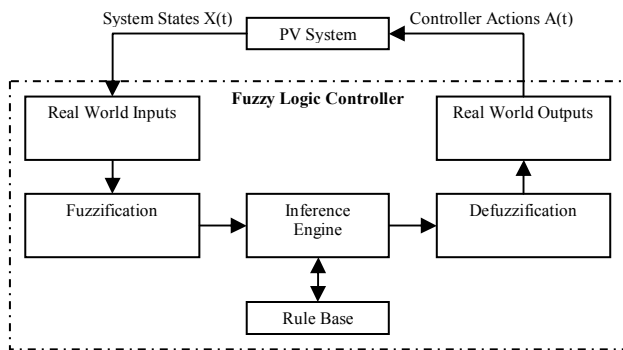


Figure 3. Figure showing the fuzzy logic controller process.

1) Fuzzification

Fuzzification is a process that takes a real-world value and maps it to a fuzzy set based on a membership function. The membership functions in this case are composed of many triangles but can be represented using other functions as well. An example of a membership function is below:

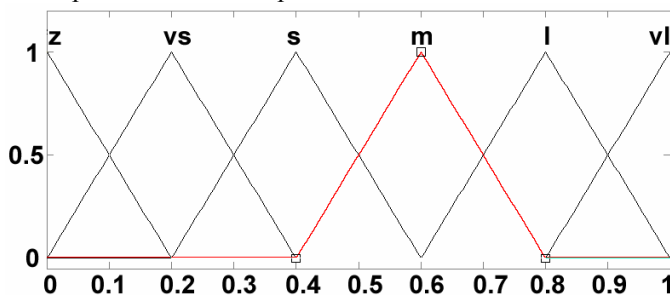


Figure 4. Membership functions example (for the “Load” input). The highlighted function is responsible for assigning the membership to the “M” fuzzy value (“M” represents “medium”).

When an input is being mapped to a fuzzy value, it is assigned a degree of membership (usually denoted by μ) for each membership function. For example, using the membership function from the previous figure (Fig. 4), the value of “Load” may take on fuzzy values “Z”, “VS”, “S”, “M”, “L”, and “VL” (each value representing “Zero”, “Very Small”, “Small”, “Medium”, “Large”, and “Very Large”, respectively). In this case, if the load is 0.75, then the degrees of membership for each fuzzy value for the load are:

$$\mu(z)=0.0, \quad \mu(vs)=0.0, \quad \mu(s)=0.0, \quad \mu(m)=0.25, \quad \mu(l)=0.75, \quad \mu(vl)=0.0$$

In the design of the PV controller, 3 input variables are used: PV Energy, Battery Charge, and Load (the sum of both the critical and non-critical load). All three variables can take on the above listed values.

2) Inference engine

Once the degrees of membership are determined for a given input using the membership functions, these values are used to fire any number of the rules from the fuzzy rule set. The inference engine (or fuzzy rules set) is just a set of rules which maps input fuzzy values to output fuzzy values. Given an input value (or values), the fuzzy rule returns an output value for the specified values. For example, if the rule has a rule that says “If Load is Large, then set the Energy to Critical Load to Large”, then the output fuzzy variable “Energy to Critical Load” is set to “Large” if the input value “Load” is “Large”. For this simulation, 216 rules are used. This set of rules cover all possibilities of inputs and outputs.

3) Defuzzification

Defuzzification is a process that takes a fuzzy value and maps it to a real-world value. Once an input value is fuzzified and passes through the inference engine, its real world output is found by using this defuzzification process. In this case, 3 output variables are used: Energy to Critical Load, Energy to Non-Critical Load, and Energy to the Battery. In the first two cases, the variables can take on the following fuzzy values: “Z”, “VS”, “S”, “M”, “L”, “VL” (each was described earlier); for the last case, it can take on fuzzy values “LD”, “SD”, “Z”, “SC”, “LC” (meaning “Large Discharge”, “Small Discharge”, “Zero”, “Small Charge”, and “Large Charge”).

In order to do this, all of the values obtained from the fuzzy rule set outputs are weighted according to the weights of their corresponding inputs. Once these values are found, there are a variety of methods for resolving ambiguities among output values. For this controller, the *centroid* method is used. This method finds the center of mass of the weighted outputs from the fuzzy rule set and returns this position as the real world output. Normally (as in this case), this value is multiplied by some value, as it is a normalized output.

B. Particle Swarm Optimization

In order to optimize the performance of this fuzzy controller, the membership functions are optimized using particle swarm optimization (PSO). PSO is an optimization algorithm which uses properties of a swarm (such as a flock of birds, school of fish, or colony of ants) to find an optimal solution [6]. In this case, the swarm is represented by 30 individuals (or *particles*) whose values change at each iteration. The performance of each particle is measured at each position using a “fitness” function. This function increases as the optimality of the solution increases; in this way, a particle with a higher fitness is considered to be a better fit than one with a lower fitness. Also, a record of the best position (*pbest*) for each particle is kept, as well as the best overall position (*gbest*) for all particles. The entire swarm

then searches around the gbest solution and each of the pbest solutions, all the while trying to find even better solutions.

This algorithm is ideal because of the nature of the structures being optimized. Each membership function is made up of a pair of values, usually from 0 to 1. These specify the width of the membership function. The height is set to 1, and its corresponding value along the x axis is taken as the midpoint of the span. Particle swarm optimization has been used previously by one of the authors in conjunction with fuzzy logic controllers [12].

The fitness function used in the optimization of the membership functions is listed below as (2). The weights used in this fitness function are based on the utility function developed for the ADHDP controller.

$$\begin{aligned}
 \text{Fitness} = & ((30/23) * CLS) + ((15/23) * ABC) \\
 & + ((13/23) * NCLS)
 \end{aligned}
 \tag{2}$$

Where:

CLS = Percentage of critical load satisfied

ABC = Average battery charge

NCLS = Percentage of non-critical load satisfied

One major requirement of this algorithm is that the fitness function be somewhat smooth and continuous over the acceptable range of input vectors. If this is not the case, then the swarm may not be able to find an optimal solution easily. In this investigation, the process is allowed to run until a suitable solution is found.

VI. RESULTS

After the PSO algorithm is done optimizing, a one year simulation of the PV system is carried out for the Caribou, Maine area. These simulations use data from the TMY2 database [9]. The solar profile (or global horizontal radiation) for a typical year for this region is illustrated in Fig. 5, while Fig. 6 shows the electrical energy collected from the PV array for a short time at the start of the year. Fig. 7 shows the updated membership function for the “Load” input after undergoing PSO optimization.

The PV energy produced by the solar array is then used to optimally power all (or part) of the loads, both critical and non-critical. The sum of the loads is shown in Fig. 8, which shows how the controllers performed during the simulation for the Caribou area during the first 250 hours of the year. It can be observed from this graph that the optimal controllers attempt to power the critical load at the expense of the non-critical load. Because of this, more of the critical load is met than the non-critical load. The results of this simulation are listed in Table I. In addition, a row called “Total Score” is added to Table I so that an objective comparison can be made between the controllers. This “Total Score” value is found by calculating the weighted sum of the results of each controller test. These weights are derived from the corresponding coefficients in the utility function (1). This result is identical to the value of the fitness function (2) evaluated for each

solution. Also, the non-optimized fuzzy controller results are included in this figure for reference purposes.

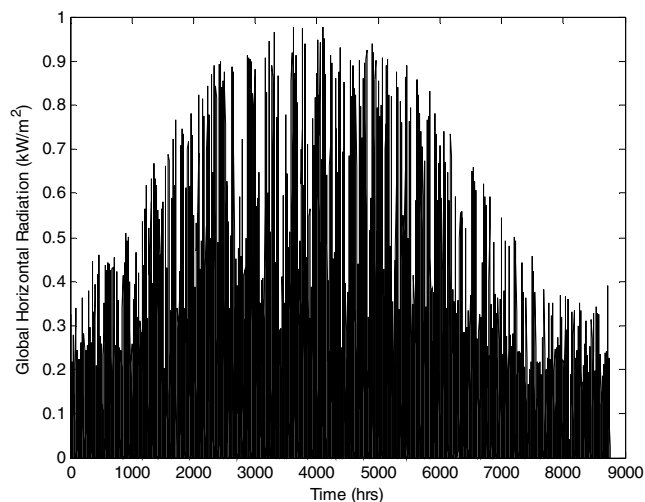


Figure 5. Solar profile for Caribou, Maine.

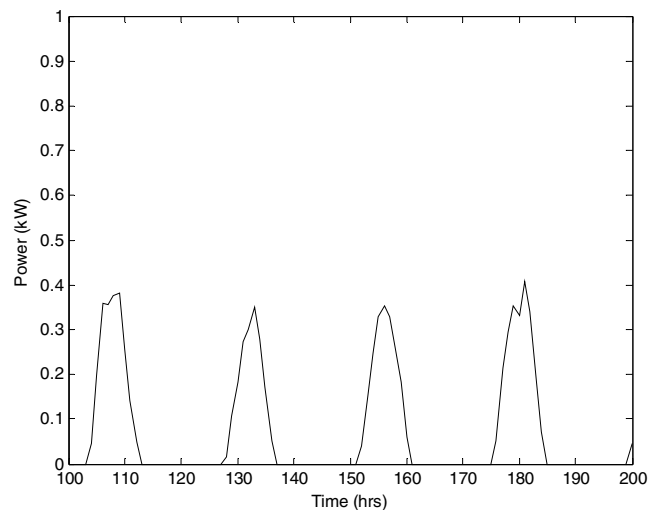


Figure 6. PV generated electricity for Caribou, Maine during early January.

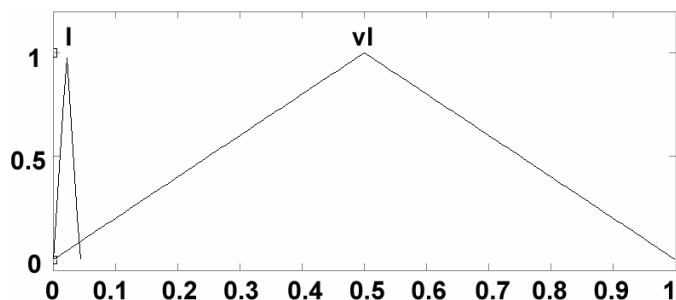


Figure 7. Optimized membership functions for the “Load” input. All membership functions not shown have been optimized to be a very small triangle near zero.

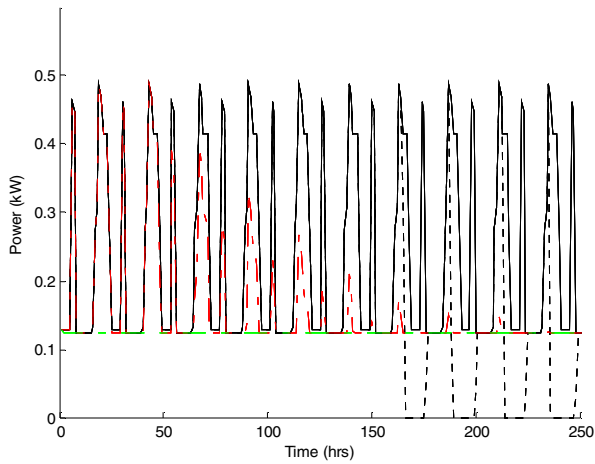


Figure 8. Sum of both critical and non-critical loads (solid black line) being satisfied by the PV-priority controller (dashed black line), the ADHDP based optimal controller (dashed red line), and the fuzzy optimal controller (dashed green line).

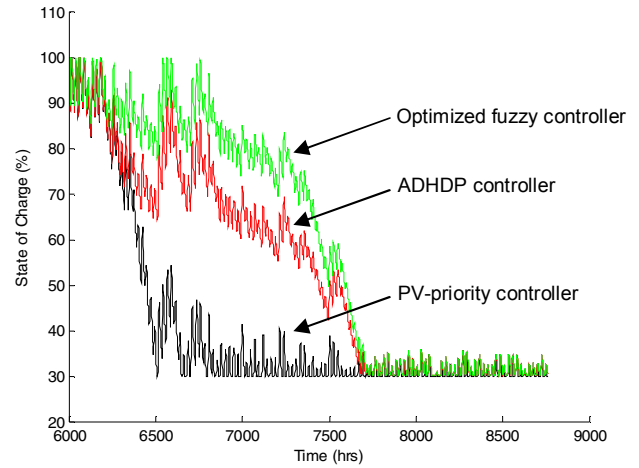


Figure 9. State of charge of the battery using the PV-priority controller (black line), the ADHDP based optimal controller (red line), and the fuzzy optimal controller (green line) during the late fall and early winter.

TABLE I. RESULTS OF ALL CONTROL STRATEGIES

City:	Caribou, Maine			
Controller:	PV-Priority	ADHDP	Non-optimized Fuzzy Logic	Optimized Fuzzy Logic
Critical Load Satisfied:	84.22%	96.54%	93.04%	92.97%
Non-Critical Load Satisfied:	77.21%	61.87%	32.16%	56.07%
Average Battery Charge:	63.87%	74.35%	76.58%	80.21%
Total Score:	1.951	2.094	1.895	2.053

Fig. 9 shows the battery state of charge for Caribou, Maine for the period of late fall and early winter using the PV-priority, ADHDP based, and optimized fuzzy logic controllers. This time period is shown because it is the most demanding situation of the year. Fig. 10 shows the state of charge of the battery for the entire year using the same controllers. Fig. 11 shows the differences in battery charge for the optimized and non-optimized fuzzy logic controllers over the entire simulated year.

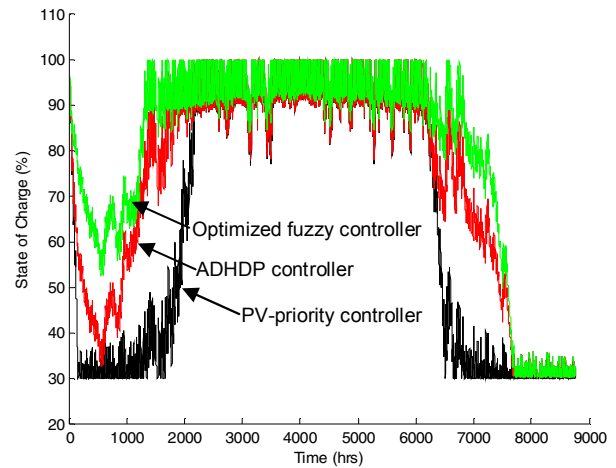


Figure 10. State of charge (for the entire year) of the battery using the PV-priority controller (black line), the ADHDP based optimal controller (red line), and the fuzzy optimal controller (green line).

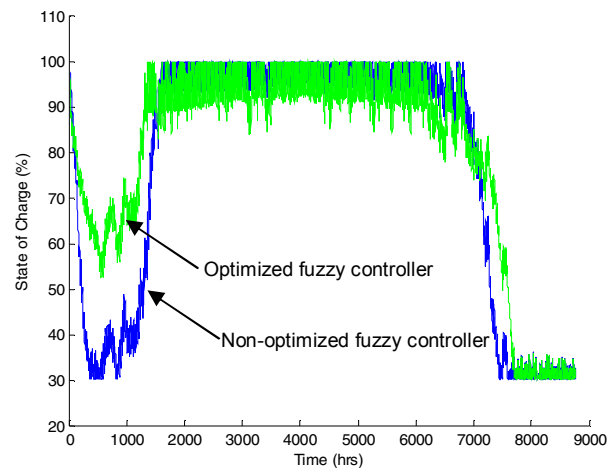


Figure 11. State of charge (for the entire year) of the battery using both the optimized (green line) and non-optimized (blue line) fuzzy logic controllers.

These results show that the optimized fuzzy logic controller keeps a higher average battery charge than the non-optimized fuzzy logic controller, which also kept a higher average battery charge than the ADHDP controller. However, the ADHDP controller is able to satisfy more of the critical and non-critical loads than both of the fuzzy logic controllers. This is evident in Table I, which summarizes the performance of each controller.

For reference, the non-optimized and optimized membership functions are included below in Figs. 12 to 20. If a membership function is not shown in the figure, then it has been optimized to the point of being a very small triangle (with nearly zero area) near one extreme of the figure. Larger memberships will be on the right end while smaller ones will be on the left end.

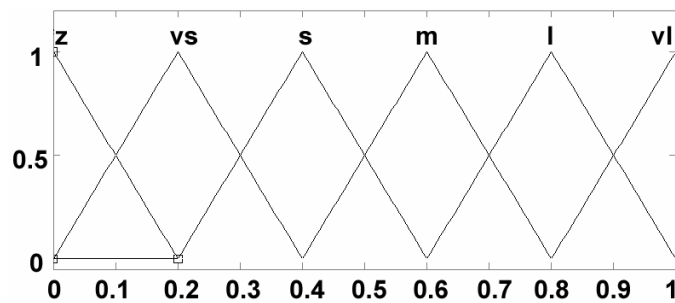


Figure 12. Non-optimized membership functions for the 3 fuzzy inputs: "PV Energy", "Battery State of Charge", and "Load".

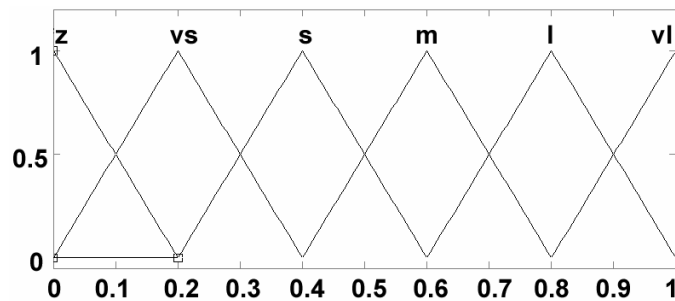


Figure 13. Non-optimized membership functions for the first 2 fuzzy outputs: "Energy to Critical Load" and "Energy to Non-Critical Load".

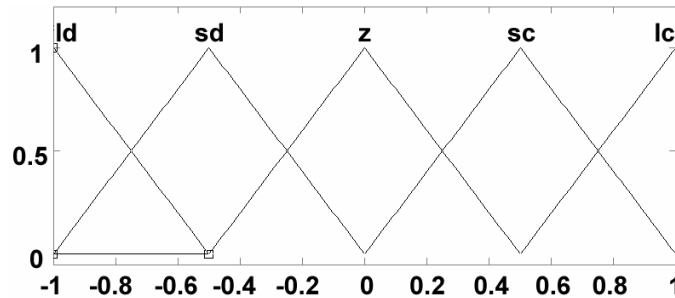


Figure 14. Non-optimized membership functions for the last fuzzy output: "Energy to the Battery".

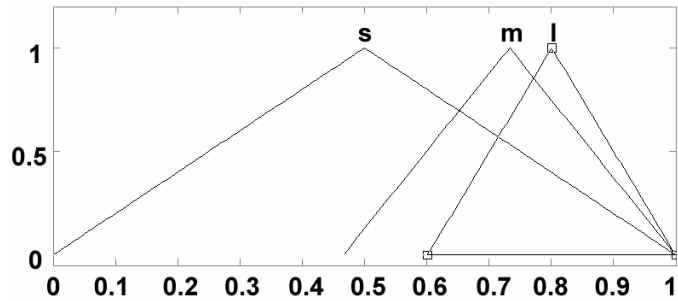


Figure 15. Optimized membership functions for the first fuzzy input: "PV Energy".

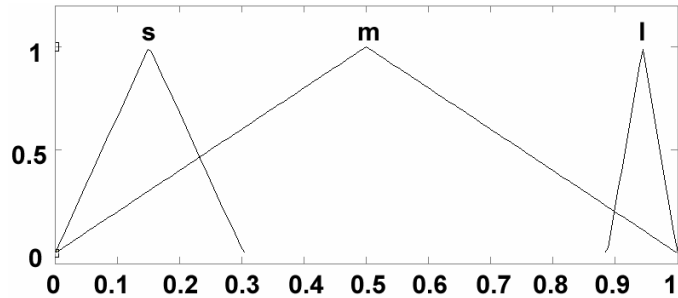


Figure 16. Optimized membership functions for the second fuzzy input: "Battery State of Charge".

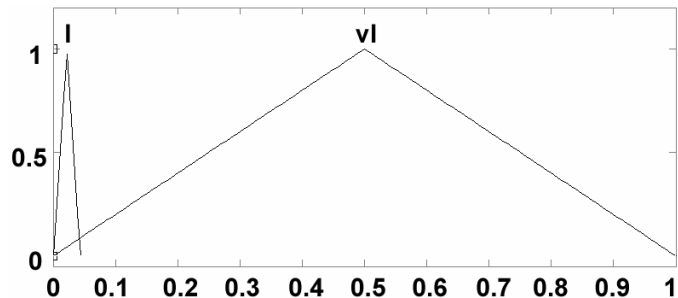


Figure 17. Optimized membership functions for the third fuzzy input: "Load".

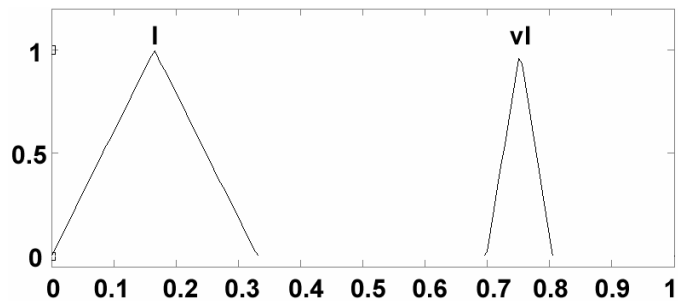


Figure 18. Optimized membership functions for the first fuzzy output: "Energy to Critical Load".

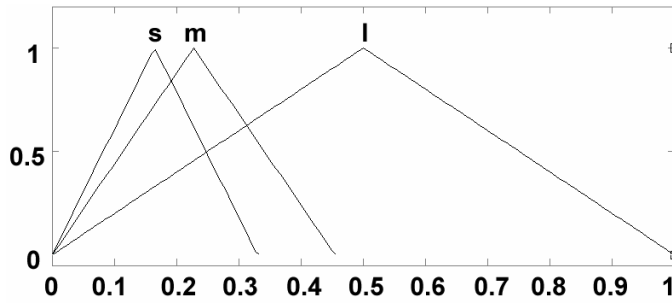


Figure 19. Optimized membership functions for the second fuzzy output: "Energy to Non-Critical Load".

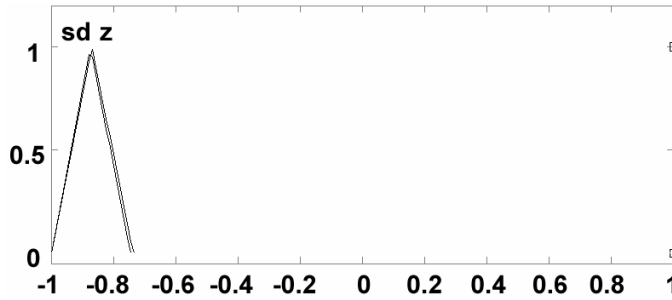


Figure 20. Optimized membership functions for the third fuzzy output: "Energy to the Battery".

VII. CONCLUSIONS

A new optimal controller utilizing fuzzy logic is designed and compared against the standard PV-priority controller as well as an ADHDP based optimal controller. The results show that both optimal controllers are able to power the critical loads for a much longer time than the standard PV-priority controller, as well as keep the battery charged to a higher average charge than the PV-priority controller.

The ADHDP controller is most likely able to satisfy more of the loads because of its ability to discount actions into the future, whereas the fuzzy logic controller does not. This property gives the ADHDP controller the ability to keep the average battery charge higher than the PV-priority controller (to be available for use in times of lesser available PV energy in some possible future state), but still allows it to power much more of the critical load and to a lesser extent the non-critical load. Not keeping the batteries fully charged at all times also allows the controller to capture more energy during the day in the summer months when the PV generated energy is higher, while still satisfying the non-critical loads. As the winter months approach, it lessens the time that the non-critical load is powered so that it can conserve battery charge during times of lower available solar energy. The optimized fuzzy controller does not act as dynamically and cannot maintain the loads for as long as the ADHDP controller, although it is much better than the PV-priority controller.

Even while able to power the critical loads for a much longer time than the standard PV-priority controller and keep the battery charged to a higher average charge than the PV-priority controller, the optimal controllers did fall short of the PV-priority controller when it came to powering the non-

critical loads. In this case, the PV-priority controller was able to power more of the non-critical load, especially when comparing against the fuzzy based controller. This is expected since both optimal controllers place a higher priority on powering the critical loads and keeping the battery charge higher. In addition to being available for potentially powering the critical load at a later date, a battery which is not depleted as often will have a longer life span and lead to a lower total cost of ownership for the owner of the PV system.

Future work will involve investigations to try to further optimize the optimal controllers to provide better performance. Specifically, the fuzzy controller will also have its rule set optimized for better control.

REFERENCES

- [1] R. A. Messenger, J. Ventre, *Photovoltaic System Engineering*, CRC Press, 2004.
- [2] G. P. Henze, R. H. Dödier, "Adaptive Optimal Control of a Grid-Independent PV-System", *Trans. ASME Journal of Solar Energy Engineering*, vol. 125, pp. 34 – 42, Feb. 2003.
- [3] A.U. Chuku, B. Oni, F. Kuate, and E. Overton, "Making Solar Energy a Viable Stand-alone Alternative Through Efficient Control Algorithm", in *International Conference on Power Systems Operation and Planning – VI (ICPSOP)*, pp. 71-79, 2005.
- [4] J.A. Momoh, A.R. Ofoli, "Load management and control of the photovoltaic (PV) system using fuzzy logic", *Large Engineering Systems Conference on Power Engineering (LESCOPE)*, pp. 184-188, 2001.
- [5] P. J. Werbos, "Approximate Dynamic Programming for Real-time Control and Neural Modeling", in *Handbook of intelligent control*, White and Sofge, Eds. New York: Van Nostrand Reinhold, pp 493-525, 1992.
- [6] J. Kennedy and R. Eberhart, *Swarm Intelligence*, Morgan Kaufman Publishers, San Francisco, CA. ISBN 1-55860-595-9, 2001.
- [7] G. K. Venayagamoorthy, R.G. Harley and D.C. Wunsch, "Comparison of Heuristic Dynamic Programming and Dual Heuristic Programming Adaptive Critics for Neurocontrol of a Turbogenerator," *IEEE Trans. Neural Networks*, vol. 13, pp. 764 -773, May 2002.
- [8] D. Prokhorov, D.C. Wunsch, "Adaptive Critic Designs", *IEEE Trans. Neural Networks*, vol. 8, pp. 997-1007, Nov. 1997.
- [9] "TMY2 User's Manual", Jun. 1995. National Renewable Energy Laboratory, Golden, Colorado. [Online] Available: http://tredc.nrel.gov/solar/old_data/nsrdb/tmy2/
- [10] Kyocera KC200GT data sheet, <http://www.kyocerasolar.com/pdf/specsheets/KC200GT.pdf>, accessed April 30, 2006.
- [11] R. L. Welch, G. K. Venayagamoorthy, "HDP based Optimal Control of a Grid Independent PV System", *IEEE Power Engineering Society*, June 2006
- [12] S. Doctor, V. K. Venayagamoorthy, "Navigation of mobile sensors using PSO and embedded PSO in a fuzzy logic controller", *IEEE Industry Applications Conference*, vol. 2, pp. 1200-1206, Oct. 2004.