Light-Weight Stackelberg Game Theoretic Demand Response Scheme for Massive Smart Manufacturing Systems

Chunghyun Lee, Laihyuk Park and Sungrae Cho
School of Computer Science and Engineering
Chung-Ang University
221 Heukseok, Dongjak, Seoul 156-756, Korea
chlee@uclab.re.kr (C.L.); lhpark@uclab.re.kr (L.P.); srcho@cau.ac.kr (S.C.)

***Abstract*—As Internet-of-things (IoT) technology emerges, smart manufacturing has recently attracted a large amount of attention. Smart manufacturing leads to smart energy management because of its significant operating expenditure savings. However, it is believed that the centralized energy management of IoT devices would impose a critically large overhead since massive numbers of IoT devices are expected to be deployed. Therefore, distributed energy management or demand response is deemed to be a better solution for emerging massive smart manufacturing systems. There have been a significant number of distributed demand response algorithms, including Stackelberg game theoretic approaches. However, Stackelberg game theoretic approaches require a large number of iterations to reach Nash equilibrium, which in turn necessitates communication overheads among IoT devices. This communication overhead causes a large amount of energy consumption as well as delay. In this paper, we propose a light-weight demand response scheme based on the Stackelberg model without iterations for massive smart manufacturing systems. The proposed scheme manages energy consumption based on a non-iterative Stackelberg model and historical real time pricing. To the best of our knowledge, our approach is the first technique that considers communication overheads for the demand response technique. The performance evaluation demonstrates that the proposed scheme shifts operations to avoid peak loads, the electricity bill is significantly reduced, operations occur at preferred times, and communication energy consumption and delay are minimized.**

***Index Terms*—Smart manufacturing, smart grid, distributed demand response, Stackelberg game model, communication overhead.**

I. Introduction

Traditional manufacturing has critical problems, including long downtime, inefficient asset utilization, labor inefficiency, scheduling inaccuracy, and inefficient energy consumption [1]. Due to the proliferation in IoT technology, smart manufacturing has emerged, disrupting the legacy of traditional manufacturing [2]. For instance, the Smart Manufacturing Leadership Coalition (SMLC) proposed an open smart manufacturing platform based on IoT technology [3] [4]. According to the SMLC report [3], energy efficiency in smart manufacturing systems is expected to increase by more than 25% with energy management or demand response (DR) [5] [6].

Considering that smart manufacturing systems are extensive energy consumers that have an already implemented infrastructure endowed with a massive number of IoT devices, a larger amount of DR in this sector is expected compared to the residential and commercial sectors [7].

DR schemes can be generally divided into *centralized* or *distributed* schemes depending on the location of the DR algorithm, i.e., whether the DR is implemented within a central entity or in each IoT device, respectively [8]. In centralized DR schemes, it is almost impossible for a central entity such as an energy management system (EMS) to receive and manage real time information from more than 10,000 IoT devices in real time. In this context, a distributed DR scheme that can determine energy scheduling using individual IoT devices is adequate for information exchange in massive smart manufacturing systems.

Distributed DR schemes have been proposed that use game theory (related work on game-theoretic DR is well discussed in [5]). Among these approaches, the Stackelberg game model has been paid a great deal of attention. This model fits well the distributed DR scheme, where the leader (the power retailer) and followers (IoT devices) compete for profits [9].

The existing Stackelberg game models have only demonstrated improved scheduling performance through iterative calculation for a small number of power consumers and do not consider extensive iterative strategy exchanges between a leader and followers. However, in massive manufacturing systems, such models have to include communication overheads since iterative calculation of the Stackelberg game model requires communication between the power retailer (the leader) and IoT devices (followers) for exchanging strategies.

These communication overheads result in extra power consumption in response to iterative strategy exchanges and will significantly delay the manufacturing operation. In manufacturing systems, in particular, the operation delay can raise the production cost and these systems require a product manufacture deadline [10]. Since material requirements and production planning are systematically based on successive planning of a deadline, it is very important to reduce the communication overhead in the manufacturing systems [11]. Therefore, the DR scheme based on the Stackelberg game has to consider the communication overhead for the stable operation of smart manufacturing systems.

In this paper, we propose the distributed demand response

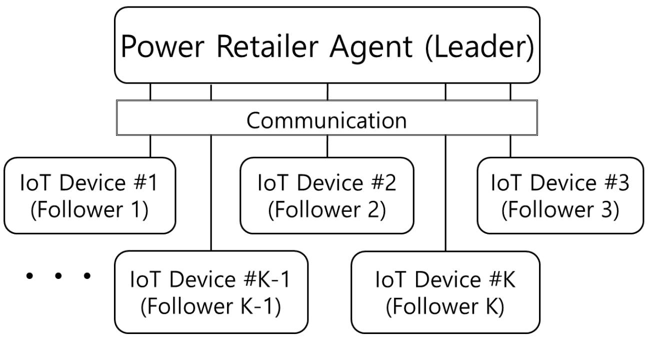


Fig. 1: Proposed architecture for a massive smart manufacturing system.

scheme with the light-weight Stackelberg game theoretic approach without an iterative calculation process for massive smart manufacturing systems. Since typical IoT devices perform similar tasks every day, abolishing the iterative calculations among IoT devices causes significant energy savings. As shown in Fig. 1, the system consists of a power retailer agent and multiple IoT devices, which can communicate with each other. In order to distribute the energy demand, the power retailer agent calculates an *estimated* day-ahead real time pricing (RTP) based on the historical energy consumption data of the IoT devices and then announces the RTP. The IoT device schedules the power demand and operating time based on the convenience of the RTP. In the proposed game, the power retailer agent plays the role of a leader to disperse the power demand, and the IoT device plays the role of a follower who wants to maximize its utility. In addition, the proposed game is a non-cooperative competition since each IoT device does not share its strategies.

The proposed scheme makes the following contributions.

- In a massive environment with more than 10,000 IoT devices, the proposed demand response scheme can reduce communication overheads that occurred by unnecessary iterative calculations. The existing schemes perform a tremendous amount of communication for iteratively exchanging strategies. Iterative communication is not necessary since significant overhead occurs and conventional manufacturing tasks are nearly static. However, the proposed scheme exchanges strategies between the power retailer agent and IoT devices only once per day. Therefore, the communication overheads are significantly reduced through the proposed scheme.
- We demonstrate that the proposed scheme can achieve near-optimal scheduling based on day-ahead RTP data. Initially, there is insufficient data on the power consumption pattern, causing a cold start problem where an appropriate peak load shift is not achieved. However, as the data accumulate gradually, the problem quickly converges to a near-optimal strategy, showing high-level performance. As a result, a performance of the proposed scheme works as well as the existing schemes.

This paper is organized as follows. Section II describes the proposed system model, basic assumption, and objective function of the power retailer agent and IoT devices. Section III provides the light-weight Stackelberg game theoretic approach model. Section IV presents the reduction of communication overheads compared with the existing scheme and the scheduling performance in a massive environment. Section V provides the conclusion and future works.

II. System Model

In this section, we formulate the mathematical model of DR for massive smart manufacturing systems. The proposed system is modeled on both the supply and demand sides.

A. Basic Assumption

In the energy prediction system model, environmental factors such as weather and issues are important. In fact, energy prediction models considering environmental factors have been extensively studied [12], [13]. However, this study does not consider environmental factors and issues since it aims to reduce communication overhead by eliminating iteration of the strategy exchange process in the existing Stackelberg model.

In the proposed system, we assume that there are a finite number of IoT devices, as shown in Fig. 1. We assume that IoT device set $\mathcal{A}$ consists of K appliances and is described by

$$\mathcal{A} = \{a_1, a_2, \cdots, a_k, \cdots, a_K\}. \quad (1)$$

Moreover, we assume that the time is divided into equal timeslots t and is described by

$$\mathcal{T} = \{1, 2, 3, \cdots, t, \cdots, T\}. \quad (2)$$

The timeslot can be represented by any unit of time. In this paper, we select hours for simplicity.

In the proposed system, the power retailer agent uses the RTP to disperse the power demand. We denote by p^t the RTP of time slot t. We assume that the RTP was sent from the power retailer agent to all IoT devices at the end of the previous day. Then, each IoT device schedules its operation time based on the RTP. Denote by $\mathcal{S}_k$ the energy scheduling vector of IoT device a_k, and it is represented by

$$\mathcal{S}_k = [s_k^1, s_k^2, s_k^3, \cdots, s_k^T], \quad (3)$$

where

$$s_k^t = \begin{cases} 1, & \text{if IoT device } a_k \text{ is operating at } t, \\ 0, & \text{otherwise.} \end{cases}$$

In this paper, we assume that IoT device a_k consumes energy E_k^{on} per each timeslot t during operation. Moreover, we assume that a_k has a preferred begin time and a preferred end time, which can be represented as C_k^{begin} and C_k^{end}, respectively, where $C_k^{begin} < C_k^{end}$. Denoting by ρ_k the required number of timeslots of device a_k, we can derive the following constraint:

$$\sum_{t=1}^{T} s_k^t = \rho_k, \forall k. \quad (4)$$

Moreover, we derive the following constraint since the preferred time section of device a_k should be less than or equal to the operation time:

$$C_k^{end} - C_k^{begin} + 1 \leq \rho_k. \tag{5}$$

Following the previous notation, denote by pr_k the profile of IoT device k, which is expressed by

$$pr_k = [E_k^{on}, C_k^{begin}, C_k^{end}, \rho_k], \tag{6}$$

where ρ_k is the required operation timeslot for a_k.

B. Supply Side Model

As mentioned earlier, the retailer agent stores the statistics of energy consumption historically with a simple moving average method [14]. Analysis of the historical power consumption pattern to predict the next power consumption is widely used [15].

Denote by $\overline{E_N^t}$ the average energy consumption of timeslot t at the Nth day, which can be obtained by

$$\overline{E_N^t} = \frac{(N-1) \cdot \overline{E_{N-1}^t} + E_N^t}{N}, \tag{7}$$

where E_N^t is the measured electricity consumption at timeslot t for day N. The retailer agent determines RTP p_N^t based on $\overline{E_{\mathcal{A}}^t}$ using the following quadratic equation [16]:

$$p_N^t = \alpha^t \cdot \overline{E_N^t}^2 + \beta^t \cdot \overline{E_N^t} + \gamma^t, \tag{8}$$

where $\alpha^t \geq 0$, $\beta^t \geq 0$, and $\gamma^t \geq 0$ at each timeslot $t \in \mathcal{T}$.

Denote by $\mathcal{P}_N$ the day-ahead RTP of the Nth day, described by

$$P_N = [p_N^1, p_N^2, p_N^3, \cdots, p_N^T]. \tag{9}$$

Algorithm 1 Day-ahead RTP Decision

1: ***Estimate Energy Consumption***
2: **if** $N = 1$ **then**
3:     **for** $t = 1 \rightarrow 24$ **do**
4:         $\overline{E_0^t} \Leftarrow 0$
5:     **end for**
6:     **for** $k = 1 \rightarrow K$ **do**
7:         **for** $t = C_k^{begin} \rightarrow C_k^{end}$ **do**
8:             $\overline{E_0^t} \Leftarrow \overline{E_0^t} + E_k^{on}$
9:         **end for**
10:     **end for**
11: **else if** $N > 1$ **then**
12:     Update E_{N-1}^t
13:     Calculate $\overline{E_{N-1}^t}$ by (7)
14: **end if**
15: ***Decision of Day-ahead RTP***
16: **for** $t = 1 \rightarrow T$ **do**
17:     Calculate p_N^t by (8)
18: **end for**
19: Return P_N

Algorithm 1 shows the day-ahead RTP decision algorithm. In order to calculate the day-ahead RTP, the estimated energy consumption is calculated based on the historical electricity consumption. However, we assume that on the first day, there are no historical data for electricity consumption. We assume that each IoT device operates during its preferred time section on the first day. Therefore, the expected electricity consumption is calculated based on the profiles of the IoT devices (Lines 2-10 in Algorithm 1). From the second day onward, the energy consumption is estimated based on realistic measured electricity consumption using (7) (Lines 11-14 in Algorithm 1). Based on the estimated energy consumption, the RTP can be calculated by (8) (Lines 16 and 17 in Algorithm 1). Then, the power retailer agent sends the RTP to the IoT devices (Line 19 in Algorithm 1).

C. Demand Side Model

Using the notation in Section II-A, we formulate the IoT devices modeling in this section. The IoT device schedules its operation time considering the electricity bill and its own convenience. Denote by $\mathcal{B}_k$ the electricity bill of IoT device a_k, which is calculated by

$$\mathcal{B}_k = \frac{1}{\tau_B} \sum_{t=1}^{\mathcal{T}} (p_N^t \cdot s_k^t \cdot E_k^{on}), \tag{10}$$

where p_N^t is the RTP of timeslot t for day N and τ_B is the scaling denominator which is an expected maximum electricity bill per timeslot for B_k [16].

As mentioned in Section II-A, we assume that an IoT device has preferred begin operation and end operation times. We assume that the user will be satisfied if the operation time scheduled by the power retailer occurs between these preferred times. Moreover, we assume that the satisfaction decreases as the scheduling result deviates from these preferred times. Denote by w_k^t the degree of user dissatisfaction for IoT device k, which represents how much the scheduled time differs from the preferred time, described by

$$w_k^t = \begin{cases} C_k^{begin} - t, & t < C_k^{begin}, \\ 0, & C_k^{begin} \leq t < C_k^{end}, \\ t - C_k^{end}, & C_k^{end} \leq t. \end{cases} \tag{11}$$

Denote by $\mathcal{D}_k$ the user dissatisfaction of the scheduling result, which we can obtain based on the user dissatisfaction degree as follows [16]:

$$\mathcal{D}_k = \frac{1}{\tau_D} \sum_{t=1}^{T} \frac{w_k^t \cdot s_k^t}{\rho_k}, \tag{12}$$

where the reason for dividing by ρ_k is that we assume the dissatisfaction of IoT device is less sensitive if it has a long operation time and τ_D is scaling denominator, which is an expected maximum user convenience per timeslot for D_k [16].

III. Problem Formulation: The Stackelberg Game

A. Problem Definition

In this section, we formulate the optimization problem and apply the Stackelberg game based on the modeling in Section II.

We assume that benefit of the power retailer agent is maximization of the electricity bill value. Therefore, the utility function of the power retailer agent is defined as

$$\mathcal{U}_R(\mathcal{P}, \mathcal{S}_K) = \sum_{k=1}^{K} \sum_{t=1}^{T} \frac{1}{\tau_B} \cdot E_k^{on} \cdot s_k^t \cdot p^t. \tag{13}$$

Since the power retailer agent wants to maximize its benefit, the optimization problem is formulated as follows:

$$\underset{\mathcal{P}}{\text{maximize}} \; \mathcal{U}_R(\mathcal{P}, \mathcal{S}_K), \tag{14}$$

subject to

$$\sum_{t=1}^{T} s_k^t = \rho_k, \forall k. \tag{15}$$

According to this optimization problem, the power retailer agent will increase the RTP when more power consumption is expected.

On the IoT device side, the benefit is defined as reduction of the electricity bill and dissatisfaction. From (10) and (12), we derive the utility function of IoT device k as follows:

$$\mathcal{U}_A(\mathcal{S}_k, \mathcal{P}) = -\sum_{t=1}^{T} \left(\frac{1}{\tau_B} \cdot E_k^{on} \cdot s_k^t \cdot p^t + \frac{1}{\tau_D} \cdot \frac{w_k^t \cdot s_k^t}{\rho_k}\right). \tag{16}$$

Since the IoT device wants to maximize its benefit, the optimization problem is defined as follows:

$$\underset{\mathcal{S}_k}{\text{maximize}} \; \mathcal{U}_A(\mathcal{S}_k, \tilde{P}), \tag{17}$$

subject to

$$\sum_{t=1}^{T} s_k^t = \rho_k, \forall k. \tag{18}$$

Therefore, the IoT device schedules its operation time using this optimization problem. Moreover, computing is possible on IoT devices with low computing performance since the objective function is standard convex form.

The derived optimization problems (14) and (17) together form the *Stackelberg game,* i.e., the power retailer agent and each IoT devices takes on the role of leader and a follower respectively. Moreover, the strategies of the followers can be affected by the leader's strategy.

Algorithm 2 Light-weight Stackelberg Game Theoretic

End of Day $(N-1)$

1) The retailer broadcasts day-ahead RTP p_t to each IoT device a_k.
2) Each IoT device a_k calculates $\underset{S_k}{\text{maximize}} \; U_A(S_k^*, \tilde{P}^*)$.
3) Each IoT device transmits the result to the retailer.

Day (N)

1) Each IoT device operates during its scheduled timeslot.

Algorithm 2 shows the proposed light-weight Stakelberg scheme. At the end of each day, the power retailer agent calculates the RTP using Algorithm 1 and announces it to all IoT devices. Then, each IoT device schedules its operation time using objective function (17) and sends the result of its scheduling to the power retailer agent.

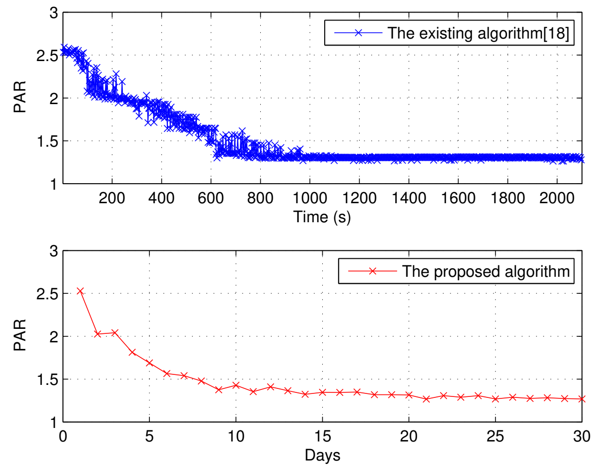


Fig. 2: Similarity between the proposed algorithm and the existing algorithm.

The proposed light-weight algorithm exchanges the strategies once a day. Since the proposed algorithm determines the strategies based on accumulated power consumption data, it is typical Stackelberg game in the long term. Therefore, the theoretical basis of the proposed algorithm is not as problematic as that of the existing algorithms. As shown in Fig. 2, the shape of the existing algorithm graph and that of the proposed algorithm graph are very similar.

B. Stackelberg Equilibrium

In the proposed game, the Stackelberg equilibrium (SE) is defined as follows. Let $\mathcal{P}^*$ be the best response for the optimization problem of the power retailer agent and $\mathcal{S}_k^*$ be the best response for IoT device k. Then, point $(\mathcal{P}^*, \mathcal{S}_k^*)$ is a SE for the proposed game if for any $(\mathcal{P}, \mathcal{S}_k)$ with $\mathcal{P} \geq 0$ $\mathcal{S}_k \geq 0$, the following conditions are satisfied:

$$\mathcal{U}_R(\tilde{P}^*, \mathcal{S}_K^*) \geq \mathcal{U}_R(\tilde{P}, \mathcal{S}_K^*) \tag{19}$$

and

$$\mathcal{U}_A(\mathcal{S}_k^*, \tilde{P}^*) \geq \mathcal{U}_A(\mathcal{S}_k, \mathcal{P}^*), \forall k. \tag{20}$$

Generally, the SE can be obtained by finding its subgame-perfect NE. However, in the proposed game, each IoT device considers only the power retailer agent, and there is no strategy exchange or competition among the IoT devices. In these non-cooperative and competitive games, we find empirical stability through the average value of the best policies determined by cumulative best responses rather than by deriving NEs. As shown in (17), the objective function of the IoT device is convex and there are no impacts from other IoT devices. Therefore, each IoT device will find its optimal scheduling (best response) and will not change its strategy at every iteration. At the power retailer agent side, the best response

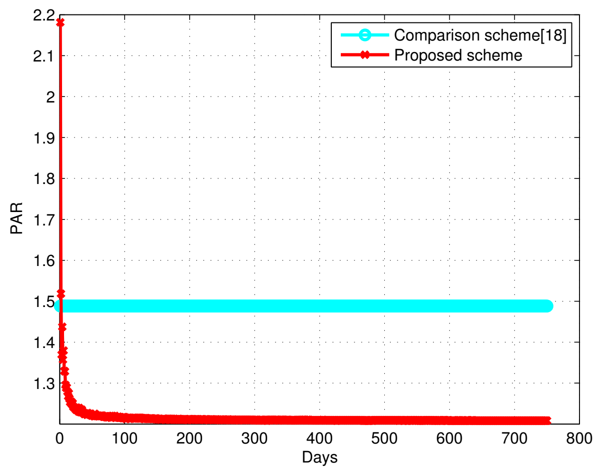


Fig. 3: PAR vs. days.

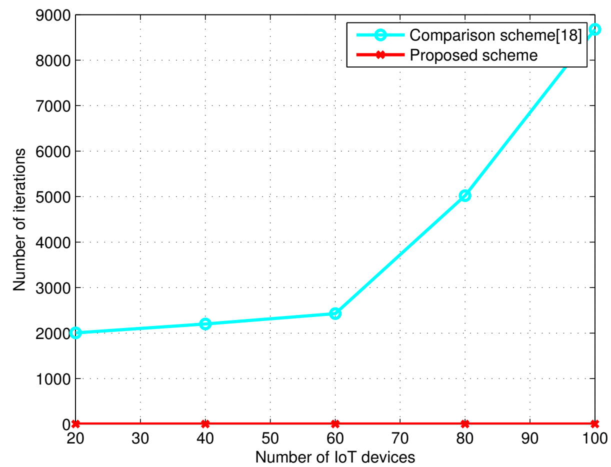


Fig. 4: Iteration vs. number of IoT devices.

of power retailer agent will be obtained from (14) because the equation is convex and there is only one player on the supplier side. For the proposed scheme, the calculation of the SE is at first gradual but quickly stabilizes and converges as follows. At the beginning of the first day, each IoT device k finds $\mathcal{S}_k^*$ by (17) for a given $\mathcal{P}$. Then, $\overline{E_{\mathcal{A}}^t}$ is updated according to the IoT device scheduling. Based on $\overline{E_{\mathcal{A}}^t}$, the power retailer agent finds $\mathcal{P}^*$ by solving (14). The next day, we calculate the objective function on the follower side again with the mean value of the best strategies of the power retailer agent we obtained the day before. As a result, the proposed algorithm is a series of processes of obtaining the SE. The conventional Stackelberg game model repeats the proposed algorithm until it obtains the SE for each timeslot. However, in a massive system with no major changes, it is possible to derive a near-optimal strategy even after one execution per timeslot, as in the proposed algorithm. Furthermore, given the statistics of the previous RTP data, this early processing can be skipped.

IV. Performance Evaluation

TABLE I: Simulation Parameters

Parameter	Value
Timeslot units	1 hour
$E_k^{on}, \forall k$ (uniform distribution)	From 1 to 100W
$E_k^{Max}, \forall k$ (uniform distribution)	From 50 to 100W
$E_k^{Min}, \forall k$ (uniform distribution)	0W
mean of $C_k^{begin}, \forall k$ (Poisson distribution)	From 1 to 24
$\rho_k, \forall k$ (uniform distribution)	From 1 to 10

In this section, we evaluate the performance of the proposed game in terms of the communications overhead and delay. For this evaluation, an event-driven simulator for the proposed game was implemented in the C programming language. Moreover, the optimization problem was computed using the MOSEK optimization tool [17].

The simulation parameters are illustrated in Table I. To demonstrate the performance of the proposed scheme, small-scale topologies (K=100), and large-scale topologies (K=10,000) are configured. In the small-scale topologies, the existing scheme presented in [18] is compared with the proposed scheme. This is because that comparison scheme cannot be simulated in large-scale topologies since its computation time is exponential. In the large-scale topologies, we verify that the proposed scheme can prevent system blackout by adjusting the electricity demand so that the peak electricity consumption remains low in the massive environment with more than 10,000 devices.

A. *Small Topologies (K = 100)*

This section compares the peak load reduction and communication overhead of the proposed scheme with the comparison scheme [18]. For comparison of peak load reduction, we analyze the peak-to-average ratio (PAR) of the power scheduling. In order to compare the communication overhead, we also analyze the number of iterations of strategy exchanges, which increases according to the number of IoT devices. Then, we simulate the impact of increasing the number of iterations on communication power consumption and delay.

1) Peak Load Reduction: Fig. 3 shows the PAR according to days. As shown in the figure, the PAR of the proposed scheme decreases with increasing number of days since the strategy converges to near-optimal (i.e., SE). The initial PAR of the proposed scheme is 2.2. However, the PAR of proposed scheme rapidly decreases in early days and then stabilizes steadily at 1.2. This shows that the proposed scheme works well over the earliest days. The problem of a very high PAR at the initial days (i.e., cold start problem) can be solved by the early accumulation of DB. On the other hand, the PAR of the comparison scheme is stably maintained at 1.4. The reason the comparison scheme maintains a stable PAR from the beginning is it uses the SE state strategy starting from the first day. The Fig. 3 shows that the convergence PAR reduction

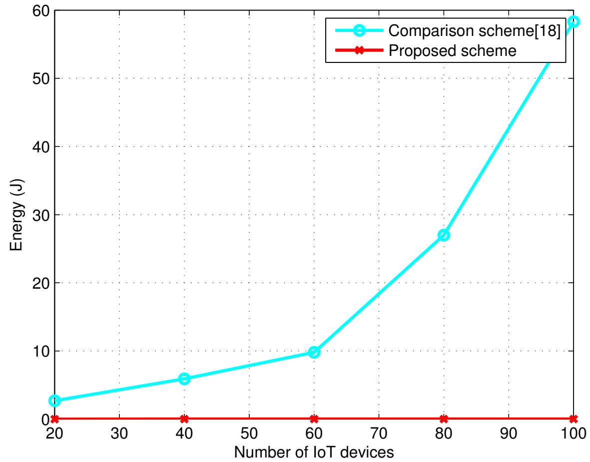


Fig. 5: Communication energy consumption vs. number of IoT devices.

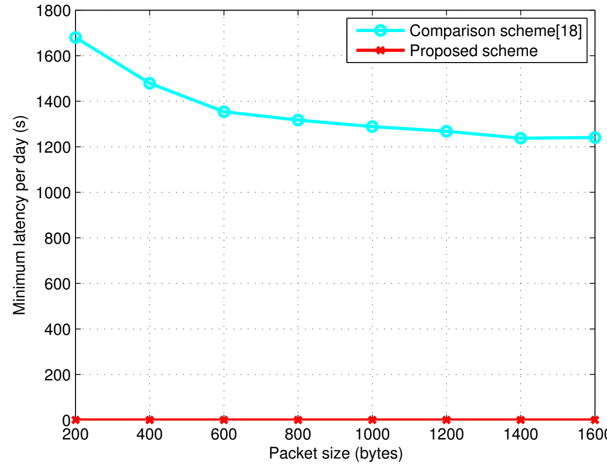


Fig. 6: Minimum latency per day vs. packet size (K = 100, Data size = 1,600 bytes).

of proposed scheme and comparison scheme are 45% and 35%, respectively.

2) Iteration of Strategy Exchanges: In order to analyze the communication overheads, we first need to analyze the characteristics of the iterative in strategy exchange. It has been experimentally demonstrated in [19] that the number of iterations increases as the number of followers increases. We also verified through several simulations that the number of iterations increases only moderately in the comparison scheme as the number of IoT devices increases as shown in Fig. 4. However, the number of iterations in the proposed scheme is always 1 regardless of the number of IoT devices since the proposed scheme exchanges strategies only once a day. Since IoT devices and the power retailer need to communicate for strategy exchange, the number of iterations is equivalent to the number of communications. Therefore, in the proposed scheme, the number of communications for strategy exchange is much smaller than that of the comparison scheme.

3) Communication Overhead: In this simulation, we assume that Wi-Fi wireless technology is used for comparative analysis of communication overheads. For convenience of calculation, the data size of the exchanged strategy is 1,600 bytes. Furthermore, due to practical limitations, it is difficult to implement wireless communication simulations using 10,000 IoT devices. Thus, energy consumption is calculated by applying energy per bit [20] in the wireless communication process.

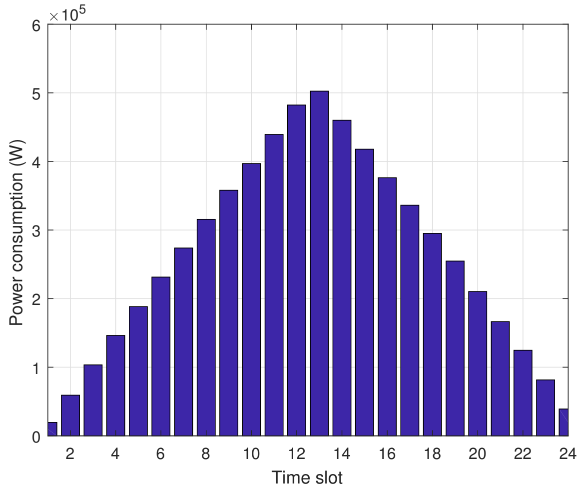
In one strategy exchange communication, energy consumption occurs as much as the strategic data size and increases in proportion to the number of iterations. Therefore, based on Fig. 4, the communication energy consumption is calculated, as shown in Fig. 5. The communication energy consumption of the proposed scheme increases just a little, but linearly since only the amount of total communication data increases as the number of IoT devices increase. On the other hand, the communication energy consumption in the comparison scheme exponentially increases with the number of IoT devices since the number of iterations increases as well as the amount of total communication data. As a result, the proposed scheme can reduce the communication energy consumption by more than 99% compared with the comparison scheme.

Latency is caused by communication in each strategy exchange. The latency is accumulated by the iterative communication, which is a critical problem for smart manufacturing systems. For comparative analysis of communication latency between the proposed scheme and the comparison scheme, we refer to the analysis data of [21]. According to [21], the packet size determines how many strategy data are fragmented in wireless communication and has a significant impact on the amount of the latency. Considering the number of communications in the comparison scheme as shown in Fig. 4, the minimum latency per day for each packet size is shown in Fig. 6. In the comparison scheme, the total latency is high since the latency is accumulated as the number of communication increases. On the other hand, the proposed scheme reduces the latency more than 99% compared with the the comparison scheme by minimizing the latency with one-time communication. As a result, the proposed scheme can dramatically reduce the communication latency compared to the existing scheme, and can operate an efficient smart manufacturing systems.

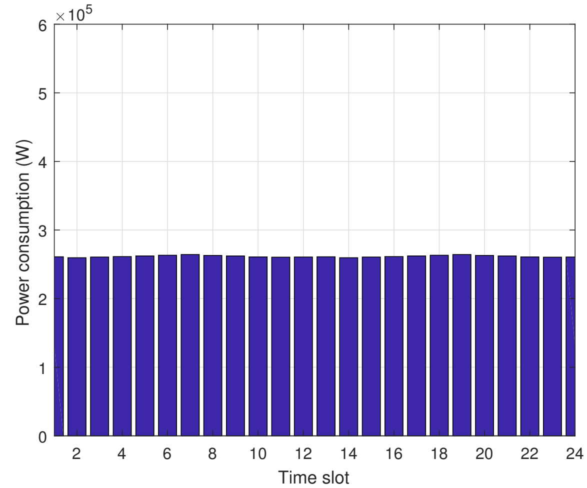
B. Large Topologies (K = 10,000)

In this section, we demonstrate a performance of the proposed scheme in the large-scale topologies. First, we show the cold start problem due to the lack of power consumption pattern data through simulation at the early days. After a sufficient amount of pattern data has accumulated, the proposed scheme stably reduces the peak load well in the large-scale environment by converging to the near-optimal strategy.

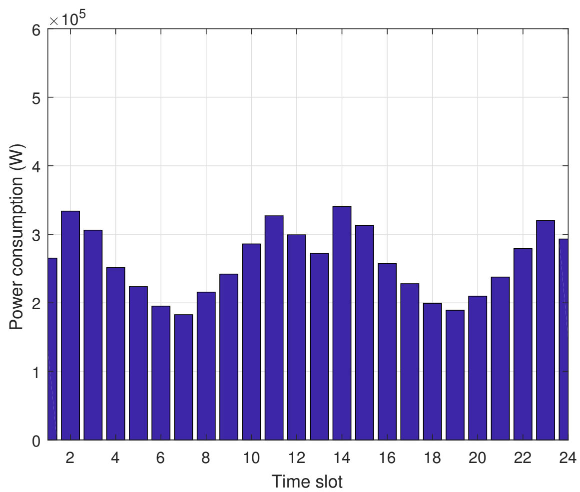
Fig. 7 shows the cold start problem that occurs in the early days of the proposed scheme when power consumption pattern data are insufficient. As shown in Fig. 7 (a), the original demand of power consumption is concentrated at 13:00h. The IoT devices shift their operation time for minimizing the cost function. Therefore, the average power consumption pattern



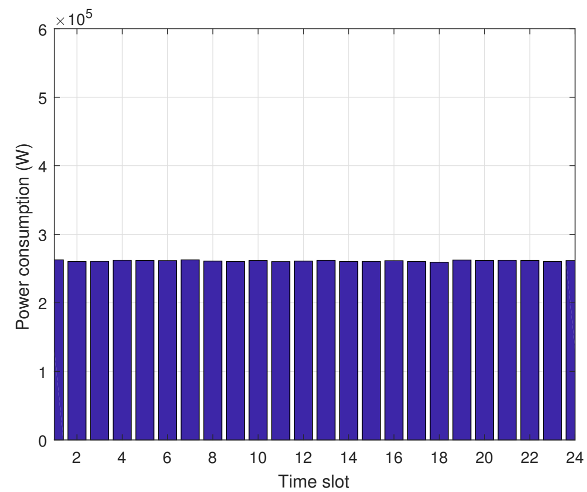
(a) Original demand



(b) 1st day



(c) 2nd day



(d) 3rd day

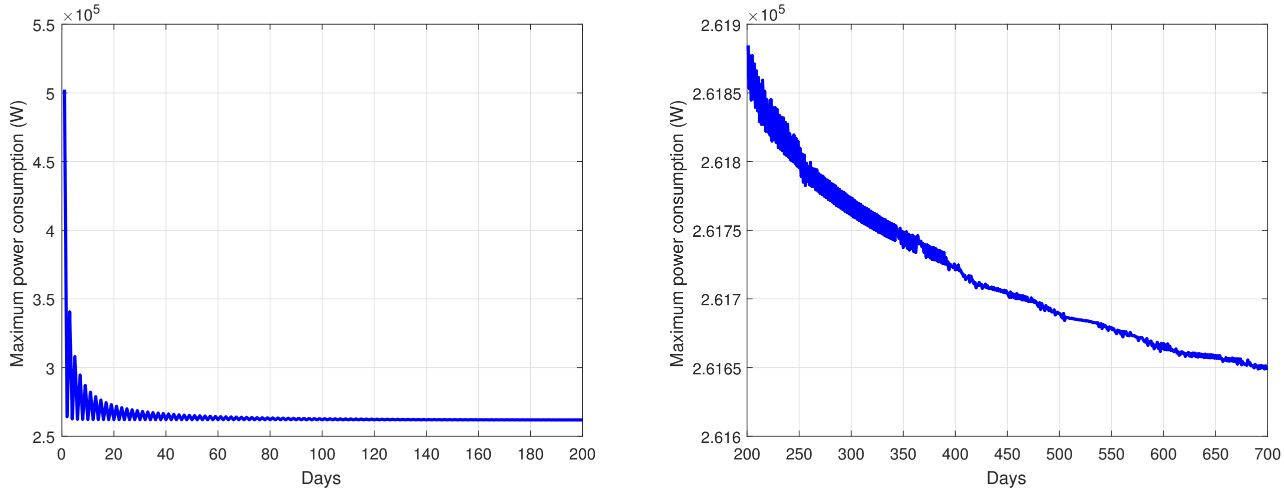
Fig. 7: Power consumption vs. timeslot for days 1-3.

with considering original demand and the first day is figured almost flat as shown in Fig. 7 (b) by algorithm 1. On the second day, the power consumption is concentrated at 11:00 and 14:00 since the degrees of user dissatisfaction are the same and electricity costs are inexpensive within the preference time. Despite the degree of user dissatisfaction being high, the power consumption is also concentrated at 2:00 and 23:00 since the cost function is minimized at those times due to low electricity costs. Therefore, the average power consumption pattern is figured as shown in Fig. 7 (c). With this iterative process, the power consumption pattern converges to SE after early days.

Fig. 8 shows the maximum power consumption per day and a process of solving the cold start problem. As shown in Fig. 8 (a), the cold start problem occurring during the early stage is solved by converging very quickly to the stable section with the accumulation of power consumption pattern data. Furthermore, as shown in Fig. 8 (b), the deviation of the maximum power consumption is gradually reduced in the stable section and the power consumption pattern converges to a near-optimal strategy. Therefore, it is possible to verify that the proposed scheme significantly reduces the peak load in large-scale topologies as well as small-scale topologies.

V. Conclusions

In this paper, we considered the power consumption of massive smart manufacturing systems consisting of more than 10,000 IoT devices. Since the power consumption of IoT devices does not fluctuate, we proposed a light-weight Stackelberg game theoretic DR scheme for massive smart manufacturing systems. This scheme is proposed to reduce the peak load and communication overheads without iterative processes to calculate the SE. For the Stackelberg game, the optimization formulation and corresponding model were proposed for the power retailer agent and IoT devices. To maximize the benefit of the power retailer agent, we proposed the day-ahead RTP algorithm, which is based on historical power consumption statistics. To maximize the benefit of the IoT devices, we proposed a scheduling optimization problem that considers the electricity bill and the degree of user convenience. Via



(a) Maximum power consumption vs. days for 1-200.

(b) Maximum power consumption vs. days for 200-700.

Fig. 8: Maximum power consumption vs. days for 1-700.

simulations, we demonstrated that the power consumption can be distributed over the day using the proposed scheme. We also verified that the proposed scheme reduces the communication overheads 99% more than the conventional Stakelberg game based scheme. By comparison with the conventional scheme, we show that the proposed algorithm can sufficiently reduce the communication overhead and delay. In addition, the proposed scheme can reduce the electricity peaks in massive smart manufacturing systems using the historical RTP. Future research will focus on the cold start problem of RTP based on appliance usage patterns using machine learning algorithms while considering environmental issues and factors.

ACKNOWLEDGEMENT

This research was supported by Korea Electric Power Corporation(Grant number:R15XA03-69) and Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science and ICT (NRF-2017R1A4A1015675).

REFERENCES

[1] M. M. Umble, "Analyzing Manufacturing Problems using V-A-T Analys" *Production and Inventory Management Journal,* Second Quarter 1992, pp. 55.

[2] F. Tao, and Q. Qi, "New IT Driven Service-Oriented Smart Manufacturing: Framework and Characteristics," to appear in *IEEE Transactions on System, Man, and Cybernetics: Systems,* 2017.

[3] Smart Manufacturing Leadership Coalition (SMLC), "Implementing 21st Century Smart Manufacturing," *Workshop Summary Report,* June 2011.

[4] J. Davis, T. Edgar, J. Porter, J. Bernaden, and M. Saril, "Smart Manufacturing, Manufacturing Intelligence and Demand-Dynamic Performance," *Computers & Chemical Engineering,* Vol. 47, No. 12, December 2012, pp. 145–156.

[5] R. Deng, M. Chow, and J. Chen, "A Survey on Demand Response in Smart Grids: Mathematical Models and Approaches," *IEEE Transactions on Industrial Informatics,* Vol. 11, No. 3, March 2017, pp. 570–582.

[6] E. Karimi, and M. Kazerani, "Impact of demand response management on improving social welfare of remote communities through integrating renewable energy resources," *in Proc. of IEEE conf. on Electrical and Computer Engineering,* April 2017.

[7] M. Starke, D. Letto, N. Alkadi, R. George, B. Johnson, K. Dowling, and S. Khan, "Demand-side response from industrial loads," *in Proc. of Clean Technol. Conf.,* January 2013, pp. 46–49.

[8] L. Siebert, L. Ferreira, E. Yamakawa, E. Custodio, A. Aoki, T. Fernandes, and K Cardoso, "Centralized and Decentralized Approaches to Demand Response Using Smart Plugs," *in Proc. of IEEE Conf. PES T&D Conference and Exposition,* April 2014, pp. 1–7.

[9] H. von Stackelberg, "Market Structure and Equilibrium," *Springer,* Berlin, Heidelberg, 2011.

[10] R. Dylewski, A. Jardzioch, and I. Krebs, "The Optimal Sequence of Production Orders, Taking into Account the Cost of Delays," *Management and Production Engineering Review,* Vol. 7, No. 2, June 2016, pp. 21–28.

[11] E. Hartweg, "Smart Factories–Self-Organizing Production Unit," *Production and Operations Management Society Conference,*

[12] Q. Zhou, W. Guan, and W. Sum, "Impact of Demand Response Contracts on Load Forecasting in a Smart Grid Environment," *2012 IEEE Power and Energy Society General Meeting,* San Diego, CA, 2012, pp. 1–4.

[13] H. R. Khosravani, M. D. M. Castilla, M. Berenguel, A. E. Ruano, and P. M. Ferreira, "A Comparison of Energy Consumption Prediction Models Based on Neural Networks of a Bioclimatic Building," *Energies,* Vol. 9, No 12. p. 57, January 2016.

[14] R. J. Hyndman, "Moving Averages. In: M. Lovric (eds) International Encyclopedia of Statistical Science," *Springer,* Berlin, Heidelberg, 2011, pp. 866–869.

[15] R. Zhou, Y. Pan, Z. Huang, and Q. Wang, "Building Energy Use Prediction Using Time Series Analysis," *2013 IEEE 6th International Conference on Service-Oriented Computing and Applications,* Koloa, HI, 2013, pp. 309–313.

[16] L. Park, Y. Jang, S. Cho, and J. Kim, "Residential Demand Response for Renewable Energy Resources in Smart Grid Systems," to appear in *IEEE Transactions on Industrial Informatics,* 2017.

[17] http://www.mosek.com/

[18] M. Latifi, A. Khalili, A. Rastegarnia, and S. Sanei, "Fully Distributed Demand Rsponse Using the Adaptive Diffusion-Stackelberg Algorithm," *IEEE Transactions on Industrial Informatics,* May 2017, pp. 2291–2031.

[19] M. Yu, "Demand Response in Smart Grid: A Stackelberg Game-Theoretic Approach," Ph.D. dissertation, Dept. Electron. Eng, Hanyang Univ. Seoul, Korea, 2015.

[20] P. Smith, "Comparing Low-Power Wireless Technologies," *Digi-Key Electronics,* White Paper, August 2011.

[21] J. Horalek, T. Svoboda, and F. Holik, "Analysis of the Wireless Communication Latency and Its Dependency on a Data Size," *in Proc. of IEEE Computational Intelligence and Informatics (CINTI),* Nov. 2016, pp. 145–150.

Chunghyun Lee received B.S. degree in Computer Science and Engineering from Chung-Ang University, Seoul, South Korea, in 2018. He is currently pursuing his M.S. degree in Computer Science and Engineering at Chung-Ang University, Seoul, Korea. His research interests include demand response, game theory, micro grids, and smart grids.

Laihyuk Park received B.S., M.S., Ph. D. degrees in Computer Science and Engineering from Chung-Ang University, Seoul, South Korea, in 2008, 2010, and 2017, respectively. He is currently an assistant professor with the Da Vinci College of General Education, Chung-Ang University, Seoul, South Korea. His research interests include demand response, smart grid, electric vehicles, Internet of things, and wireless network.

Sungrae Cho received his Ph.D. degree in electrical and computer engineering from Georgia Institute of Technology, Atlanta, Georgia, USA, in 2002, and his B.S. and M.S. degrees in electronics engineering from Korea University, Seoul, Korea, in 1992 and 1994, respectively. He is currently a full professor with the School of Computer Science and Engineering, Chung-Ang University, Seoul, Korea. Prior to joining Chung-Ang University, he was an assistant professor with the Department of Computer Sciences, Georgia Southern University, Statesboro, Georgia, USA, from 2003 to 2006, and a Senior Member of Technical Staff with Samsung Advanced Institute of Technology (SAIT), Kiheung, Korea, in 2003. From 1994 to 1996, he was a Member of Research Staff with the Electronics and Telecommunications Research Institute (ETRI), Daejeon, Korea. From 2012 to 2013, he held a Visiting Professorship with the National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, USA. He is an Editor of the Elsevier Ad Hoc Networks Journal since 2012 and has served numerous international conferences as an organizing committee member, such as IEEE SECON, ICOIN, ICTC, ICUFN, TridentCom, and IEEE MASS. His research interests include wireless networking, ubiquitous computing, performance evaluation, and queuing theory.