2016 IEEE CONFERENCE ON WIRELESS SENSORS

ICWiSe

10TH – 12TH OCTOBER 2016

HOLIDAY VILLA RESORT

LANGKAWI, KEDAH

MALAYSIA

ISBN: 978-1-5090-1626-6

Part Number: CFP16QAA-ART

ORGANIZED BY

IEEE computer society
Malaysia

2016 IEEE Conference on Wireless Sensors (ICWiSE)

IEEE Catalog Number: CFP16QAA-ART
ISBN: 978-1-5090-1626-6

TABLE OF CONTENTS

Copyright Notice

Full Papers

IEEE
Advancing Technology
for Humanity

ICWiSe
IEEE Conference on Wireless Sensors

IEEE
computer
society
Malaysia Section

Certificate of Participaion

We hereby certify that

Aryuanto Soetedjo

has presented a paper titled

Implementation of Optimization Technique on the Embedded Systems and Wireless Sensor Networks for Home Energy Management in Smart Grid

at

ICWiSe 2016
Holiday Villa Beach Resort & Spa Langkawi, Malaysia
From 10 - 12 October 2016

Prof. Dr. Ali Selamat
Chair, IEEE Computer Society
Malaysia Section

Implementation of Optimization Technique on the Embedded Systems and Wireless Sensor Networks for Home Energy Management in Smart Grid

Aryuanto Soetedjo*
Dept. of Electrical Engineering
National Institute of Technology
Malang, Indonesia
aryuanto@gmail.com

Abraham Lomi
Dept. of Electrical Engineering
National Institute of Technology
Malang, Indonesia
abraham@itn.ac.id

Yusuf Ismail Nakhoda
Dept. of Electrical Engineering
National Institute of Technology
Malang, Indonesia
yusuf_nakhoda@yahoo.com

***Abstract*—This paper presents the implementation of optimization technique on the HEMS (Home Energy Management System). The objective of load scheduling optimization problem is to minimize the peak hourly load power consumption. The Raspberry Pi module is employed as the smart controller installed at a home. The smart controller is used to solve the optimization problem using MILP (Mixed Integer Linear Programming). It communicates with the load controllers implemented on the Arduino microcontroller over the ZigBee wireless network. The experiment results show that the proposed system is able to compute the MILP in real-time at 396 ms, very fast compared to the hourly interval used by the optimization technique. Further, the transmission time from smart controller to the local controller, and vice versa is 167 ms and 187 ms respectively.**

***Index Terms*—Energy management; linear programming; Raspberry Pi; Arduino; ZigBee.**

I. INTRODUCTION

The Smart Grid is the next generation technology where the components of the grid system, i.e. the generation system, the transmission system, the distribution system, and the consumers are intelligent [1]. In the Smart Grid, both the electrical and the information flow in bidirectional ways. It allows the consumers to sell the electricity to the grid.

Recently, researches on the Smart Grid increase significantly. The main topics are on the development of ICT infrastructure and algorithms on the energy management [2]. One of the interesting topics in the energy management is the demand side management (DSM), such as addressed in [2]-[7]. The effectiveness of the DSM in the Smart Grid framework requires the implementation of energy management in the home called as the home energy management systems (HEMS) [4]. The HEMS offers the users (costumers) to schedule the operation of home appliances for reducing the costs according to the DSM's scheme. In this research, we focus on the energy management for the home as proposed in [2]-[4], [6], [7].

In the HEMS, several optimization techniques have been proposed to provide the effective solution in the DSM [2]-[7]. The integer linear programming techniques were employed for optimizing the power consumption scheduling in the home [2], [7]. In [2], the optimization technique was proposed to minimize the peak hourly load. The mixed integer linear programming (MILP) technique was employed for scheduling the smart appliances to minimize the cost [7]. The comparative study of knapsack technique was conducted for optimizing the consumption of electricity in the HEMS [3]. The Genetic Algorithm (GA) was employed to solve the nonlinear optimization problem that combining the real-time pricing (RTP) and the inclining block rates (IBR) [4]. An offline method and an online method were proposed to minimize cost over the time [5]. The offline method is used to optimize the power generation, the power requirement, and duration are fixed and known. The online method is used for long-term cost minimization, where the model is stochastic. A power scheduling based communication protocol was proposed to reduce the peak demand within a home by allowing the smart appliances in the home communicate each other over a home area network (HAN) [6].

In the works described previously, the optimization techniques were simulated using the software running on a personal computer, such as MATLAB [2], [4], [7]. The implementations of HEMS on the embedded hardware were proposed in [8]–[10]. But they deal with the simple algorithm for energy management. The home energy management that controls the appliances by considering the tariff, the status of

978-1-5090-1626-6/16/$31.00 ©2016 IEEE

grid, and the state of charge (SOC) of battery was developed in [8].

In this paper, we implement the optimization technique using MILP on the embedded platform for minimizing the peak hourly load proposed by [2]. The MILP is a classical method to solve the linear optimization problem such as the load scheduling addressed in [2]. Since our aim is to explore the feasibility of the real implementation of optimization techniques on the embedded hardware, the linear optimization problem in [2] is a good example to be implemented. More complex problems which are effectively solved by the GA will be treated as the further work.

In the proposed system, the Raspberry Pi single board computer is employed as the smart controller (SC) for calculating the MILP. The SC collects data and sends the scheduling command to the local controllers (LCs) over the ZigBee wireless communication. The proposed system exhibits that the home energy management, more specifically the load scheduling using linear programming could be implemented in the real-time on the embedded platform and the wireless sensor networks (WSN).

The rest of the paper is organized as follows. Section 2 presents our proposed system. Section 3 discusses the experimental results. Conclusion is covered in Section 4.

II. PROPOSED SYSTEM

A. System Architecture

The proposed HEMS is illustrated in Fig. 1. It is assumed that the Smart Controller could communicate with the Smart Grid Center over the wireless communication. In this work, we focus on the smart home component, which consists of the Smart Controller (SC) and the Local Controllers (LCs). The SC communicates with the LCs over the ZigBee wireless network.

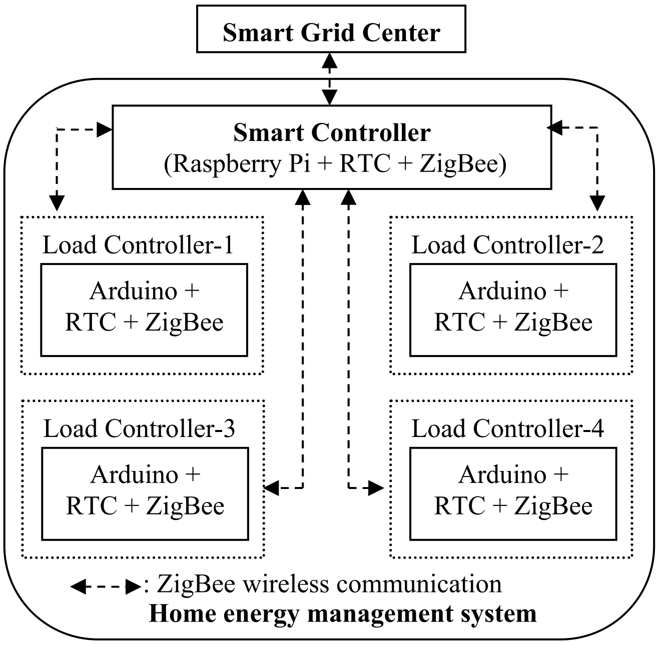


Fig. 1. Proposed HEMS.

The objective of energy management is to minimize the peak hourly load as proposed in [2]. The optimization algorithm is implemented on the SC as described in the following. The SC is the central controller of the HEMS. It consists of a Raspberry Pi module, a real-time clock (RTC) module, and a ZigBee module. The major functions of SC are: a) Receive the hourly load consumption of each load sent by the LCs; b) Compute the optimization technique; c) Send the command to LCs for switching on/off the appliances.

The LC is a smart node connected to load/appliance. It monitors the load power consumption and switch on/off the load according to the SC command. In the paper, only the main parts, i.e. an Arduino microcontroller module, a RTC module, and a ZigBee module are discussed. In the experiment, four LCs are employed to represent the different types of appliances [2], i.e: non shiftable load, power shiftable load, and time shiftable load. The details explanation of the SC and LC are discussed in the next sections.

B. Optimization Technique

The load scheduling optimization technique used in this paper is based on the method proposed by [2]. The main contribution of our work is to implement the optimization technique, i.e. MILP algorithm in the real-time using the embedded system. To simplify the problem, the optimization formula is modified in the following. Reader may refer to [2] for the original formula.

Let us define $\boldsymbol{a_h}$, $\boldsymbol{b_h}$, and $\boldsymbol{c_h}$ are the hourly load power consumption of load-1, load-2, and load-3 at hour-*h* respectively, where load-1 and load-2 are the non shiftable load, while load-3 is the power shiftable load. The preferred working hour of load-3 is defined as $[hw_s, hw_{s+1}, \ldots, hw_f]$. The minimum and maximum hourly power consumptions of load-3 are denoted as *cmin* and *cmax* respectively. The on/off status of load-4 at hour-*h* is defined as $\boldsymbol{s_h}$, where the value of 1 represents the load is switched on, and the value of 0 is switched off. Load-4 is the time shiftable load, and it is assumed that it operates one hour everyday with the power consumption is defined as ***ld***. The total load power consumptions for one day of load-1, load-2, and load-3 are defined as *la, lb,* and *lc* respectively.

The optimization problem is defined as

$$Min\,(a_t + b_t + c_t + lds_t), \qquad t \in [1,2,\ldots,24] \tag{1}$$

such that

$$(a_h + b_h + c_h + lds_h) \le (a_t + b_t + c_t + lds_t), \forall h \in [1,2,\ldots,24], h \ne t \tag{2}$$

$$\sum_{i=1}^{24} a_i = la \tag{3}$$

$$\sum_{i=1}^{24} b_i = lb \tag{4}$$

$$\sum_{i=1}^{24} c_i = lc \tag{5}$$

$$a_h \ge 0,\ \forall h \in [1,2,\ldots,24] \tag{6}$$

$$b_h \ge 0,\ \forall h \in [1,2,\ldots,24] \tag{7}$$

$$c\min \le c_h \le c\max,\ \forall h \in \left[hw_s, hw_{s+1}, \ldots, hw_f\right] \qquad (8)$$

$$0 \le s_h \le 1, s_h \in [0,1] \qquad (9)$$

Note that the above optimization problem is solved for every $t \in [1,2,\ldots,24]$ as expressed in (1). When there are several solutions with the same optimal value, the algorithm selects one of them as the solution. For practical reason, the solution with earlier hour is chosen.

C. Smart Controller

In our proposed system, the optimization technique expressed by (1)–(9) is implemented on the SC installed at a home. Since the optimization problem should be solved using the MILP, the SC should have a high computation power. Further, the computer system should be small in size for implementation on the HEMS. Therefore we choose the Raspberry Pi, a small size single board computer as the main processor on the SC.

The data required to solve (1)-(9) is collected from the LCs over the ZigBee wireless communication in the real-time. Thus the ZigBee module is attached to the Raspberry module to build the communication network with the LCs. In addition, the RTC module is added as the time reference to the whole system. The SC collects data from LCs every hour and sends the command to LCs in one hour interval based on the RTC time. To achieve the accurate timing, the SC synchronizes the RTC time to all LCs regularly. The SC module is illustrated in Fig. 2.

In the experiment, it is assumed that the measured data of hourly load power consumption is already stored on the Arduino memory. Then the Arduino sends the data to the SC every hour. When the Arduino receives the load scheduling command from the SC, it will switch on/off the load accordingly.

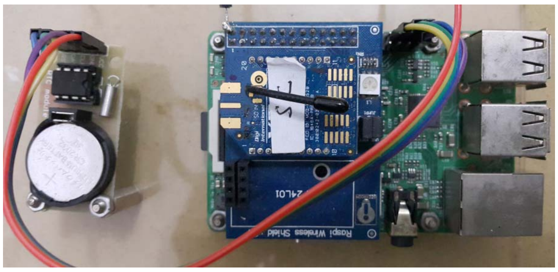


Fig. 2. Smart Controller module.

D. Local Controller

The LC is a module that is interfaced with the load/appliance. It reads the power consumption of the load, and switches on/off the load. In this paper, the electrical/electronic parts such as current and voltage sensors, the relays are not discussed. We focus on the digital part, i.e. the microcontroller module and the communication part.

The Arduino microcontroller module is employed as the main processor of the LC. It communicates with the SC using the ZigBee module. Similar to the SC, the RTC module is added to the microcontroller. The LCs modules are illustrated in Fig. 3.

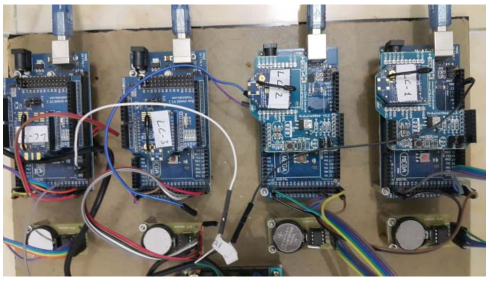


Fig. 3. Local Controllers modules.

E. Communication Protocol

The proposed system employs the ZigBee wireless network for communicating between the SC and the LCs. In the network, the SC module is assigned as the coordinator. While four LCs are assigned as the routers as shown in Fig. 4. The PAN-ID is set to 5, and the API mode is set to 2.

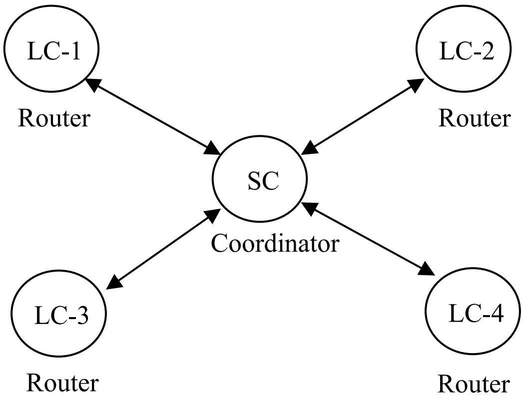


Fig. 4. ZigBee network.

In the proposed system, three modes are introduced, i.e.: a) Time synchronization mode; b) Learning mode; c) Running mode. These modes are sent by the SC to the LCs. In the real application, three modes run simultaneously. In the time synchronization mode, the SC sends the current time to be synchronized to all LCs. Upon receiving this command, the current time of LC is updated according to the time sent by the SC. It ensures that all RTCs in the system run synchronously.

In the Learning mode, every LC sends the hourly load power consumption to the SC every hour. The data consists of the number of hour (1 to 24) and the respective load power consumption. Since the destination is the SC, the LC uses the default destination address of the coordinator, i.e. 0000H.

According to (1)-(9), the 24 hourly data are required to compute the optimization technique. Thus the SC waits until all 24 hour data are collected from the LCs. It means that the optimized load scheduling is applied for one day ahead.

In the Running mode, the SC sends the optimized load scheduling to the LCs. The data consists of the number of hour (1 to 24) and the respective optimized load power. Since the data is unique for each LC, the SC sends the data to the LCs using their address differently. Since the ZigBee network uses the API mode, changing the destination address could be made easily on the fly.

III. EXPERIMENTAL RESULTS

The proposed system is tested on the real hardware as follows. The SC module: Raspberry Pi 2 Model B with a 900MHz quad-core ARM Cortex-A7 CPU; Xbee Pro Series S2B (ZigBee) with wire antenna, 63 mW; DS1302 RTC module. The LCs: Arduino Mega 2560; Xbee Series 2 (ZigBee) with wire antenna, 1 mW; DS1302 RTC module. The MILP is written using C++ language running on the Raspberry Pi with Raspbian operating system. The CBC (Coin-or branch and cut) library [11] is employed to solve the MILP.

We conduct several experiments to verify our proposed system. In the experiments, we measured the transmission time for sending and receiving data between the SC and the LCs over the ZigBee network, and the time required by the Raspberry Pi to compute the MILP. To speed up the testing process, the hourly data is sent from the LCs every minute, instead of every hour. It is noted here that the hourly data is already stored in the memory of Arduino. Also, the Raspberry Pi sends the optimized hourly load schedule to the LCs every minute, instead of every hour.

To measure the transmission time from the SC to the LC, we add the current time of SC while sending the data into the data packet which is sent to the LC. After receiving the data packet, the LC computes the time different between the time sent by the SC and the current time of LC while receiving the packet. The transmission time from LC to the SC is done similarly. The execution time of MILP is calculated directly from the program in the Raspberry Pi.

Four scenarios are tested during the experiments. Each scenario differs on the load power consumption profile as listed in Table 1 and Table 2. The load-1 and load-2 are the non shiftable loads, where the load-1 is represented by the refrigerator, and the load-2 is represented by the lighting. The load-3 is the power shiftable load, which is represented by the water boiler. The load-4 is the time shiftable load, which is represented by the washing machine.

The load power consumption profiles of scenario-1, scenario-2, scenario-3, and scenario-4 are plotted in Fig. 5, Fig. 6, Fig. 7, and Fig. 8 respectively. In the figures, figure (a) shows the non-optimized load power profile, while figure (b) shows the optimized one obtained by the optimization technique calculated on the SC. In the figure, blue line, green line, black line, orange line, and red line denote the load power profiles of load-1, load-2, load-3, load-4, and total hourly load respectively.

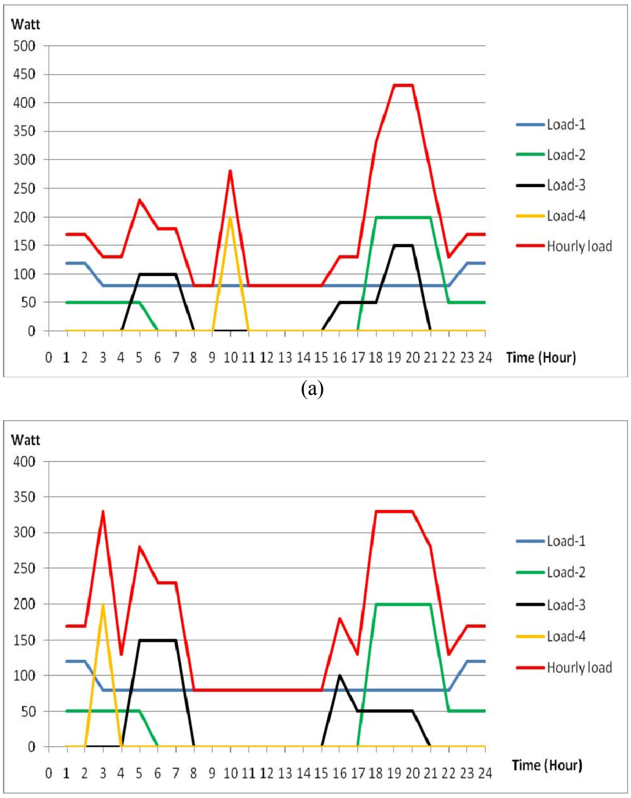
From Fig. 5(a), it is obtained that for non-optimized load scheduling, the peak hourly load is 430 watt which appears at 19th and 20th hours. Fortunately after it is optimized, the peak hourly load decreases to 330 watt which appears at 3rd, 18th, 19th, and 20th as shown in Fig. 5(b). It is also shown that the load power profiles of load-1 and load-2 on Fig. 5(a) and Fig. 5(b) are the same. It is caused by the fact that the load-1 and load-2 are the non shiftable loads, thus they are not affected by the optimization technique. While the profiles of load-3 and load-4 are changed by the optimization technique as shown in the figures.

TABLE I. LOAD POWER PROFILE: SCENARIO-1 AND SCENARIO-2

No	**Load-1 (Watt)**		**Load-2 (Watt)**		**Load-3 (Watt)**		**Load-4 (Watt)**	
	S1	**S2**	**S1**	**S2**	**S1**	**S2**	**S1**	**S2**
1	120	100	50	250	0	0	0	0
2	120	100	50	250	0	0	0	0
3	80	100	50	250	0	0	0	0
4	80	100	50	250	0	0	0	0
5	80	100	50	250	100	0	0	0
6	80	100	0	250	100	200	0	0
7	80	100	0	0	100	200	0	400
8	80	100	0	0	0	0	0	0
9	80	100	0	0	0	0	0	0
10	80	100	0	0	0	0	200	0
11	80	100	0	0	0	0	0	0
12	80	100	0	0	0	0	0	0
13	80	100	0	0	0	0	0	0
14	80	100	0	0	0	0	0	0
15	80	100	0	0	0	0	0	0
16	80	100	0	0	50	0	0	0
17	80	100	0	0	50	150	0	0
18	80	100	200	0	50	150	0	0
19	80	100	200	500	150	350	0	0
20	80	100	200	500	150	350	0	0
21	80	100	200	500	0	0	0	0
22	80	100	50	500	0	0	0	0
23	120	100	50	250	0	0	0	0
24	120	100	50	250	0	0	0	0

TABLE II. LOAD POWER PROFILE: SCENARIO-3 AND SCENARIO-4

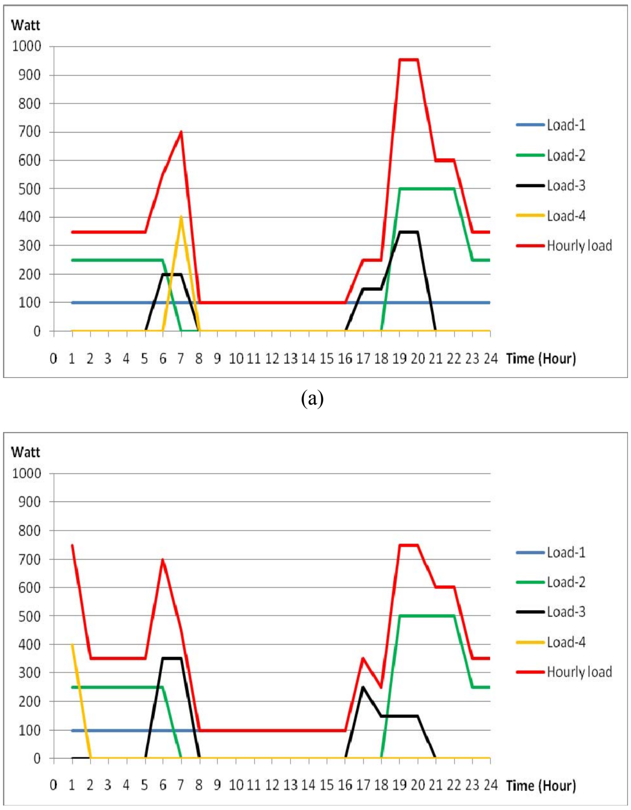
No	**Load-1 (Watt)**		**Load-2 (Watt)**		**Load-3 (Watt)**		**Load-4 (Watt)**	
	S3	**S4**	**S3**	**S4**	**S3**	**S4**	**S3**	**S4**
1	150	50	200	50	0	0	0	0
2	150	50	200	50	0	0	0	0
3	150	50	200	50	0	0	0	0
4	150	50	200	50	0	0	0	0
5	150	50	200	50	0	120	350	0
6	100	50	0	50	300	120	0	0
7	100	80	0	0	300	80	0	0
8	100	80	0	0	300	80	0	0
9	100	80	0	0	0	80	0	0
10	100	80	0	0	0	0	0	0
11	100	80	0	0	0	0	0	0
12	100	80	0	0	0	0	0	0
13	100	80	0	0	0	0	0	0
14	100	80	0	0	0	0	0	0
15	100	80	0	0	0	0	0	0
16	100	80	0	0	0	0	0	150
17	100	80	0	0	0	0	0	0
18	100	80	0	100	400	100	0	0
19	100	80	0	200	400	100	0	0
20	100	80	300	200	400	100	0	0
21	150	80	500	200	0	0	0	0
22	150	80	500	100	0	0	0	0
23	150	80	300	50	0	0	0	0
24	150	80	300	50	0	0	0	0



(a)

(b)

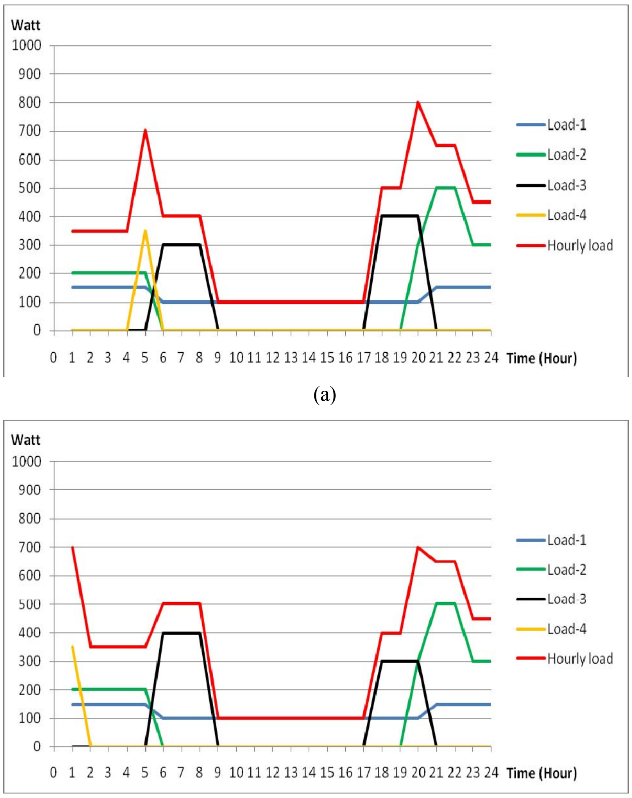
Fig. 5. Load power profile of scenario-1: (a) Non optimized; (b) Optimized.



(a)

(b)

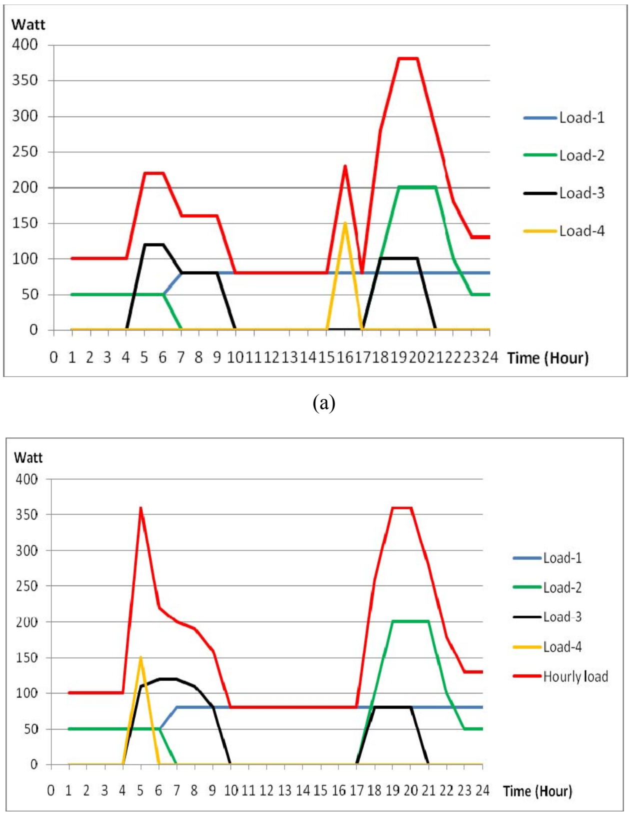
Fig. 6. Load power profile of scenario-2: (a) Non optimized; (b) Optimized.



(a)

(b)

Fig. 7. Load power profile of scenario-3: (a) Non optimized; (b) Optimized.



(a)

(b)

Fig. 8. Load power profile of scenario-4: (a) Non optimized; (b) Optimized.

The similar results are obtained for scenario-2, scenario-3, and scenario-4, where the peak hourly loads of the optimized load scheduling are lower compared to the ones of the non-optimized load scheduling. The peak hourly load decreases from 950 watt to 750 watt in scenario-2 (see Fig. 6), from 800 watt to 700 watt in scenario-3 (see Fig. 7), from 380 watt to 360 watt in scenario-2 (see Fig. 8).

TABLE III. AVERAGE OF TRANSMISSION TIME AND EXECUTION TIME

No.	Parameter	Average
1.	Transmission time from LC to SC	187 ms
2.	Transmission time from SC to LC	167 ms
3.	MILP execution time	396 ms

The results of transmission time from LC to SC, and vice versa, and the MILP execution time are listed in Table 3. The transmission times of 187 ms and 167 ms suggest us that the proposed system could be realized in the real condition for the real-time load scheduling. Further the MILP execution time is also lower than 1 second. Thus it is fast enough for implementing the optimization technique proposed here, where the optimized variables are processed hourly.

It is worthy to say that even though the optimization technique proposed in the paper is the simplified version of the one proposed in [2], the proposed system provides us the promising method for implementing such optimization techniques in real-time.

IV. CONCLUSION

The MILP is implemented on the embedded system to solve the optimization problem in HEMS. The capability of the proposed system to run in the real-time is tested by several experiments, and achieves the promising results. The optimization technique implemented on the Raspberry module could optimize the load power scheduling properly. The wireless communication between the Raspberry and the local controller could transfer the data around the hundredth millisecond. This speed is suitable for our current application.

In future, the system with the large numbers of loads will be developed. Further, the proposed system could be employed to implement the other optimization techniques.

ACKNOWLEDGMENT

This work is supported by the Research Grant 2016, 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] A. Khanna, "Smart Grid, Smart Controller and Home Energy Automation-Creating the Infrastructure for Future," Smart Grid and Renewable Energy, Vol. 3, pp. 165-174, 2012.

[2] Z. Zhu, J. Tang, S. Lambotharan, W.H. Chin, Z. Fan, "An Integer Linear Programming Based Optimization for Home Demand-side Management in Smart Grid," Proceedings of IEEE PES Innovative Smart Grid Technologies (ISGT), Washington DC, USA, 2012.

[3] A. Arbaz, M. Waqas, K. Shehzad, A. Mahmood, "Home Energy Management and Knapsack Technique in Smart Grid Environment," Proceedings of 2nd International Conference on Engineering & Emerging Technologies (ICEET), Lahore, Pakistan, 2015.

[4] Z. Zhao, W.C. Lee, Y. Shin, K.B. Song, "An Optimal Power Scheduling for Demand Response in Home Energy Management System, " IEEE Transactions on Smart Grid, Vol. 4, No. 3, pp. 1391-1400,2013.

[5] I. Koutsopoulos, L. Tassiulas, "Control and Optimization Meet the Smart Power Grid: Scheduling of Power Demands for Optimal Energy Management," Proceedings of 2nd International Conference on Energy-Efficient Computing and Networking, New York, USA, 2011.

[6] G. Xiong, C. Chen, S. Kishore, A. Yener, " Smart (In-home) Power Scheduling for Demand Response on the Smart Grid," Proceedings of IEEE PES Innovative Smart Grid Technologies (ISGT), California, USA, 2011.

[7] K.C. Sou, J. Weimer, H. Sanberg, K.H. Johansson, "Scheduling Smart Home Appliances Using Mixed Integer Linear Programming," Proceedings of 50th IEEE Conference on Decision and Control and European Control Conference, Florida, USA. 2011.

[8] A. Tascikaraoglu, M. Uzunoglu, M. Tanrioven, A. R. Boynuegri, O. Elma, "Smart Grid-Ready Concept of a Smart Home Prototype: A Demonstartion Project in YTU," Proceedings of 4th International Conference on Power Engineering, Energy and Electrical Drives, Istanbul. Turkey, 2013.

[9] P. Han, J. Wang, Y. Han, Q. Zhao, "Novel WSN-Based Residential Energy Management Scheme in Smart Grid," Proceedings of IEEE International Conference on Information Science and Technology, Hubei, China, 2012.

[10] M. Kuzhu, M. Pipattanasomporn, S. Rahman, "Hardware Demonstration of a Home Energy Management System for Demand Response Applications, " IEEE Transactions on Smart Grid, Vol. 3, No. 4, pp. 1704-1711, 2012.

[11] https://projects.coin-or.org/Cbc.