Energy Conversion and Management 51 (2010) 928–937

Contents lists available at ScienceDirect

Energy Conversion and Management

journal homepage: www.elsevier.com/locate/enconman

Energy dispatch fuzzy controller for a grid-independent photovoltaic system

Richard L. Welch, Ganesh Kumar Venayagamoorthy *

Real-Time Power and Intelligent Systems Laboratory, Missouri University of Science & Technology, Rolla, MO 65409, USA

ARTICLE INFO

Article history:
Received 2 May 2009
Accepted 27 November 2009
Available online 29 December 2009

Keywords:
Energy management
Fuzzy logic controller (FLC)
Grid independent photovoltaic system
Particle swarm optimization (PSO)

ABSTRACT

This paper presents the development of an optimized fuzzy logic based photovoltaic (PV) energy dispatch controller using a swarm intelligence algorithm The PV system considered is grid-independent and consists of a fuzzy logic controller (FLC), PV arrays, battery storage, and two types of loads: a constant critical load and a time-varying non-critical load. The swarm intelligence applied in this paper is the particle swarm optimization (PSO) algorithm and is used to optimize both membership functions and rule set of the FLC. By using PSO algorithm, the optimized FLC is able to maximize energy to the system loads while also maintaining a higher average state of battery charge. This optimized FLC is then compared with the standard energy dispatch controller, referred to as the "PV-priority" controller. The PV-priority controller attempts to power all loads and then charge the battery resulting on lesser number of days of power to critical loads unlike the optimized FLC.

1. Introduction

As the cost of traditional fossil fuels continues to rise, the cost of electricity generated by traditional means also increases. However as technology and manufacturing processes improve the cost of alternative energy sources such as solar and wind energy decreases [1]. This narrowing gap between the rising cost of traditional energy sources and lowered cost of renewable energy is driving demand growth for renewable energy to unprecedented levels. Even so, the difference in cost of electricity generated by wind (and especially solar) and that generated by conventional sources is not insignificant, thus making some optimal control of a renewable energy source a good way to make the overall system more economical. By using a smart or optimal energy dispatch controller, more of the load can be met than by using a traditional controller with the same sized system. Additionally, others [2] have employed a decentralized topology when utilizing large numbers of photovoltaic (PV) arrays in order to better match individual components, but even these technologies can benefit from optimal control. Researchers have reported on improving the efficiency of photovoltaic systems by carrying out optimal control of PV systems with fuel cells [3]. In this case, the excess solar energy is stored by transforming it into hydrogen.

Traditionally, PV energy dispatch controllers have been simple devices that do not assign priority to various loads. Instead, they attempt to power all loads all of the time, and if there is any excess energy, then that excess energy is used to charge the batteries. Previous work by other researchers have reported on the development of non-optimized fuzzy logic controllers for large solar thermal plants [4,5], as well as small grid-independent systems [6]. Additionally, one effort [7] produced a fuzzy logic controller for a solar thermal plant with a rule base optimized using a genetic algorithm. The authors have reported optimizing the membership functions of a fuzzy logic controller [8].

In this paper, a fuzzy logic controller (FLC) is developed to assign priority to the installed system loads such that all critical loads receive a higher priority than the non-critical loads, and so when there exists a shortage of available energy the critical loads are first met before attempting to power the non-critical loads. This energy dispatch controller is also optimized to maintain a higher battery charge so that the controller is better able to power critical loads during an extended period of unfavorable weather conditions or low solar insolation. In this study, the simultaneous optimization of the membership functions and rule base of a fuzzy logic controller is carried out. A simulation study is carried out using Matlab with hourly data from the typical meteorological year 2 (TMY2) database [9]. The TMY2 database maintains hourly meteorological data for many cities, and most important for this study is the solar radiation or insolation received at each location. In this paper, three locations are chosen that represented varying degrees of average insolation. For the low average insolation case, Caribou, ME is selected; for the average case, Omaha, NE is chosen; and finally, for the high average insolation case Las Vegas, NV is selected.

The rest of the paper is organized as follows. Section 2 describes the grid-independent photovoltaic solar energy system model. Sections 3 and 4 describe the PV-priority controller and fuzzy logic controller design, respectively. Finally, Section 5 presents the results and conclusions.

* Corresponding author.
E-mail address: gkumar@ieee.org (G.K. Venayagamoorthy).

0196-8904/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.enconman.2009.11.031

2. Grid-independent photovoltaic solar energy system

The complete photovoltaic system model is composed of the PV array, maximum power point tracker, energy dispatch controller, battery charge controller, batteries, inverter, critical loads and non-critical loads. The critical load consists of loads that need to be powered all the time (such as refrigeration and emergency radio communication), while the non-critical load contain items which are non-essential (television, etc.).

In order to simplify the simulation and focus on the controller aspect of this system, all of the supporting system components (such as the inverter, maximum power point tracker, wiring, batteries, etc.), are assumed to operate at 100% efficiency; in the real world however, these components are non-trivial, with several maximum power point tracking algorithms being recently proposed [10–14]. Likewise, new energy storage systems [15,16] may eclipse the traditional lead-acid batteries in terms of performance. Also, the efficiency of the PV array model is taken as 11% to account for various non-optimal conditions (such as array misalignment, dust on the arrays, temperature, etc.). This value is representative of the current commercially available range of efficiencies for PV arrays. Generally, PV panels vary in efficiency from 6% to 30%; although the high efficiency panels are generally reserved for spacecraft usage because of their high radiation tolerances and higher power-to-weight ratio. A rough equivalent to the PV arrays being simulated in this paper would be an array of eight Kyocera KC200GT panels. These panels are over 16% efficient and output 200 W during optimal conditions [17]. A minimum state of charge for the battery is taken to be 30%, and is consistent with standard deep cycle lead-acid batteries.

Due to insufficient PV energy during winter months in several locations and no PV energy being available at night, a control system is required to decide the amount of energy to be dispatched to the different loads, including charging of the battery. The schematic diagram form of the system studied is shown Fig. 1 (energy flow depicted by arrows).

The system states consist of the PV energy (see nomenclature for complete list of system abbreviations that are used for this paper), critical load, non-critical load and current battery charge. The control signals are energy dispatch to the critical load, energy dispatch to the non-critical load and finally energy dispatch to the battery.

Additionally, a map showing the total solar resource availability for the United State of America is shown in Fig. 2. The average output from the PV array over a year at three different selected locations is shown in Fig. 3 in order of increasing average insolation (Caribou, ME, then Omaha, NE, and finally Las Vegas, NV).

3. PV-priority controller

The standard controller, called the "PV-priority" controller [18], is an energy dispatch controller which always tries to meet the loads (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 remaining is drawn from the battery. If instead there is an excess, then whatever is left over after meeting the loads is dispatched to charge the battery. In this way, the controller attempts 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 an abundant supply of PV energy. However, when there is insufficient PV energy, then the battery will not be fully recharged and the loads will not be met. The weather and user loads are stochastic in nature; therefore there is no one definitive model at all times. PV-priority control is not an optimal strategy. Thus, it is logically to look at intelligent model-free learning methods of controlling such a system optimally. The flowchart for operation of this PV-priority controller is fuzzy logic controller. Each of the main components is discussed below.

4.1.1. Fuzzification process

The input membership functions take the inputs to the controller (after they have been normalized by some value suitabg-242nporby thd

4.2. Optimized fuzzy logic controller

In order to optimize the fuzzy logic controller, particle swarm optimization (PSO) [19,20] is used to optimize both the membership functions in the fuzzification and defuzzification modules, and rule set of the inference engine. PSO is an iterative algorithm that represents possible solutions to a given problem with a series of multidimensional vectors. Each vector is called a *particle* and contains one complete solution. Each dimension of each particle represents one parameter of a solution to be optimized. In this paper, 30 particles are chosen to represent each possible controller with the following parameters:

Each fuzzy set in each membership function (besides the first and last fuzzy set in each membership function) is represented by three values: the first one for the left-most point, the second one for the right-most point, and the third one for the middle point. The end values had a fixed leg (either the left or right, depending on which end of the membership function the fuzzy set occupies), so they are only specified by two values. Since there are five membership functions with six fuzzy sets (three inputs and two outputs) and one membership function with five fuzzy sets (one output), this equates to 93 parameters just to represent the membership functions.

Each rule is represented by three values (one for each output). Since there are three inputs and each can take on as many as six values, there are 216 rules. Since each is represented by three values, this adds another 648 parameters.

Summing each of these values up, it can be seen that each solution is represented with 741 parameters, this means each PSO particle has 741 dimensions. PSO optimizes these values by using a process based on social interaction, much like a flock of birds or school of fish. In PSO, a collection of particles takes on values that represent a possible solution. As the swarm of particles moves about (according to a defined velocity determined by how well each is doing), the particles' values change. As they change, a record of each particle's best position (called *pbest*) is kept as well as the global overall best position (called *gbest*). The equations to determine velocity and position updates are shown below in (1) and (2), respectively. In each, the index i ranges over the number of particles.

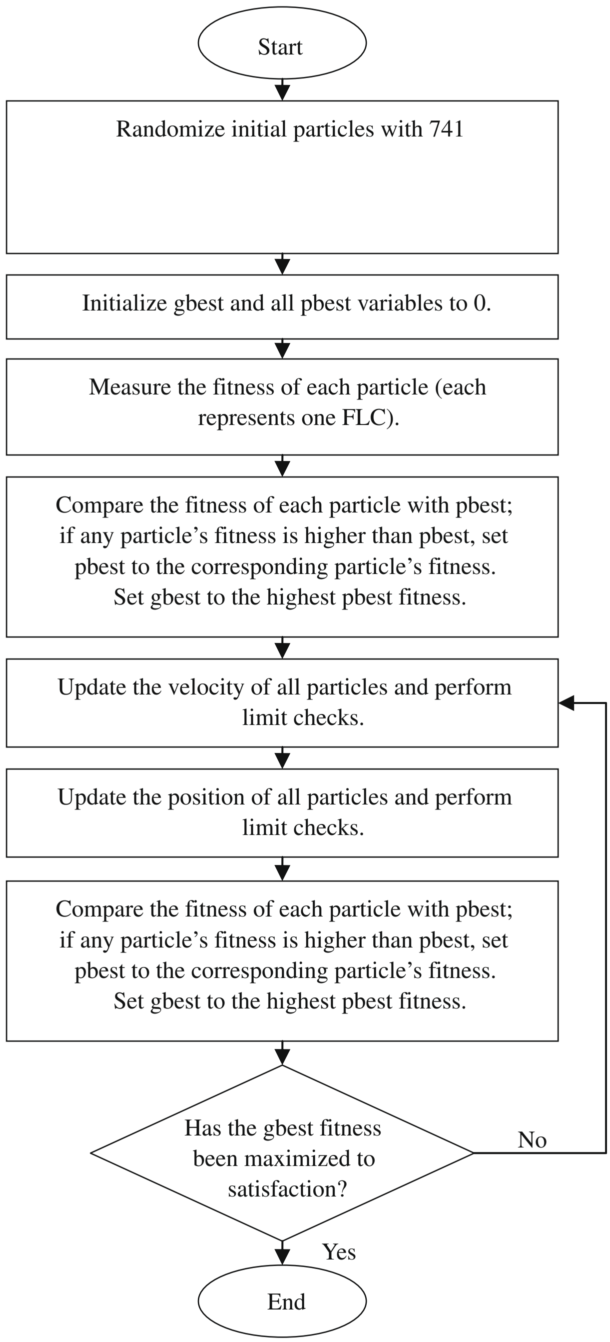
$$\begin{aligned}\mathrm{Velocity}(i) = {} & 0.1 * \mathrm{rand} * \mathrm{Velocity}(i) + 2 * \mathrm{rand} * (\mathrm{pbest}(i) \\ & - \mathrm{Position}(i) + 2 * \mathrm{rand}(\mathrm{gbest} - \mathrm{Position}(i)\end{aligned} \tag{1}$$

$$\mathrm{Position}(i) = \mathrm{Position}(i) + \mathrm{Velocity}(i) \tag{2}$$

The quality of solution for each particle is measured by the *fitness function* when evaluated at the particle's current position. In this paper, the fitness function is given by (3). Each term in (3) is a percentage calculated over the entire year, where A_1, A_2, and A_3 are defined to be 30/23, 15/23, and 13/23, respectively. These values give emphasis to meeting the critical load over the other objectives, and were empirically determined.

$$\begin{aligned}\mathrm{Fitness} = {} & A_1 \times (\mathrm{Critical\ Load\ Satisifed}) \\ & + A_2 \times (\mathrm{Average\ Battery\ State\ Of\ Charge}) \\ & + A_3 \times (\mathrm{Noncritical\ Load\ Satisfied})\end{aligned} \tag{3}$$

A higher fitness function value (or just called the fitness) results from a better performing individual. As the algorithm progresses, it is expected that the best solution should continue to improve over time (which is shown by an improving fitness over time). The flow-entry in any membership function spans a distance of less than 0.1. Otherwise the membership function entry is widened to a width of 0.1. Also, some checks are put into place to verify that the entire width of the input space is mapped to a fuzzy set. That is to say that no possible input could fall outside of a membership function entry. Lastly, checks are placed on each dimension of the final instantiated object to always be within a boundary that made physical sense. That is, all dimensions having to do with optimizing the membership functions had to be within a range of 0–1 (inclusive); and all dimensions having to do with optimizing rules had to be within a similar range so that all rules associated with an instantiated FLC could be mapped to an appropriate value.



5. Results and discussion

5.1. Results

The results from this study is much improved from the authors' previous results of optimizing a fuzzy logic controller [8], in which only the membership functions were optimized. In this study, the FLC's rule base is also simultaneously optimized such that the performance of the entire FLC is superior to that of the FLC where just the fuzzy membership sets were optimized. Also in this study, one FLC is optimized using PSO with data from one city (Caribou), and then the performance of this energy dispatch controller is tested at two other locations (Omaha and Las Vegas). The training city is selected because it receives less insolation that the other two cities. After running the PSO optimization for 50 iterations, the final gbest fitness is 2.086250. This can be seen from Fig. 8 showing how the best particle's (gbest) fitness increases throughout the simulation:

The actual results (as well as the results of the PV-priority and un-optimized fuzzy logic controllers) for all three cities are listed in Tables 1–3. The *Relative Performance* row is calculated by evaluating (4) for each controller and city combination. This gives an objective method of comparing relative controller performance.

$$\text{Relative performance} = \frac{1}{365}\sum_{\text{Day}=1}^{365} \times \frac{\sum_{\text{Hour}=1}^{24}\left(\left(A_1 * \frac{L_{CL}}{L_{CL,\max}}\right) + \left(A_2 * \frac{E_B}{E_{B,\max}}\right) + \left(A_3 * \frac{L_{NCL}}{L_{NCL,\max}}\right)\right)}{24 * (A_1 + A_2 + A_3)\left(\frac{\sum_{\text{Hour}=1}^{24} E_{PV}}{\sum_{\text{Hour}=1}^{24}(E_{PV} - E_{PV,\text{wasted}})}\right)} \quad (4)$$

The data from Table 1 shows that the FLC that had both the membership functions and fuzzy rule set optimized performed 5.22% better in energy usage than the FLC from [8] that only had the membership functions optimized, and also outperformed the initial un-optimized FLC and PV-priority scheme. Table 2 shows the performance of the optimized FLC for the average insolation case of Omaha, NE. Again in this comparison the FLC that had both the membership functions and fuzzy rule set optimized performed better by 26.13% in energy usage than un-optimized FLC. Finally, Table 3 shows the controller performance for the case of high insolation in Las Vegas, NV. Again, the FLC with the optimized membership functions and rule set is the best performer and is 37.28% better in energy usage than un-optimized controller. Furthermore, the solar insolation in Las Vegas is much more than in Caribou, ME and therefore, the FLC efficiency can be still improved if the FLC is fine tuned custom to Las Vegas, and likewise to each location.

Table 1
Summary of controller performance for the Caribou, ME area.

	PV-priority	Un-optimized fuzzy	Optimal fuzzy	Optimized membership functions only [8]
Percentage critical load met	84.22 [914.8 kW h]	93.04 [1011.0 kW h]	95.51 [1038.0 kW h]	92.97 [1010.4 kW h]
Percentage non-critical load met	77.21 [778.4 kW h]	32.16 [324.2 kW h]	61.8 [623.0 kW h]	56.07 [565.2 kW h]
Average percentage of battery charge	63.87 [22.07 kW h]	76.58 [26.47 kW h]	75.3 [26.02 kW h]	80.21 [27.72 kW h]
Total energy supplied (kW h)	1715.27	1361.67	1687.02	1603.32
Relative performance	0.6168	0.5632	0.6921	0.6498

Table 2
Summary of controller performance for the Omaha, NE area.

	PV-priority	Un-optimized fuzzy	Optimal fuzzy
Percentage critical load met	88.66 [963.0 kW h]	97.08 [1054.0 kW h]	99.00 [1075.0 kW h]
Percentage non-critical load met	85.85 [865.5 kW h]	36.03 [363.3 kW h]	71.44 [720.2 kW h]
Average percentage battery charge	68.98 [23.80 kW h]	84.30 [29.10 kW h]	84.36 [29.20 kW h]
Total energy supplied (kW h)	1852.30	1446.40	1824.40
Relative performance	0.6170	0.5568	0.6972

Table 3
Summary of controller performance for Las Vegas, NV area.

	PV-priority	Un-optimized fuzzy	Optimal fuzzy
Percentage critical load met	98.3 [1068.0 kW h]	99.62 [1082.0 kW h]	100.00 [1086.0 kW h]
Percentage non-critical load met	97.46 [982.5 kW h]	35.55 [358.4 kW h]	89.83 [905.6 kW h]
Average percentage battery charge	82.06 [28.40 kW h]	97.08 [33.60 kW h]	92.34 [31.90 kW h]
Total energy supplied (kW h)	2078.90	1474.00	2023.50
Relative performance	0.5981	0.4415	0.6142

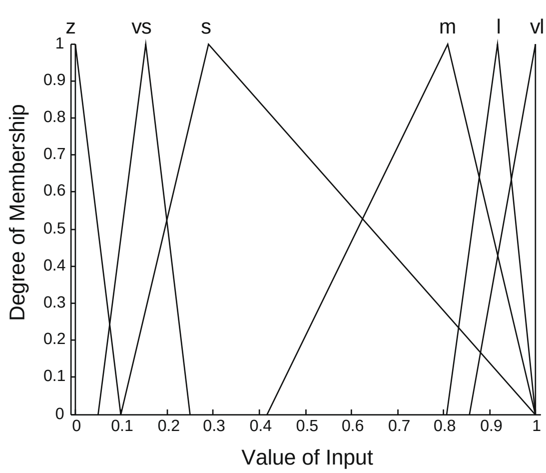


Fig. 9. Optimized membership function for current load input.

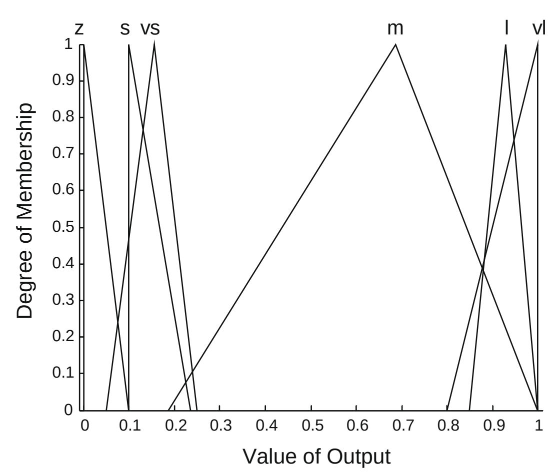


Fig. 10. Optimized membership function for energy to critical load output.

These results show that optimizing the membership functions as well as the rule set allows the fuzzy logic controller to perform significantly better than the un-optimized fuzzy controller and the PV-priority controller. Fig. 9 shows the optimized membership functions for the current load input, while Fig. 10 shows the optimized membership functions for energy dispatched to the critical load output. Figs. 11 and 12 show the unmet critical and non-critical loads, respectively, using data from the Caribou area and the optimized FLC. It is shown that the optimized FLC is able to meet the vast majority of the critical load and most of the non-critical loads, except into the early winter where the insolation falls to inadequate levels. Of course, all the unmet load demand can be supplied with energy by increasing either the size of the battery capacity or increasing the size of the PV output or doing both. Figs. 13–15 show the annual battery state of charge for the Caribou area using the PV-priority controller, initial un-optimized FLC, and optimized FLC, respectively. Fig. 16 shows the relative performance optimization, especially those with higher rule numbers as these

connected photovoltaic system. IEEE Trans Ind Electron 2006;53(4):1027–35.
[12] Park JH, Ahn JY, Cho BH, Yu GJ. Dual-module-based maximum power point tracking control of photovoltaic systems. IEEE Trans Ind Electron 2006;53(4):1036–47.
[13] Kwon JM, Nam KH, Kwon BH. Photovoltaic power conditioning system with line connection. IEEE Trans Ind Electron 2006;53(4):1048–54.
[14] Mutoh N, Ohno M, Inoue T. A method for MPPT control while searching for parameters corresponding to weather conditions for PV generation systems. IEEE Trans Ind Electron 2006;53(4):1055–65.
[15] Jiang Z, Dougal RA. A compact digitally controlled fuel cell/battery hybrid power source. IEEE Trans Ind Electron 2006;53(4):1094–104.
[16] Lemofouet S, Rufer A. A hybrid energy storage system based on compressed air and supercapacitors with maximum efficiency point tracking (MEPT). IEEE Trans Ind Electron 2006;53(4):1105–15.
[17] Kyocera Corporation. KC200GT data sheet; 2006. <http://www.kyocerasolar.com/pdf/specsheets/KC200GT.pdf>.
[18] Henze GP, Dodier RH. Adaptive optimal control of a grid-independent PV-system. ASME J Solar Energy Eng 2003;125:34–42.
[19] Kennedy J, Eberhart R. Swarm intelligence. San Franscisco (CA): Morgan Kauffman Publishers; 2001 [ISBN 1-55860-595-9].
[20] Engelbrecht AP. Computational intelligence an introduction. Wiley; 2003.