Jurnal_Aryu-2017-2.pdf

by

International Journal of Innovative Computing, Information and Control
Volume **13**, Number **3**, June **2017**

ICIC International ©2017 ISSN 1349-4198
pp. 711–726

INCORPORATING SCADA SOFTWARE AND HIGH LEVEL PROGRAMMING LANGUAGE FOR IMPLEMENTING THE OPTIMIZATION TECHNIQUE IN SMART GRID

Aryuanto Soetedjo, Abraham Lomi and Yusuf Ismail Nakhoda

Department of Electrical Engineering
National Institute of Technology
Jalan Raya Karanglo KM 2 Malang, Indonesia
aryuanto@gmail.com

Received October 2016; revised February 2017

Abstract. *This paper presents the software development by incorporating the Winlog SCADA software and the Delphi program to implement the optimization technique in smart grid. The approach exploits the benefits of the SCADA software in developing the human machine interface and the Delphi program in calculating the optimization algorithms. A Modbus TCP/IP protocol is adopted for the data exchange between them. A linear programming technique is employed to minimize the daily electricity cost of the smart home appliances that are supplied by the grid and the renewable energy resources. The experimental results show that the longest execution time of 2.884 minutes is achieved to solve the linear programming technique, which is suitable for real-time implementation of the given optimization problem. Further, the visualization of the system's parameters developed by the Winlog SCADA could be used to monitor the smart grid effectively.*

Keywords: Delphi program, Linear programming, Modbus TCP/IP, Smart grid, Winlog SCADA

1. **Introduction.** Smart grid (SG) is a new technology in delivering the electrical energy to the costumers. It integrates the information and communication technology (ICT) into the electrical networks. This technology considers the costumers as the integral part of the SG system. Thus, it allows the users to modify the electrical usages for achieving the higher efficiency [1].

One important aspect tackled by the SG is the demand side management (DSM) as discussed in [2]. The DSM is an approach to optimize the user's energy consumption in the long term. Regarding to the optimization methods, the DSM could be classified into [2]: a) individual and cooperative users; b) deterministic and stochastic optimization; c) day-ahead and real-time operating plan. The objective functions in the optimization problems are to minimize the cost of electricity, minimize the discomfort level of the device operation, and maximize the local generation use.

The DSM for residential users is the interesting topic addressed by the researchers [3-9]. They employed the linear programming techniques to solve the optimization problems for minimizing the costs [3-7], and minimizing the peak hourly loads [8,9].

The integer linear programming (ILP) was used to minimize the costs of the electricity and the heat, under the constraint that the demand in the house should be matched with the supply [3]. The binary linear programming (BLP) was used to minimize the daily electricity cost in the smart home with the renewable energy resources [4]. Since the variables to be solved are the switch control of the appliances, the binary form was adopted. The mixed BLP was used to optimize the home appliances scheduling, where the load variables consist of both binary and integer values [5]. The binary value was assigned to the time-shiftable appliance, while the integer value was assigned to the power-shiftable appliance.

To solve the optimization problems that involve the integer and non-integer numbers, the mixed ILP (MILP) was adopted in [6-9]. In [6], the scheduling was used to minimize the power profile, i.e., the power assignment to the appliances as a function of time. In the work, the real value was used to assign the power profile, while the binary integer value was used to indicate whether the power was processed by the device. The MILP was used to solve the optimal scheduling of household appliances, in which the objection function considers three criteria [7], i.e., a) minimizing the energy prices; b) maximizing the scheduling preferences; c) maximizing the climatic comfort.

Instead of minimizing the prices, the optimization technique in [8,9] was used to minimize the peak hourly load. Similar to [5], they used the time-shiftable load and the non time-shiftable load. The method was suitable for fixed price electrical consumption.

Most of the works mentioned before used the simulation program on the computer to implement the optimization algorithms, such as the MATLAB program [4,6,8] or Java program [7]. [9] implemented the optimization algorithm on the embedded system, which was suitable for the real-time application.

In the real application, a supervisory control and data acquisition (SCADA) system is commonly employed to monitor and control the SG [10-18]. In [10], the SCADA was employed to monitor and control the grid distribution system, which was equipped with the smart meter and the GSM communication. The smart meter was used to monitor the supply parameters, the fault detection and location. By monitoring the supply parameters, the electricity bill could be monitored and reduced accordingly.

A human machine interface (HMI) was developed in [11] to monitor the SG system. The HMI was built using the SCADA software (InduSoft Web Studio). The HMI visualized all functions in the SG such as the turning on/off relays, trending screen of the voltage and current, and the alarming screen.

The SCADA testbeds were developed to test the SG system on the laboratory [12,13]. The testbed in [12] consists of the main server called plant information (PI) system for data receiving and visualization, OPAL-RT real-time digital simulator, LabView boards, and phase measurement unit (PMU). Different communication protocols such as IEEE-C37.118 and Modbus protocols were adopted to transfer the data between the devices and the PI system. The testbed in [13] consists of the loads, the PV generator, the wind turbine generator, the grid simulators, and the SG center. The Winlog SCADA software [19] was implemented on the SG center. The Xbee wireless communication was employed to connect the simulator and the SG center. The Modbus protocol was used for communicating the simulator and the SCADA software on the SG center.

The Eclipse SCADA, an open source SCADA project, was employed to develop the SG co-simulation [14]. They combined the SCADA, the traditional power system simulation GRIDLAB-D, the information and communication technology (ICT) using the common information model (CIM) as the intermediate model. The ICT was modeled using the open source toolkit AKKA. The developed system could be used to analyze the SG under the real and simulated environments.

A SCADA house intelligent management (SHIM) was developed in [15] to manage the electricity demand and supply. The artificial neural network (ANN) was employed to learn to user profile. The system employed the deterministic and genetic algorithm methods for solving the optimization problem. They implemented the algorithm using the MATLAB program. The SHIM was tested to reduce the total energy consumption using the real data of domestic home consumers [16], and to realize the real-time pricing application [17]. The SHIM was improved to be integrated with OPAL-RT for the real-time simulation [18].

As described in the previous works, the SCADA software provides an easy tool for developing the HMI to monitor and control the SG system. However, it is difficult or impossible to implement the complex algorithm such as the LP techniques using the SCADA software. Alternatively, the researchers employ the high level programming language such as Java, C++, or MATLAB to implement the LP techniques for solving the optimization problems in the SG systems. Unfortunately, it is difficult and consumes the time to build the HMI using the high level programming language.

This paper deals with the software development for implementing the optimization technique in the SG system, more specifically to minimize the daily electricity cost as proposed by [4]. Instead of simulating the algorithm [4], our approach proposes a technique for the real-time implementation. Our approach exploits the benefits of both SCADA software and high programming language, in the sense that the superiority of SCADA software for developing the HMI is incorporated with the superiority of high programming language for solving the optimization problems. The main contributions of our approach are: a) The real-time implementation of the SG could be developed using the SCADA software; b) The visualization of the SG's parameters could be designed easily; c) The optimization technique (BLP) could be implemented in the real-time application.

In the proposed system, the Winlog SCADA [19] is employed as the SCADA software, while the Delphi programming is employed as the high level programming language. A novel approach using the Modbus TCP/IP protocol [20] is proposed to incorporate both the Winlog SCADA and the Delphi program.

The rest of paper is organized as follows. Section 2 presents the system architecture of the proposed method. The implementation is described in Section 3. The experimental results are described in Section 4. The conclusion is covered in Section 5.

2. Proposed System.

2.1. System configuration. The configuration of the proposed system is illustrated in Figure 1. It consists of two main components, i.e., the Winlog SCADA software and the Delphi XE6 software. Both software packages are installed on a computer that represents the SG center. The Modbus TCP/IP is used as the communication protocol to exchange the data. The reason to adopt the Modbus TCP/IP protocol is described in the following. The Modbus TCP/IP is an open source protocol that is widely used in the industrial applications. Almost all SCADA software packages support the Modbus TCP/IP protocol. The protocol offers a simple request/reply message defined by the function codes. Thus, it is easy for deploying the protocol on the other software, such as the Delphi which is used

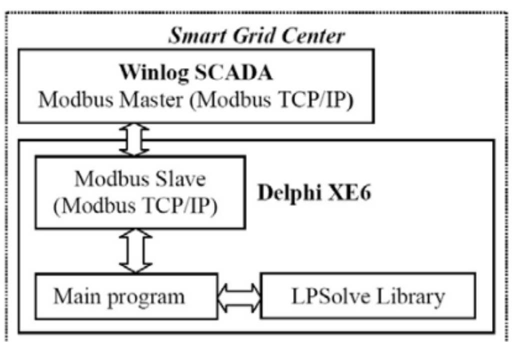


FIGURE 1. Configuration of the proposed system

in this work. Even though the protocol is usually used to connect two or more devices, since the Modbus TCP/IP protocol runs on the Ethernet, it could be adopted for a single device (computer) by assigning an IP address of the local host.

In the proposed system, the Winlog SCADA initiates the communication with the Delphi program. The Delphi program replies the data when it is requested by the Winlog SCADA. Thus, in the Modbus configuration, the Winlog SCADA acts as the Modbus master, while the Delphi program acts as the Modbus slave.

The Winlog SCADA is used to visualize the status and parameters of the SG, in this case the hourly power consumed by the appliances. The Winlog SCADA receives the data from the local controllers on the fields, i.e., the power generators and smart homes. Then the data are sent to the Delphi to be processed (optimized). After processing, they are sent back to the Winlog SCADA as the control command to the local controllers.

In the research, the Delphi program is used to calculate the LP for solving the optimization problem proposed in [4]. The Delphi program performs following three tasks: a) receive the data from the Winlog SCADA; b) calculate the LP; c) send the result to the Winlog SCADA.

The main issue in the incorporation of the Winlog SCADA and the Delphi is the different behavior of the SCADA software and the Delphi program, where the SCADA program transfers the data periodically in a specific time interval, while the LP is calculated by the Delphi only once. Therefore, it needs a mechanism to check the time when the data are ready to be processed and the time when the calculation is finished. It will be discussed in Section 3.

2.2. **Optimization technique.** In this paper, the optimization technique proposed in [4] is used as a sample case for verifying the proposed method. In the system [4], the appliances in a smart home are supplied from the electricity grid, the wind energy and the solar energy. The optimization technique is employed to optimize the load scheduling problem, i.e., to schedule the operation of appliances in order to minimize the daily electricity cost under the varying hourly prices of the electricity grid from the utility company, and the availability of renewably energy resources.

In our work, to simplify the formulation, the number of home appliances is limited to four, and the formulas are rewritten accordingly as described in the following. Readers may refer to [4] for the original formulas. It is assumed that each load is supplied by one resource at any time interval. Let us define xw_{ih}, xp_{ih}, and xg_{ih} are the load switching vectors representing the states that the load-i at the time interval-h is supplied by the wind turbine generator, the PV generator, and the grid, respectively. In this case, $i \in \{1, 2, 3, 4\}$ and $h \in \{1, 2, 3, \ldots, 24\}$, where the time interval-h is expressed in hour unit. The hourly load power consumption of load-i is expressed as lwh_i. The hourly electricity prices at time interval-h of the wind turbine generator, the PV generator, and the grid are expressed as cw_h, cp_h, and cg_h, respectively. The hourly available energies of the wind turbine generator and the PV generator are expressed as ew_h and ep_h, respectively. For the scheduled load-i, the total operation hours, the start time, and the stop time are expressed as oh_i, sta_i, and sto_i, respectively.

The hourly cost of each load is calculated as the multiplication of hourly load power consumption and the price of electric generator (the grid, the wind turbine, and the solar power) that supplies the load at the particular hour. The daily electricity cost is defined as the total daily cost (24 hours) of all loads. The objective of optimization technique is to minimize the daily electricity cost which is expressed as

$$Min \sum_{h=1}^{24} \sum_{i=1}^{4} lwh_i \left(cw_i \times xw_{ih} + cp_i \times xp_{ih} + cg_i \times xg_{ih} \right) \tag{1}$$

such that

$$xw_{ih}, xp_{ih}, xg_{ih} \in \{0, 1\} \tag{2}$$

$$xw_{ih} + xp_{ih} + xg_{ih} \leq 1 \quad \forall i \in \{1, 2, 3, 4\} \tag{3}$$

$$0 \leq \sum_{i=1}^{4} lwh_i \times xw_{ih} \leq ew_h \tag{4}$$

$$0 \leq \sum_{i=1}^{4} lwh_i \times xp_{ih} \leq ep_h \tag{5}$$

$$0 \leq \sum_{i=1}^{4} lwh_i \times xg_{ih} \tag{6}$$

$$\sum_{h=sta_i}^{sto_i} xw_{ih} + xp_{ih} + xg_{ih} = oh_i \quad \forall i \in \{1, 2, 3, 4\} \tag{7}$$

The constraints in (2)-(7) are described as follows. Constraint (2) denotes that the load switching vectors are binary. The value is zero when it is switched off, and it is one when it is switched on. Constraint (3) is used to ensure that at any time interval, only one generator could supply the load. Constraint (4) expresses that at any time interval, the maximum wind energy consumed by all loads should be lower than or equal to the available energy generated by the wind turbine generator at that time. Constraint (5) expresses that at any time interval, the maximum solar energy consumed by all loads should be lower than or equal to the available energy generated by the PV generator at that time. Constraint (6) expresses that the grid will supply the energy to the loads when it could not be satisfied by the wind turbine generator or the PV generator. Constraint (7) expresses the total operation hours of the scheduled loads from the start time to the stop time.

3. Implementation.

3.1. Data exchange via Modbus TCP/IP.

As stated previously, the challenging task of our proposed system is in the incorporation between the Winlog SCADA software and the Delphi program, due to their different behavior. Since the Winlog SCADA and the Delphi should exchange the data on one computer, the data exchange method should be developed properly. In this work we propose a simple novel approach to exchange the data via the Modbus TCP/IP protocol. The mechanism of the data exchange is illustrated in Figure 2.

The Winlog SCADA requests the data to the slave periodically. There are two types of requests sent from the master. The first request is to write the data required to solve the optimization problem as expressed in (1)-(7). In the Modbus protocol, the slave will receive the data and send a response with the acknowledgment to the master. Upon receiving and saving the data completely from the master, the slave (Delphi program) calculates the LP technique and saves the results into the memory.

The second request is to read the load scheduling data, i.e., load switching vector, and the status from the slave. Since the request is polled periodically, the slave will send the data to the master, even though the LP calculation has not been finished yet. To

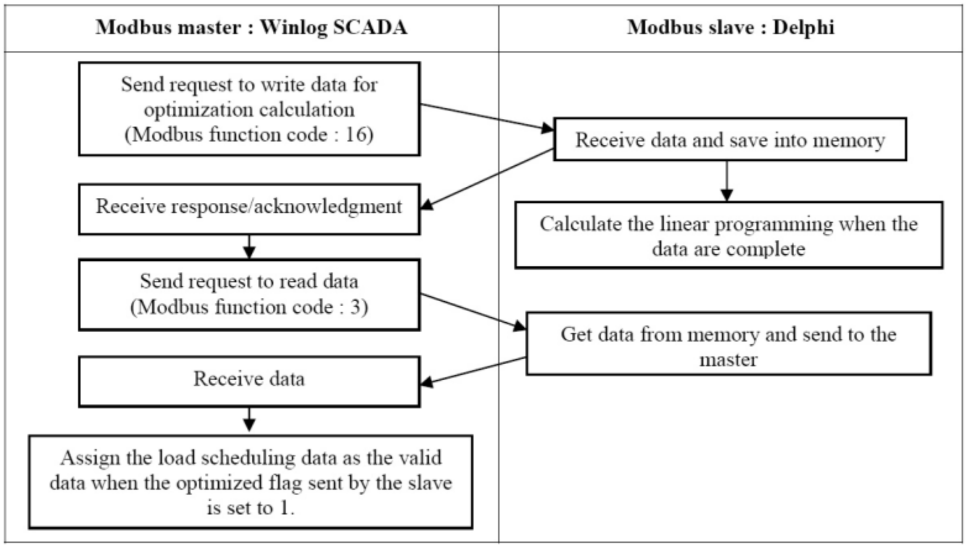


FIGURE 2. Data exchange between the Winlog SCADA and the Delphi

ensure that the resulted data are valid, the master should check the flag that represents the status of optimization calculation. Once the flag is set to one, the master confirms that the received load data scheduling is valid. The detailed mechanism is discussed in the following.

The data transfer is performed using Modbus function 3 (read holding registers) and 16 (write holding registers). The writing command is used by the master to send the data from the Winlog SCADA to the Delphi. The data consists of the hourly load power consumption, the hourly electricity prices, the hourly available energies, the total operation hours, the start and stop time of the scheduled loads. While the reading command is used by the master to read the data from the Delphi consisting of the load switching vectors, the state of optimization calculation, and the value of objective function. The data addressing on the Modbus registers is listed in Table 1.

TABLE 1. Data addressing on the Modbus registers

Address	Modbus function	Variable name
0 to 3 [4 regs.]	16 (write holding registers)	Hourly load power consumption
4 to 27 [24 regs.]	**3 (read holding registers)**	**Load switching vector**
30 to 53, 60 to 83, 90 to 113 [72 regs.]	16 (write holding registers)	Hourly electricity price
120 to 143, 150 to 173 [48 regs.]	16 (write holding registers)	Hourly available energy
210 to 213 [4 regs.]	16 (write holding registers)	Total operation hours of scheduled load
214 to 217 [4 regs.]	16 (write holding registers)	Start hour of scheduled load
218 to 221 [4 regs.]	16 (write holding registers)	Stop hour of scheduled load
226 [1 reg.]	**3 (read holding registers)**	**Status of optimization calculation**
227 to 228 [2 regs.]	**3 (read holding registers)**	**Value of objective function**

When the SCADA is running, it polls the Modbus registers (writing and reading) on a specific interval defined by the SCADA configuration. In this work, the interval is set to one second, i.e., the fastest interval allowed by the Winlog SCADA. It is the normal operation in the SCADA system. In this work, the Modbus communication is employed in the different way, i.e., to exchange the data between the Winlog SCADA and the Delphi program.

The optimization technique developed in this paper computes the LP for load scheduling in a day. It collects the 24 hours data from the day-ahead, and results in the 24 hours load scheduling represented by the hourly load switching vector. Thus, the Delphi program calculates the LP once a day. Since the data from the Winlog SCADA is sent periodically, the Delphi program should check the data readiness to start the LP calculation.

It is assumed that the Winlog SCADA has collected the 24 hours data and starts to send the data to the Delphi for LP calculation. According to Table 1, 136 registers starting from the address 0 will be written by the Modbus master (Winlog SCADA) to the Modbus slave (Delphi program). It will be written repeatedly in the specific interval. Since the time and duration for writing these 136 registers could not be known exactly, the Delphi program will count the numbers of written registers starting from the address 0. If the numbers reach 136 registers, then the busy flag will be set, and at the same time the LP calculation is started. During this time, the program ignores the contents of Modbus registers. Further, the status of optimization calculation in the Modbus register at address 226 is set to 0 as an indication that the calculation has not been finished yet.

After completing the LP calculation, the results are written to the corresponding registers, i.e., the load switching vector at address 4 to 27, and the value of objective function at address 227 to 228. To inform the Modbus master that the result is ready, the Modbus slave writes the status of 1 to the register address 226. Then the busy flag is reset, so the Delphi program is able to read the data for the next day calculation.

At the Modbus master side, the Winlog SCADA polls the register address 226. If the value is 1, the load switch vector in the register addresses 4 to 27 are valid; otherwise they are ignored. Then the Winlog SCADA sends the load switch vector to the local controllers for the load scheduling. Using the mechanism, it ensures that the data to be used in the LP calculation are valid; likewise the optimization result is read by the Winlog SCADA properly.

3.2. **Winlog SCADA.** In the work, the Winlog SCADA is used to visualize the load scheduling problem in the SG. The visualization covered in this work is used to confirm that the SCADA software could be incorporated with the high level programming for handling the complex optimization problems in the SG. In the actual implementation, the visualization could be extended to the other features of the SG system.

The developed HMI SCADA is illustrated in Figure 3. It consists of five main parts: the loads parameters, the wind turbine generator parameters, the PV generator parameters, the grid parameters, and the load switching status. In this work, for experimental purpose, the parameters are defined manually using the Winlog SCADA tool called the code builder.

The load switching vector is visualized by the switches on the middle of main display. Using the visualization, it is easier to monitor and debug the optimization technique. For instance, the load switching vector of xg_{ih} is illustrated in the middle left part of the figure. The switches on the first row represent the switches that connect the load-1 to the grid. While the columns represent the hour (hour-1 to hour-24). From the figure, we can see that the load-1 is supplied by the grid at hour-1 to hour-6 and hour-21 to hour-24.

FIGURE 3. HMI SCADA

3.3. **Delphi program.** The Delphi XE6 is one of the high level programming language that is employed in this work. The main function is to solve the optimization problem in the SG, which could not be solved by the SCADA software. The Delphi program is used to calculate the LP technique. Since the Delphi program communicates with the Winlog SCADA using the Modbus TCP/IP protocol, the protocol should be implemented in the Delphi program.

To solve the LP, we adopt ***Lpsolve***, an open source MILP program [21]. The ***Lpsolve*** provides the library that could be called from the Delphi language. Using the library, programming the LP becomes the easy and simple task. The effort is on converting the parameters from the Modbus registers to the variables and the data structure defined by the ***Lpsolve***. Fortunately, it requires the average programming skill.

Since Modbus protocol is the popular communication protocol, many Modbus libraries are available in the Internet. We adopt the Modbus library from [22], which supports the Delphi language. This library allows us to develop the Delphi program for accessing the Modbus registers easily.

4. **Experimental Results.** Several experiments are conducted to verify our proposed method. The objectives of experiments are: a) to measure the execution time of the LP calculation; b) to compare the daily electricity cost with and without optimization. The first objective is to ensure that our proposed method could be implemented in the real-time application, while the second objective deals with importance of applying the optimization technique in the SG.

In the experiments, the hourly electricity prices of the energy resources are taken from [4]. The hourly costs of the grid during 24 hours are listed in Table 2. The hourly costs

TABLE 2. Hourly electricity price of the grid [4]

Hour	Price ($/MWh)	Hour	Price ($/MWh)	Hour	Price ($/MWh)
1	85	9	100	17	105
2	80	10	110	18	100
3	70	11	130	19	105
4	70	12	140	20	110
5	70	13	130	21	120
6	70	14	95	22	130
7	80	15	100	23	120
8	90	16	105	24	100

TABLE 3. Hourly available energies of the wind turbine generator and the PV generator [4]

Hour	Available energy (Watt)		Hour	Available energy (Watt)	
	Wind	PV		Wind	PV
1	0	0	13	100	1,550
2	0	0	14	250	1,500
3	0	0	15	600	1,350
4	0	0	16	600	1,250
5	0	0	17	900	900
6	0	100	18	600	500
7	0	400	19	600	125
8	0	650	20	300	0
9	0	900	21	0	0
10	0	1,250	22	0	0
11	20	1,350	23	0	0
12	50	1,500	24	0	0

of wind turbine generator and the PV generator are fixed during 24 hours, i.e., 20$/MWh and 50$/MWh respectively. The hourly available energies of the wind turbine generator and the PV generator are listed in Table 3.

Five different loads patterns listed in Table 4 are tested in the experiments. A refrigerator (**RG**) represents the load-1, a washing machine (**WM**) represents the load-2, a personal computer (**PC**) represents the load-3, and a television (**TV**) represents the load-4. The loads pattern-2 (**P2**) differs from the loads pattern-1 (**P1**) in the total operation hours of the television. The loads pattern-3 (**P3**) differs from P1 in the hourly load power consumptions. The loads pattern-4 (**P4**) differs from P1 in the hourly load consumptions, and the total operation hours. The loads pattern-5 (**P5**) differs from **P4** in the stop hour of **WM**.

4.1. **LP execution time.** To measure the LP execution time, we divide the measurements into several terms, i.e., a) **Total_time**: time required from the start of command to calculate the LP until the result is ready (measured by the Winlog SCADA); b) **TR_time**: the time required by the Delphi program to receive/read the complete data (136 registers) from the Winlog SCADA for the LP calculation; c) **LP_time**: the time required by the Delphi program to execute the LP; d) **WT_time**: the wasted time occurring during the

Table 4. Loads patterns

		P1	P2	P3	P4	P5
Hourly load power consumption (Watt)	**RG**	180	180	100	200	200
	WM	500	500	300	400	400
	PC	100	100	150	200	200
	TV	180	180	100	125	125
Total operation hours	**RG**	24	24	24	24	24
	WM	2	2	2	3	3
	PC	4	4	4	8	8
	TV	5	8	5	6	6
Start hour	**RG**	1	1	1	1	1
	WM	4	4	4	4	4
	PC	5	5	5	1	1
	TV	1	1	1	17	17
Stop hour	**RG**	24	24	24	24	24
	WM	21	21	21	9	22
	PC	24	24	24	24	24
	TV	24	24	24	24	24

Table 5. Measurement of LP calculation time

	P1	P2	P3	P4	P5
TR_time (ms)	4,250	4,219	4,219	4,218	4,234
LP_time (ms)	168,140	122,250	16	516	22,437
WT_time (ms)	641	938	405	422	673
Total_time (ms)	173,031	127,407	4,640	5,156	27,344

data exchange. From the definition, the above terms could be expressed as

$$Total\_time = TR\_time + LP\_time + WT\_time \tag{8}$$

In the experiments, **Total_time**, **TR_time**, and **LP_time** are measured by the Winlog SCADA and the Delphi program, while **WT_time** is calculated using (8). It is noted that **WT_time** occurs due to the polling mechanism of the Winlog SCADA.

The measurement results of LP calculation time for five loads patterns are given in Table 5. The experiments are performed on a personal computer 3.4 GHz Intel Core i7 processor with 4 GB of RAM under 64-bit Windows 10 operating system. From the table, it is obtained that the time required to transfer the amount of 136 data from the Winlog SCADA to the Delphi program, i.e., **TR_time** is about 4 seconds. It is also obtained that **LP_time** varies according to the loads pattern. It means that the LP calculation time depends on the complexity of the data to be solved. The fastest time is achieved for solving the problem of **P3**. It requires the total execution time of 4.640 seconds. While the longest time is achieved for solving the problem of **P1**, which requires the total time of 173.031 seconds or 2.884 minutes.

Since the optimization problem to be solved deals with the hourly parameters, it should be solved below one hour. The result suggests that the execution time of our proposed method is suitable for implementing the given optimization technique.

4.2. **Optimized load scheduling.** To verify the results of the load scheduling, we compare the optimized load scheduling of the five loads patterns to the ones without optimization. In the optimized load scheduling, the loads are operated according to the schedule obtained by the LP. The non-optimized load scheduling means that the loads are operated arbitrarily. To simplify the experiments, the non-optimized load scheduling is obtained by operating the loads from the start hour consecutively until the total operation hours are fulfilled. For instance, the **WM** in **P1** is operated at hour-4 and hour-5, and the **TV** in **P3** is operated from hour-1 to hour-5.

The profile of wind turbine and PV generation is illustrated in Figure 4, where the lines with triangle and square markers represent the wind turbine generator and the PV generator, respectively. Since the hourly price of the wind generator is the lowest, the optimization technique prefers to draw the electricity from the wind generator. It is clarified by the experiments as discussed in the following.

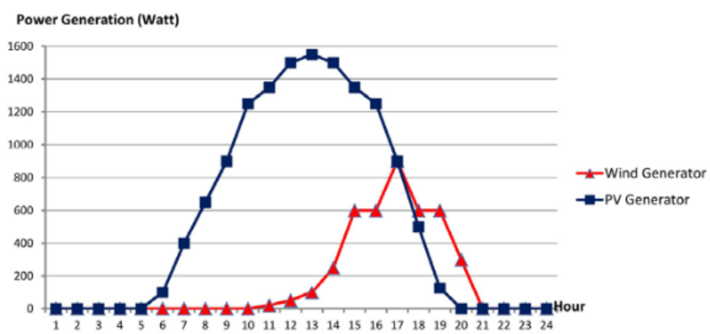


FIGURE 4. Profile of wind turbine and PV generation

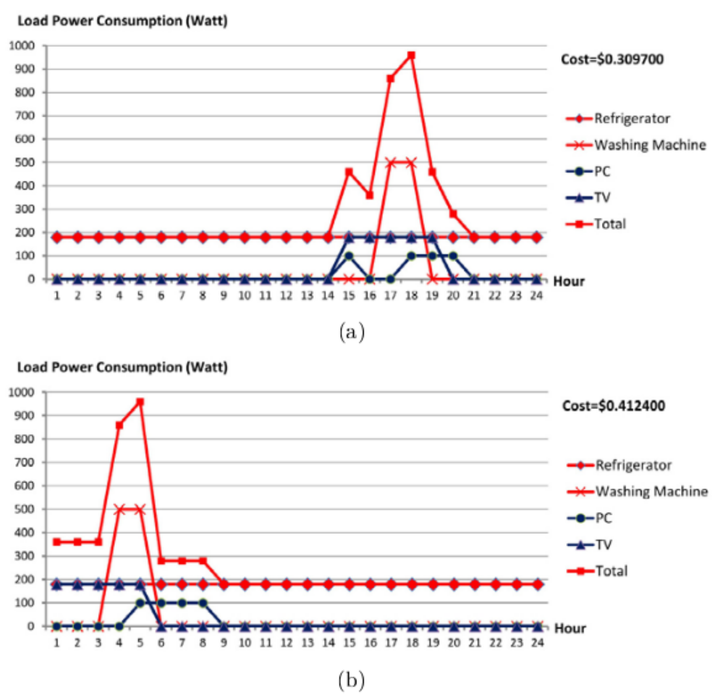


FIGURE 5. Load scheduling of **P1**: (a) optimized; (b) non-optimized

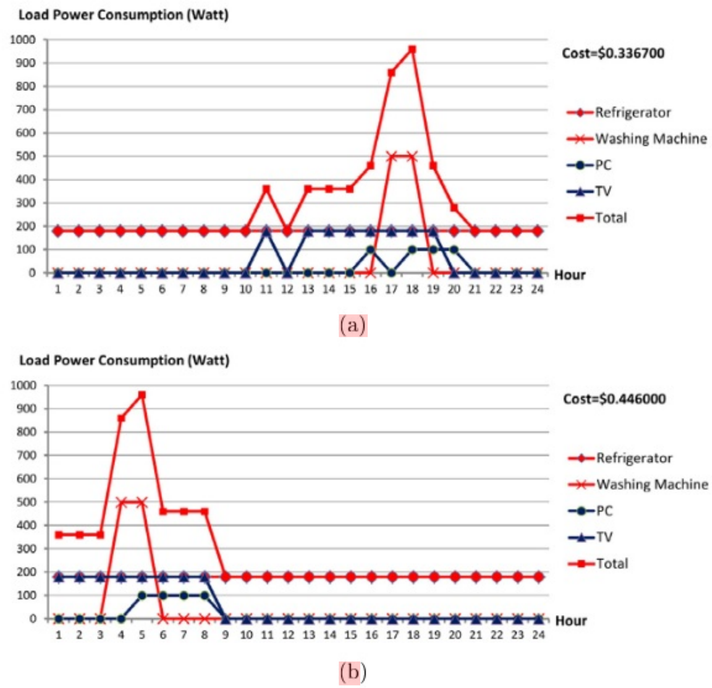


FIGURE 6. Load scheduling of **P2**: (a) optimized; (b) non-optimized

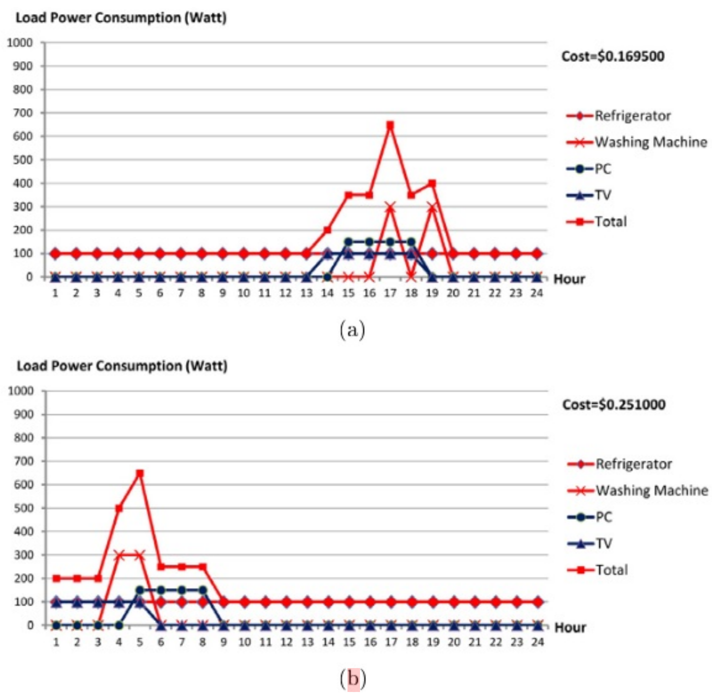
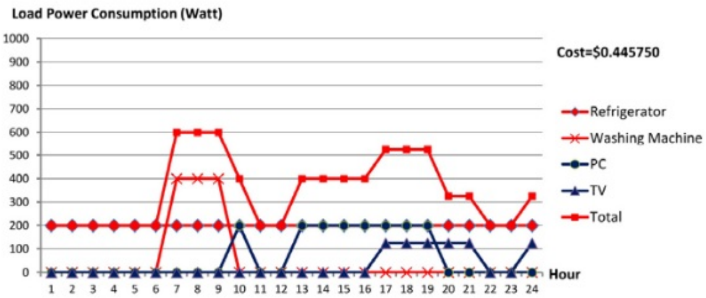
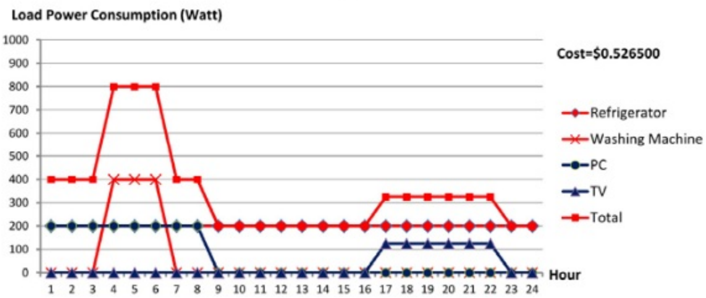


FIGURE 7. Load scheduling of **P3**: (a) optimized; (b) non-optimized

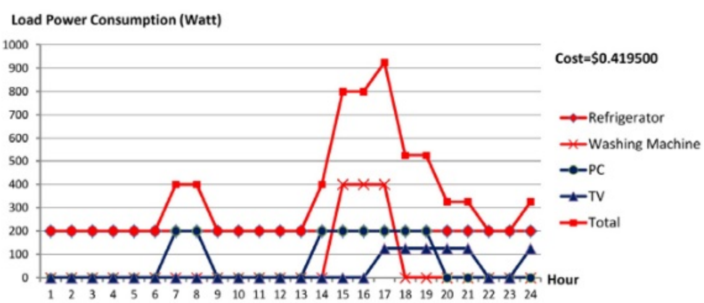


(a)

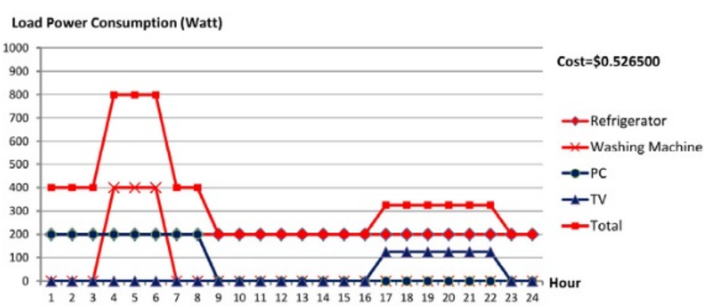


(b)

FIGURE 8. Load scheduling of **P4**: (a) optimized; (b) non-optimized



(a)



(b)

FIGURE 9. Load scheduling of **P5**: (a) optimized; (b) non-optimized

The load scheduling results of five loads patterns (**P1** to **P5**) are illustrated in Figure 5 to Figure 9. In the figure, the load consumption profiles of the **RG**, the **WM**, the **PC**, the **TV**, and the total loads are represented by the lines with rhombus, cross, circle, triangle, and square markers, respectively. In each figure, the upper picture (figure (a)) is the optimized load scheduling, while the lower picture (figure (b)) is the non-optimized load scheduling. The daily electricity cost is given in the figure.

From the results, it is obtained that for all five loads patterns, the daily electricity cost of the optimized load scheduling is lower than the one of the non-optimized load scheduling. From Figure 5 to Figure 9, it is clearly shown that the optimized load scheduling is obtained by shifting the operation of loads to the afternoon, when the wind turbine and the PV are available. In Figure 8, the optimized local scheduling shifts the operation of

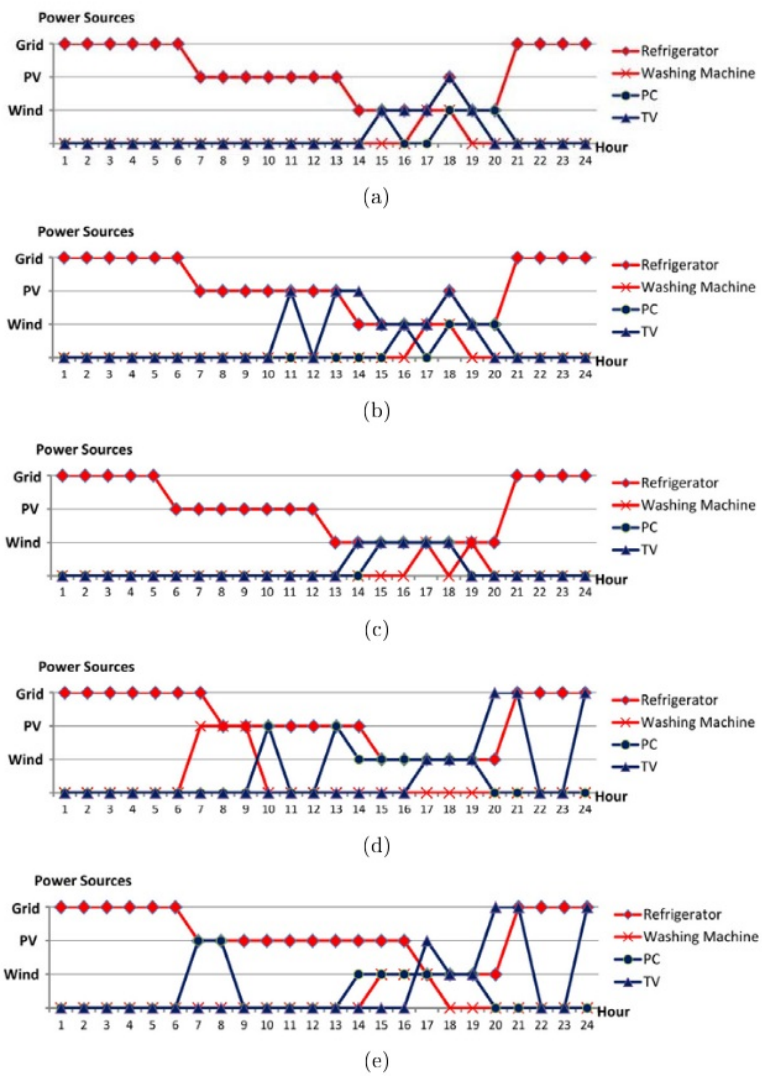


FIGURE 10. The optimized operation time of the energy resources of loads pattern: (a) **P1**; (b) **P2**; (c) **P3**; (d) **P4**; (e) **P5**

the washing machine to hour-7 to hour-9 in the morning, when the PV is available. It is caused by the fact that the stop hour of the washing machine in **P4** is at hour-9 (see Table 4).

The optimized operation time of the energy resources is illustrated in Figure 10, where Figure 10(a), Figure 10(b), Figure 10(c), Figure 10(d), and Figure 10(e) correspond to **P1**, **P2**, **P3**, **P4**, and **P5**, respectively. In the figure, the vertical axis represents the energy resources, i.e., the grid, the PV generator, and the wind turbine generator.

From Figure 10, it is obtained that the grid is operated when the wind turbine generator or the PV generator is not available, i.e., at the early morning and the night. For instance, let us observe Figure 10(a). In the case of refrigerator, it will be supplied by the PV generator from hour-7, where the PV generator is available. However, when the wind turbine generator is available at hour-14, it will take over the PV generator, due to its lower price.

5. Conclusion.

The incorporation of the Winlog SCADA and the Delphi program was proposed to solve and visualize the optimization problem in the SG. The data exchange between two applications was performed using the Modbus TCP/IP protocol. A simple mechanism was introduced to exchange the data properly. The experimental results showed that when the optimization problem is easy to be solved by the LP, the time for data exchange becomes a significant parameter. Otherwise, the LP calculation time played an important role in the calculation of the total computation time.

In future, the approach will be extended to solve the complex optimization problems, such as using the genetic algorithm. Further, the implementation in the real application will be conducted.

Acknowledgment. This work is supported by the Research Grant 2016, the Excellent Research on Higher Institution Scheme from Directorate General of Higher Education, Ministry of Research and Technology and Higher Education, Republic of Indonesia (SP/DIPA. 023.04.1.673.453/2016).

REFERENCES

[1] P. Siano, Demand response and smart grids – A survey, *Renewable and Sustainable Energy Reviews*, vol.20, pp.461-478, 2014.
[2] A. Barbato and A. Capone, Optimization models and methods for demand-side management of residential users: A survey, *Energies*, vol.7, no.9, pp.5787-5824, 2014.
[3] A. Molderink, V. Bakker, M. G. C. Bosman, J. L. Hurink and G. J. M. Smit, Domestic energy management methodology for optimizing efficiency in smart grids, *Proc. of IEEE PowerTech Conference*, Bucharest, Romania, pp.1-7, 2009.
[4] M. Jaradat, M. Jarrah, Y. Jararweh, M. Al-Ayyoub and A. Bousselham, Integration of renewable energy in demand-side management for home appliances, *Proc. of International Renewable and Sustainable Energy Conference*, Ouarzazate, Morocco, pp.571-576, 2014.
[5] T. Yu, D. S. Kim and S. Y. Son, Optimization of scheduling for home appliances in conjunction with renewable and energy storage resources, *Int. J. of Smart Home*, vol.7, no.4, pp.261-272, 2013.
[6] K. C. Sou, J. Weimer, H. Sandberg and K. H. Johansson, Scheduling smart home appliances using mixed integer linear programming, *Proc. of the 50th IEEE Conference on Decision and Control and European Control*, FL, USA, pp.5144-5149, 2011.
[7] A. Agnetis, G. de Pascale, P. Detti and A. Vicino, Load scheduling for household energy consumption optimization, *IEEE Trans. Smart Grid*, vol.4, no.4, pp.2364-2373, 2013.
[8] Z. Zhu, J. Tang, S. Lambotharan, W. H. Chin and Z. Fan, An integer linear programming based optimization for home demand-side management in smart grid, *Proc. of IEEE PES Conference on Innovative Smart Grid Technologies*, Washington DC, USA, pp.1-5, 2012.

[9] A. Soetedjo, A. Lomi and Y. I. Nakhoda, Implementation of optimization technique on the embedded systems and wireless sensor networks for home energy management in smart grid, *Proc. of IEEE Conference on Wireless Sensors*, Langkawi, Malaysia, 2016.
[10] M. H. Dhend and R. H. Chile, Innovative scheme for smart grid distribution SCADA system, *Proc. of IEEE the 2nd International Future Energy Electronics Conference*, Taipei, Taiwan, pp.1-6, 2015.
[11] S. H. Raman, M. A. M. Hanafiah, M. R. A. Ghani and W. N. S. E. W. Jusoh, A human machine interface (HMI) framework for smart grid system, *Proc. of IEEE Innovative Smart Grid Technologies – Asia Conference*, Kuala Lumpur, Malaysia, pp.318-322, 2014.
[12] H. G. Aghamolki, Z. Miao and L. Fan, A hardware-in-the-loop SCADA testbed, *Proc. of North American Power Symposium (NAPS)*, Charlotte, NC, USA, pp.1-6, 2015.
[13] A. Soetedjo, A. Lomi and Y. I. Nakhoda, Smart grid testbed using SCADA software and Xbee wireless communication, *Int. J. of Adv. Comp. Sci. and Appl.*, vol.6, no.8, pp.86-92, 2015.
[14] D. Bytschkow, M. Zellner and M. Duchon, Combining SCADA, CIM, GridLab-D and Akka for smart grid co-simulation, *Proc. of IEEE PES Conference on Innovative Smart Grid Technologies*, Washington DC, USA, pp.1-5, 2015.
[15] L. Gomes, F. Fernandes, T. Sousa, M. Silva, H. Morais, Z. Vale and C. Ramos, Contextual intelligent load management with ANN adaptive learning module, *Proc. of the 16th International Conference on Intelligent System Applications to Power Systems*, Crete, Greece, pp.1-6, 2011.
[16] F. Fernandes, H. Morais, P. Faria, Z. Vale and C. Ramos, SCADA house intelligent management for energy efficiency analysis in domestic consumers, *Proc. of IEEE PES Conference on Innovative Smart Grid Technologies Latin America*, Sao Paulo, Brazil, pp.1-8, 2013.
[17] L. Gomes, F. Fernandes, P. Faria, Z. Vale, C. Ramos and H. Morais, Contextual intelligent load management considering real-time pricing in a smart grid environment, *Proc. of the 7th IEEE International Conference on Intelligent System Applications to Power Systems*, Tokyo, Japan, 2013.
[18] F. Fernandes, M. Silva, P. Faria, Z. Vale, C. Ramos and H. Morais, Real-time simulation of energy management in a domestic consumer, *Proc. of the 4th IEEE PES Innovative Smart Grid Technologies Europe*, Copenhagen, Denmark, pp.1-5, 2013.
[19] *https://www.sielcosistemi.com/.*
[20] *http://www.modbus.org/.*
[21] *https://sourceforge.net/projects/lpsolve/.*
[22] *https://sourceforge.net/projects/delphimodbus/.*

Jurnal_Aryu-2017-2.pdf

ORIGINALITY REPORT

4%	2%	5%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Aryuanto Soetedjo, Abraham Lomi, Yusuf Ismail Nakhoda. "Implementation of optimization technique on the embedded systems and wireless sensor networks for home energy management in smart grid", 2016 IEEE Conference on Wireless Sensors (ICWiSE), 2016 Publication	3%
2	zdoc.site Internet Source	2%

Exclude quotes	On	Exclude matches	< 2%
Exclude bibliography	On		