

스마트 그리드 환경에서 수요 측면의 에너지 관리

(Demand-Side Energy Management on Smart Grid)

정재윤*, 리진*, 홍원기***

* 포항공과대학교 컴퓨터공학과 분산처리 및 네트워크관리 연구실

** 포항공과대학교 정보전자융합공학부 분산처리 및 네트워크관리 연구실

{dejavu94, gunine, jwkhong}@postech.ac.kr

요 약

에너지 소비를 줄이고 낭비를 막기 위하여 가정에서의 효과적인 에너지 관리는 스마트 그리드 환경에서 필수적인 요소이다. 우리는 이 논문에서 가정 내 에너지 관리를 deadline이 있는 작업 스케줄링 문제로 모델링 하고 이 문제를 near-optimal로 풀 수 있음을 보인다. 또한 제안하는 방법을 사용하여 가정의 에너지 관리 시스템을 구현하였으며 시뮬레이션을 통해 제안하는 에너지 스케줄링 방법의 성능을 확인한다. 우리가 본 연구에서 중점적으로 다루는 내용은 수요자 측면의 에너지 관리이며, 스마트 그리드 환경에서 demand-response 프로그램과 가정용 배터리를 사용함으로써 효과적으로 에너지 수요를 분산시키는 것이다.

Keywords: 스마트 그리드, demand-response, 에너지 관리

1. 서론

오늘날 많은 국가에서 에너지 수요, 특히 전력에 대한 수요가 급속도로 증가하고 있다. 이에 따라 전력 가격은 수년간 급격하게 증가하였으며, 전력 최대 수요 시간(peak time) 대에 발생하는 전력 부족과 정전에 대한 예방과 대처가 더욱 중요해 지고 있다. 에너지 절약에 대한 노력으로 가정에서의 효율적인 에너지 사용은 미래의 스마트 그리드 환경에서 더욱 중요하다. 스마트 그리드 환경에서 우리는 다음 세 가지 특성에 주목하였다.

- **보다 나은 에너지 관리:** 현재 에너지 관리 서비스는 사용자에게 아주 단순한 정보만을 주고 있기 때문에 에너지 사용 현황에 대해 쉽게 파악하기 어렵다. 상세한 에너지 사용 정보 제공은 스마트 그리드 환경에서 필수적이다.
- **Demand-response(DR) 프로그램:** 수요 측면의 에너지 관리는 demand-response 프로그램에 사용자가 대응하는 것으로 global level에서의 효율성도 고려해야 한다.
- **가정용 고용량 배터리:** 가정용 고용량 배터리는 가까운 미래에 상용화 될 것으로 보이며 electric vehicles (EV)의 배터리 역시 가정용 배터리로 사용되어 에너지를 충전 및 소비 할 수 있을 것이다.

DR 프로그램은 1999년도부터 미국에서 이상 고온 발생시에 전력 발전량 부족을 극복하기 위해 특별히 높은 전력 가격 정책을 사용한 것을 그 시작으로 보고 있다. 실제 DR 프로그램은 2001년 여름 동안 미국의 전력 시스템의 긴급상황 발생 완화에 큰 역할을 하였다[1]. DR 프로그램의 주요 장점 중 한가지는 전력 관리에서 가장 중요하게 여겨지는 인자 중 하나인 최대 전력 수요를 분산 시킬 수 있다는 점이다. 높은 전력 수요를 충당하기 위하여 전력 공급자는 화석 연료 등을 사용한 고비용 긴급 발전을 하기도 한다.

DR 프로그램은 전력 공급자로 하여금 소비자의 최대 전력 수요를 분산 시킬 수 있게 하는 역할을 한다. DR 프로그램의 가격 정책은 크게 두 가지 방법으로 구분 될 수 있다. 첫째는 미리 지정한 에너지 소비량을 넘길 경우 높은 가격을 부과하는 것이고, 둘째는 시장 가격을 기반으로 에너지 가격을 결정하는 것이다. 이 두 가지 가격 정책은 모두 가격을 사용하여 소비자를 압박함으로써 특정 시간대에 전력 사용량이 물리는 것을 방지한다. 따라서 스마트 그리드 환경에서 DR 프로그램이 시행된다면 전력 소비자 역시 실시간 혹은 특정 시간 단위로 변동하는 전력 가격을 고려하여 전력을 소비하는 것이 중요하게 된다.

실제로 기존 연구 중 소비자가 시장 기반의 전력 가격 변동에 노출되어 있을 때, 소비자들의 전력 수요가 변화하면서 공급 가격과 수요량이 균형점을 찾아 간다는 것을 보여주고 있다[2][3]. 하지만 DR 프로그램이 이론적으로 여러 이익이 있다는 것이 보이지만, 미국의 뉴욕에서 조사 한 결과에 따르면 54%의 가정에서만 DR 프로그램에 대응하여 에너지 소비 행태를 조절 할 것이라는 응답이 나타났다.[4]. 다음은 가정에서 적극적으로 DR 프로그램에 대응하기 어려운 이유이다.

- **정보 분석 능력 부족:** 에너지 소비자는 일반적으로 에너지 관리에 기술적인 지식이 없으며, 이는 단순히 에너지 소비량이나 가격 정보를 제공하는 것 만으로는 소비자들이 스스로 에너지 사용량을 효과적으로 분산 시킬 수 없음을 의미한다.
- **에너지 사용 계획 능력 부족:** 같은 맥락에서 소비자들은 자신의 에너지 수요를 최적으로 스케줄링 하는 것에 어려움을 느낀다.
- **안정성 추구:** 대부분의 소비자들은 안정적인 에너지 공급을 무엇보다 중요하게 생각하기 때문에 높은 가격의 에너지 사용을 그다지 꺼려하지 않는다. 실제로 많은 가정에서 시장 가격 기반의 DR 프로그램 보다 조금 더 비싸지만 안정적이고 고정된 가격 정책을 선호한다.

결과적으로 우리는 효과적으로 가정의 에너지 관리를 할 수 있는 알고리즘과 시스템이 필요하다. 이 논문에서는 Green Home Service (GHS)를 디자인 하고 구현하였음을 보이고, DR 프로그램이 시행될 때 자동으로 전력 사용을 스케줄링 할 수 있는 방법을 제시한다. 그림 1은 GHS가 스마트 그리드 테스트 환경에 배치될 모습을 보여주고 있다. 스마트 그리드 환경에서 GHS는 가정용 에너지 관리 시스템의 외부 인터페이스가 된다. 가정에서 사용한 에너지 소비량을 공급자에게 전송하거나 DR 가격과 전력 제어 요청을 받을 수 있다. 이때 GHS의 서버는 가정 내 가전제품들과 통신하며 가전 제품별 상세한 에너지 사용 정보를 수집하고 원격 제어를 가능하게 한다. GHS의 design feature로서: 1) GHS는 사용자의 에너지 소비 작업을 자동으로 스케줄링 할 수 있는 알고리즘을 포함할 수 있다. 우리가 제안하는 알고리즘은 사용자의 만족도를 가능한 보장하면서 전체적인 최대 전력 수요를 낮추는 방법이다. 또한 배터리 사용을 통해 전력 사용을 분산시키는 효과를 더욱 크게 한다. 2) GHS는 가전제품의 전력 사용을 모니터링 하고 제어 할 수 있도록 하며 다양한 종류의 인터페이스와 통신 프로토콜을 지원 할 수 있도록 디자인 한다. 이는 Aspect-Oriented Programming (AOP)를 사용 함으로써 각 프로토콜의 구현 및 접목을 용이하게 하며 제안하는 에너지 관리 방법의 효용성을 증명하기 위하여 시뮬레이션을 통한 검증을 하였다.

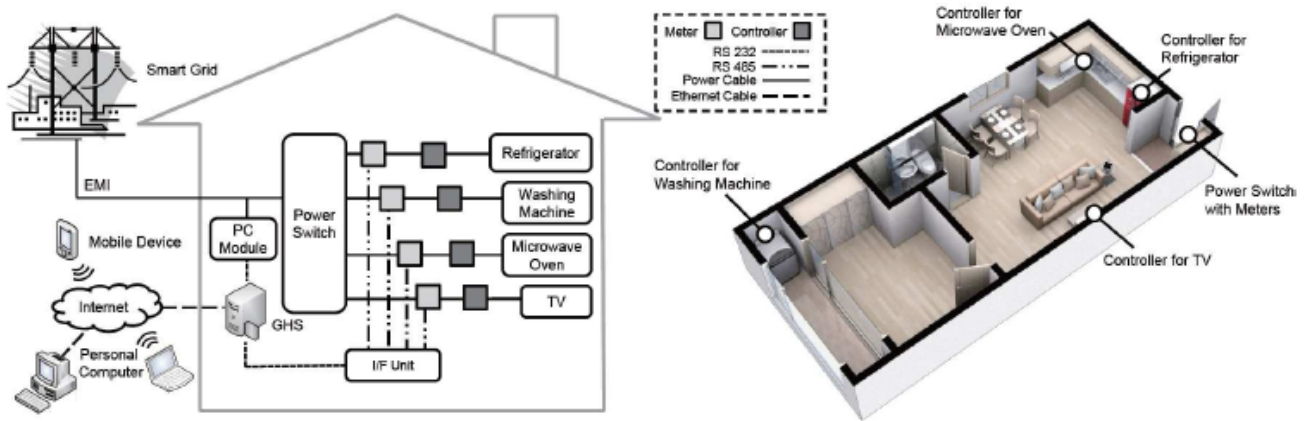


그림 1. 스마트 그리드 환경의 Green Home Service

2. 관련 연구

Optimization technique를 에너지 관리에 적용한 연구로서 [5]는 마이크로 그리드에서 공급자 측면의 최적 전력 생산에 초점을 두었다. 마이크로 그리드에서 발생하는 태양이나 바람 에너지 같은 재생 가능한 에너지를 사용함으로써 얻는 이점이 있지만 전력 품질에 대한 제어가 필요하다는 이슈가 있었다. [6][7]은 가정의 에너지 효율을 최적화 하기 위하여 난방 장치를 최적으로 조절하는 부분에 중점을 두고 있다. 에너지 소비를 최소화 하면서 실내 온도를 일정하게 유지하는 것이 목표이며 사용자의 선호 난방 시간이 일부 고려되기도 하였다. [8]은 에너지 수요-공급 시장에서 경매 입찰 방식을 기반으로 한 에너지 가격 최적화에 대한 연구결과를 제시하였다. 에너지 수요를 다른 시간으로 이동시킴으로써 사용자 관점의

에너지 가격의 최적화에 중점을 두고 있지만 에너지 사용시간의 이동으로 인해 사용자가 불편을 감수해야 하는 한계를 갖고 있다. 에너지 스케줄링에 관한 연구 [9]는 인센티브 기반의 에너지 관리 방법을 제시하고 있지만, 역시 **deadline**이 없는 유연한 작업 스케줄링 문제를 모델로 하고 있어 실제 생활 환경과는 동떨어진다.

우리는 본 연구에서 소비자 관점의 에너지 스케줄링을 **deadline** 기반의 **atomic** 작업을 시간 축에 배치 하는 것으로 여긴다. 때문에 우리 모델에서 축구 경기를 시청하기 위한 TV의 사용은 다른 시간으로 옮겨지지 못하기도 한다. 또한, 하나의 에너지 수요가 사용되기 시작하면 사용 도중에 멈출 수 없다. 이는 실생활에 보다 가까운 모델을 제시하였음을 의미한다. 우리의 접근 방법과 유사한 연구가 [10]에서 제시되었다. 에너지 스케줄링 문제를 전통적인 스케줄링 문제인 Longest Process Time (LPT)와 Just-In Time (JIT)로 여겼으며 이는 NP-hard 문제로 알려져 있다[11]. 때문에 근사 알고리즘을 사용한 문제 해결 방법이 제시 되었으나 에너지 수요의 길이나 DR에 의한 에너지 가격 변화는 고려되지 않았다.

스마트 그리드의 에너지 관리와 제어는 Automatic Metering Infrastructure (AMI)를 기반으로 하여 정확하고 효율적으로 에너지 사용을 모니터링 함으로써 에너지 사용 현황을 보다 자세하게 파악하려는 노력이 계속되고 있다. 공급자 측면에서 Automatic Meter Reading (AMR)[12]가 사용자의 에너지 사용을 효과적으로 모니터링 및 검침 하는 방법으로 제시되었으며, 사용자 역시 자신의 에너지 사용 상태에 대한 정보를 상세하게 제공 받게 함을 목표로 하고 있다. 하지만 이러한 활동들은 생산자 위주로 이루어져야만 하는 한계가 있다. 본 연구에서 제안하는 GHS는 소비자 수준에서 에너지 사용 정보를 수집하고 관리 하는 것에 중점을 두고 있기 때문에 스마트 그리드 환경에서 소비자와 표준화된 Energy Management Interface (EMI) 사이의 격차를 seamless하게 채워 줄 수 있다. 본 논문에서는 DR에 의한 real-time pricing 환경의 수요 분산 알고리즘을 제시하고 검증하였으며, 향상된 검침 환경과 전력 사용 원격 제어 환경을 구현하였다.

3. 최적의 수요 측면의 에너지 관리

3.1 사용자 측면의 에너지 관리 모델

본 연구에서는 한 가정 내에서의 에너지 사용 스케줄링에 중점을 두고 있다. 가정에서 하루 동안의 에너지 수요는 특정 ‘시간 내에 수행되어야 하는 작업’들을 리스트로 나타낼 수 있다. 이 작업들 중 일부는 고정적(e.g. 생방송 TV 시청)이거나 하루 종일 균일하게 사용되는 가전제품(e.g. 냉장고)이다. 하지만 나머지 작업은 사용자가 원하는 시간 내에서만 수행되면 되는 상대적으로 유연한 작업들이다(e.g. 세탁기). 또한 작업들은 세탁기나 오븐처럼 한번 시작되면 사용 도중에 멈출 수 없어야 한다.

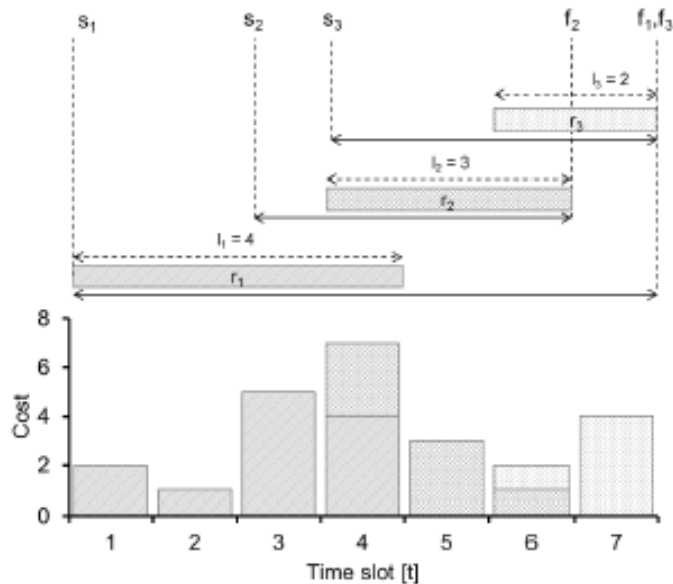


그림 2. 수요 측면의 에너지 스케줄링 문제

그림 2는 본 논문의 에너지 스케줄링 문제를 간략하게 보여주고 있다. 각 작업(에너지 수요)은 에너지 사용 요구 d 로 나타내며 시간당 에너지 소비량 r , 시작 시간 s 와 끝나는 시간(deadline) f 를 갖고 있다. 이때, 길이 l 의 작업은 s 와 f 사이에서 할당 되어야만 한다. 에너지 가격은 시장 가격을 기반으로 하는 DR이

사용된다고 가정하였기 때문에 각 시간마다 변하게 된다. 각 작업의 에너지 소비는 시간에 따라 변하지 않고 고정되어 있다고 가정하였다.

제시하는 에너지 스케줄링 문제는 곧 다음과 같이 수식화 된다.

주어진 소비자의 에너지 수요 $d_j = (s_j, f_j, r_j, l_j) \in D$ where $1 \leq j \leq m$ 에 대하여, 주어진 시간 범위 내에 작업을 할당함으로써 다음의 수식을 최적화 하는 것이다.

$$\min\left\{\max\left\{\sum_{d_j \rightarrow t} P(r_j)\right\}\right\}$$

이때 $d_j \rightarrow t$ 는

$$s_j \leq t \text{ and } f_j \geq t + l_j - 1$$

의 조건을 지켜야 한다.

위의 모델의 제약조건은 다음과 같다. 1) 스케줄링 전과 후의 에너지 총량은 같아야 한다. 2) 하나의 작업은 여러 개의 작업으로 나뉘어서 수행될 수 없다. 3) 스케줄링은 24시간 단위로 수행된다.

우리의 작업 스케줄링 문제에서 작업 d_j 는 항상 시간 슬롯 $\{s_j, f_j\}$ 안에 할당되어야 하며, 시간당 r_j 만큼의 전력을 소비한다. 이때, 작업을 시간 $(d_j \rightarrow t)$ 에 할당한다는 것은 d_j 를 연속적인 시간 슬롯 $\{s_j, f_j\}$ 안에 $t + l_j - 1$ 의 작업을 할당함을 의미한다. 제안하는 스케줄링 문제의 목표는 전체 에너지 사용 비용을 최소화 하는 것이 아니라 시간 슬롯에 할당된 누적 작업 중 최대값을 최소로 하는 것이다. 이는 우리의 작업 스케줄링이 최대 전력 수요를 낮춰주는데 목표를 두고 있음을 뜻하며, 시간에 따른 에너지 수요를 균일하게 유지할 수 있도록 도와준다. 전력 공급자는 전력 가격 $P()$ 를 변화시켜 사용자를 압박함으로써 전력 수요 분산을 유도할 수 있다.

Algorithm 1 minMax Scheduling

Input:

n number of time slots, m number of demands

Time slot $t \in T$ where $1 \leq t \leq n$

Consumer demand $d_j = (s_j, f_j, r_j, l_j) \in D$ where $1 \leq j \leq m$

$s \in T$ is start time, $f \in T$ is finish time, r is consuming rate, l is number of time slots of demand.

Cost for demand rate r at time slot t $P_t(r)$

Output:

$scheduleMatrix[n][m]$ such that $scheduleMatrix[i][j]$ is the energy cost of j -th demand at time slot i

```

1: Initialize  $scheduleMatrix[n][m]$  to zeros
2: Initialize  $c_{1..n}$  to zeros
3: Assign tasks in  $D$  with fixed schedules (i.e.  $f - t + 1 = l$ ) to time slots, and adjust  $c$  accordingly
4: Sort  $D$  based on  $r$  as the major key in descending order
5: Sort  $D$  based on  $l$  as the minor key in descending order
6: for  $i = 1$  to  $m$  do
7:    $minMax \leftarrow \infty$ 
8:    $schedulingSlot \leftarrow 0$ 
9:   for  $t = s_i$  to  $f_i - l_i + 1$  do
10:     $minMaxAtT = \text{maximum}(c_t + P_t(r_i), c_{t+1} + P_{t+1}(r_i), \dots, c_{t+l-1} + P_{t+l-1}(r_i))$ 
11:    if  $minMaxAtT < minMax$  then
12:       $minMax \leftarrow minMaxAtT$ 
13:       $schedulingSlot \leftarrow t$ 
14:    end if
15:  end for
16:  for  $j = schedulingSlot$  to  $l - 1$  do
17:     $scheduleMatrix[i][j] \leftarrow P_j(r_i)$ 
18:  end for
19: end for

```

3.2 minMax 스케줄링 알고리즘[14]

$minMax$ 스케줄링 알고리즘(Algorithm 1)은 LPT greedy search heuristic을 기본으로 하고 있지만 두 가지

다른 점이 있다. 첫째, LPT에서는 가장 수행시간이 긴 작업부터 할당하지만 minMax 알고리즘에서는 에너지 사용이 가장 큰 작업부터 할당한다. 둘째, 작업을 할당 할 때에는 에너지 소비량이 아닌 에너지 가격을 기반으로 할당을 한다.

알고리즘은 먼저 가장 에너지 소비가 많은 작업부터 할당하는데 이 때 동일한 에너지를 소비하는 작업이 2개 이상이라면 그 중 가장 많은 시간이 소요되는 작업을 선택하여 할당한다. 작업을 할당할 때에는 할당 가능한 시간 슬롯들 중(시작 시간과 deadline이 있으므로) 시간 슬롯의 누적 가격을 최소로 하는 곳에 작업을 할당한다. minMax 알고리즘의 시간복잡도는 $O(mn)$ 이다.

우리는 [14]에서 minMax 알고리즘이 NP-hard 문제를 근사적으로 풀 수 있음을 증명하였다. 이 증명 과정에서 minMax 알고리즘은 deadline이 없는 에너지 스케줄링 문제를 near-optimal solution로 풀고 있음을 보였으며 deadline이 있을 경우에도 근사 정확도가 조금 낮아질 뿐 비슷한 결과를 얻을 수 있었다. 알고리즘의 근사 정확도는 작업이 같은 시간 슬롯에 얼마나 많이 할당 될 수밖에 없는지에 관련된다. 즉 타이트한 deadline의 작업이 많은 경우 스케줄링의 여지가 적기 때문에 그 결과가 optimal에서 멀어지게 된다.

3.3 BatMax 스케줄링 알고리즘

minMax 알고리즘의 결과는 가정용 고용량 배터리의 사용을 통해 에너지 수요를 보다 적극적으로 분산시키기 위함이다. 본 연구에서 제시하는 BatMax 알고리즘(Algorithm 2)은 피크 타임의 에너지 수요를 미리 저장해 놓은 배터리의 전력을 사용하여 분산시키는 것이다. BatMax 알고리즘은 다음과 같이 동작한다. 1) input으로 minMax 알고리즘에 의해 스케줄링 된 결과를 사용한다. 2) 마지막 시간 슬롯에서부터 에너지 수요의 평균(처음부터 현재 슬롯까지의 평균)을 넘는 양은 앞쪽 시간 슬롯으로 이동시킨다. 3) 이를 반복하면서 모든 시간 슬롯에 대해 더 이상 앞쪽으로 이동 시킬 에너지가 없으면 알고리즘을 마친다.

Algorithm 2 BatMax Scheduling

Use output of minMax as input:

n number of time slots, m number of demands

For each time slot $t \in T$ where $1 \leq t \leq n$:

Aggregate energy demand $E_t = \sum_j r_j$ for all task j assigned to t

b_t is the battery buffer at time t and B is the charging rate of battery per hour

Output:

b_t for all time slot t

```

1: Initialize  $b_t$  to zeros
2: Compute  $avg = \sum_t^n E_t/n$ 
3: for  $i = n$  to 2 do
4:   if  $E_t + b_t < avg$  then
5:      $avg = avg + \frac{avg - E_t - b_t}{i-1}$ 
6:   end if
7:   if  $E_t > avg$  then
8:      $o = E_t - avg$ 
9:     for  $k = i - 1$  to 1 do
10:      Find time slot  $k$  where  $E_k + b_k < avg$ 
11:      Fill  $b_k$  with  $o$  such that  $E_k + b_k = avg$ , or  $b_k = B$ , or  $o = 0$ 
12:      Break loop when  $o = 0$ 
13:    end for
14:    if  $o > 0$  then
15:       $E_i = E_i + o$ 
16:       $avg = avg - \frac{E_t + b_t - avg}{i-1}$ 
17:    end if
18:  end if
19: end for

```

BatMax 알고리즘의 시간복잡도는 $O(n)$ 이며 n 은 시간 슬롯의 개수이다. BatMax 알고리즘을 통해 배터리 사용의 optimal 결과를 얻을 수 있다. 만약에 평균을 넘어서는 에너지 수요에 대해서 뒤쪽 시간 슬롯으로 이동 시킬 수 있다면 우리는 항상 매 시간 일정한 양의 에너지를 소비할 것이 당연하다. 하지만 배터리의 충전은 에너지 사용 시간 보다 항상 앞서야 하기 때문에 다음과 같은 증명이 필요하다.

LEMMA 1: 시간 슬롯 1에서 k 까지, 주어진 에너지 수요 E 에 대해서 k 슬롯의 에너지 누적 에너지 수요 E_k 가 평균을 넘는다고 할 때, $E_k - avg$ 만큼을 항상 시간 슬롯 1에서 $k-1$ 까지로 이동 시킬 수 있다. 왜냐하면 k 슬롯에서 평균 이상의 수요가 있기 때문에 나머지 슬롯의 에너지 수요의 합은 $E_k - avg$ 만큼 낮으므로 평균 이하의 에너지 수요를 가진 슬롯이 존재한다.

THEOREM 1: BatMax 알고리즘은 모든 시간 슬롯 1에서 n 까지의 에너지 수요를 최대한으로 분산시킨다.

LEMMA 1에서 각 iteration i 에서 BatMax 알고리즘은 시간 슬롯 i 를 최적으로 만든다. 배터리 용량이 충분히 크다면 항상 평균만큼의 에너지 수요만 남겨놓고 앞쪽 시간 슬롯으로 수요를 이동 시킬 수 있기 때문이다. 배터리 충전에 의해 에너지 수요가 증가하는 시간 슬롯에서 역시 평균 수요를 넘지 않는 선에서 충전을 하기 때문에 항상 평균 수요를 유지한다. 이 경우 에너지 수요가 최대한으로 분산 되었음은 당연하다. 만약 평균 수요보다 현재 시간 슬롯의 에너지 수요가 더 낮을 경우, 현재 시간슬롯 이전의 에너지 수요 평균은 그만큼 높아지게 된다. 때문에, 다음 iteration으로 넘어가게 되면 시간 슬롯 1에서 슬롯 $i-1$ 까지의 높아진 평균을 사용하여 에너지 수요를 다시 평균으로 분산시킨다. 다시 말해, BatMax 알고리즘은 주어진 조건에서 에너지 수요를 최대로 분산시키도록 수요를 이동시킨다.

제시하는 BatMax 알고리즘의 배터리 충전 효율과 용량을 구하기 위해 다음의 과정을 거쳤다. E_p 를 minMax 알고리즘에 의해 얻어진 결과에서의 peak demand라고 했을 때 F_p 를 E_i 가 전체 평균을 넘기는 연속된 시간 슬롯이라고 하자. 같은 방법으로, E_v 를 가장 낮은 에너지 수요를 나타내고 F_v 가 전체 평균을 넘지 않는 연속된 시간 슬롯이라고 할 때, 우리는 배터리 용량을 다음과 같은 식으로 근사 하였다.

$$0.8 * \min \{ (E_p - avg) \max(F_p), (avg - E_v) \max(F_v) \}$$

그리고 시간당 배터리 충전 비율은 다음과 같이 정의했다.

$$0.8 * \min \{ (E_p - avg), (avg - E_v) \}$$

즉, 24시간 중 연속적으로 충전 혹은 충전 가능해야 하는 에너지 양의 80%를 배터리 용량으로 보았으며, 단위 시간 동안 최대로 충전 혹은 충전 가능해야 하는 에너지 양의 80%를 충전 효율로 보았다. 예를 들어 38개의 작업을 스케줄링 할 때, 시간당 에너지 수요의 평균이 1.747 kW이고 E_p 는 4.669 kW, F_p 는 3시간, E_v 는 1.175 kW, F_v 는 역시 3시간 일 때, 적절한 배터리 용량은 4.193 kW이고 충전 효율은 시간당 1.398 kW이다. GE의 new WattStation 프로젝트를 살펴보면 최신 EV 배터리 기술은 시간당 1.71 kW를 충전할 수 있으며 24 kW의 용량을 갖는다. 이는 본 논문에서 요구하는 배터리 성능이 실용적인 범위 임을 보여준다.

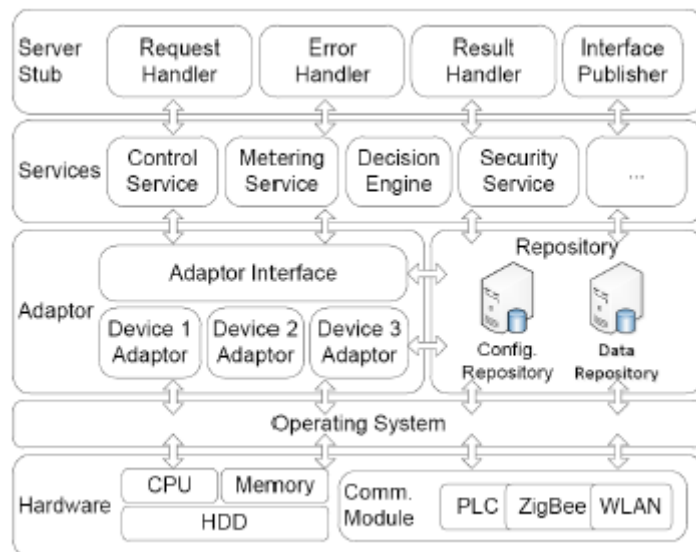


그림 3. Green Home Server 아키텍처

4. Green Home Service (GHS)

4.1 GHS 아키텍처

GHS는 가정용 서버 어플리케이션으로 스마트 그리드의 EMI를 통해 외부와 통신하며 DR 가격 정보 등을 받고 원격으로 전력을 제어 할 수 있게 한다. 뿐만 아니라 다양한 통신 프로토콜을 사용하여 주기적인 전력 모니터링과 제어, 플래닝을 가능하게 한다. GHS 아키텍처는 4개의 기본 컴포넌트로 구성되어 있다(그림 3).

- **Server Stub:** 웹 서비스 인터페이스로서 클라이언트 어플리케이션과 통신한다. Smart grid의 EMI역시 이 인터페이스를 통해 통신 가능하다.
- **Services:** GHS 검침, decision engine, 보안 서비스 등 기능들을 제공 해준다. minMax 알고리즘과 BatMax 알고리즘은 decision engine 서비스의 한 부분이 된다. 또한 GHS가 주기적으로 가전제품의 전력 사용량을 측정하고 스케줄링에 따른 가전제품 제어를 가능하게 한다.
- **Repository:** 검침 데이터와 장비 및 프로토콜 메타 정보를 저장하고 있는 DBMS이다. Repository는 사용자에게 따라 부동한 권한으로의 접근을 가능하게 한다.
- **Adapter:** 각각의 adapter는 가전제품 (통신 프로토콜)에 특화되어 구현된다. Adapter를 통해 GHS 서버는 가전제품과 통신 할 수 있다. 이는 아직 표준화 되지 않은 가전제품 제조사들의 통신 프로토콜과 데이터 포맷을 포괄하기 위함이다.

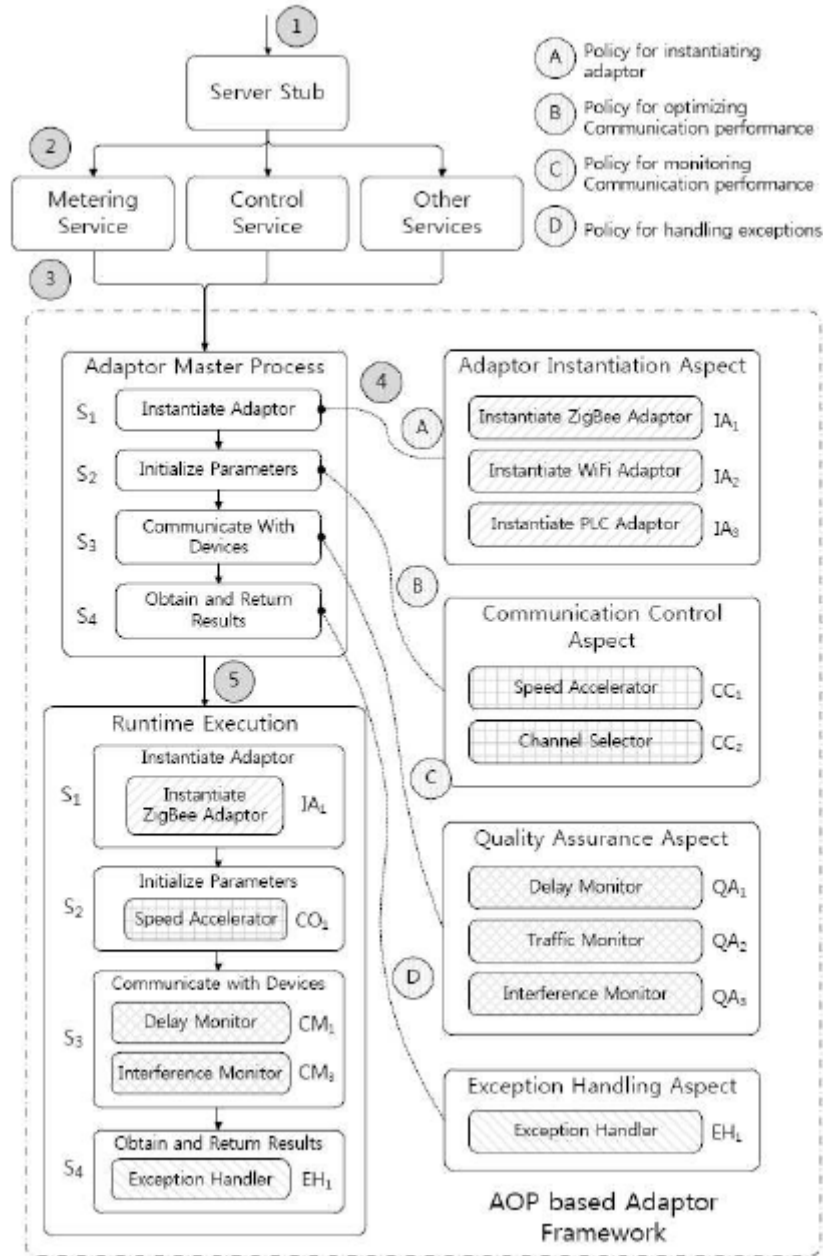


그림 4. AOP 기반의 Adapter framework

GHS가 효과적이고 자동적으로 동작하기 위해서 GHS서버는 가정 내 검침 장치 및 가전제품과 통신 할 수 있는 환경이어야 한다. 실제로 최근 출시되는 가전제품은 통신 인터페이스가 있어 외부와 통신 가능하다. 하지만 통신 프로토콜과 제어 및 상태 정보를 얻기 위한 데이터 포맷이 표준화되지 않은 상태이다. 따라서 GHS는 아래에 열거한 요구사항들을 만족 시켜야 한다.

- **상호 운용성:** GHS는 어플리케이션 및 사용자한테 특정 기기 전유의 프로토콜을 노출 시키지 않으면서 다양한 통신 프로토콜과 네트워킹 기술을 지원 가능해야 한다.
- **확장성:** 새로운 기능의 가전제품이나 새로운 통신 프로토콜을 차후 GHS에 추가하더라도 서비스의 중절이 발생하지 말아야 한다.
- **효율성:** 가전제품 및 클라이언트에 대한 Operation은 최적화 되어야 하며 통신뿐만 아니라 시스템 작동 시 발생하는 에러 까지도 적절하게 처리 되어야 한다.

위에 열거한 요구사항들을 만족 시키기 위하여 GHS의 Adapter 디자인 시 Aspect-Oriented Programming (AOP) 접근 방법이 사용되었다.

4.2 Aspect Oriented Programming in GHS

AOP 기반의 시스템 디자인은 마스터 프로세스 (GHS services) 구현함에 있어서 시스템 특정 로직(어플리케이션 adaptor 로직)과의 얽힘에서 분리한다. GHS의 경우, adapter가 특정 이벤트를 처리하기 위하여 호출을 하면, Pointcut이라고 불리는 지점에서 우리가 별도로 정의한 advice가 weave 되어 전체 프로세스가 만들어지고 이는 최종적으로 runtime시 실행 된다. 이는 마스터 프로세스를 명시적인 방식으로 hooking하는 전통적인 프로그래밍 방법과 차별화 되는 점이다. 그러므로, 특정 가전제품의 프로토콜을 위한 프로그래밍 코드 로직이나 함수 호출은 시스템 runtime시 쉽게 추가/삭제가 가능하다. 또한 이런 동적인 advice의 추가 및 삭제는 시스템 halt를 필요로 하지 않는다. Aspect는 Pointcut과 Advice의 조합으로 GHS 시스템 구현 시 policy 형태로 정의 되었다[15].

그림 4는 AOP기반의 adapter framework을 보여준다. Adapter Master Process (AMP)는 general한 프로세스이며 다양한 adaptor들을 지원 하기 위하여 아래와 같은 흐름을 통하여 runtime시 자체 프로세스를 재구성 한다. 1) Server Stub가 다른 종류의 클라이언트로부터 요청을 받는다. 2) Server Stub이 Name Space와 서비스 이름을 사용해 요청을 구분한다. 3) 서비스는 특정 요청을 해석하고, 해석된 결과를 근거로 가전제품과 통신한다. 4) 이 단계에서 AMP가 invoke 되지만 곧바로 실행 되지는 않는다. AMP invoke가 발생하면 adaptor framework은 관련 aspect들을 미리 정의된 policy를 근거로 선택한다. 5) 마지막으로 앞 단계에서 선택된 aspects는 runtime시 AMP에 통합되어 하나의 완전한 프로세스를 구성하며 최종 실행된다.

5. 시뮬레이션

minMax 알고리즘과 BatMax 알고리즘을 검증하기 위하여 실제 국내 가전제품 사용 패턴을 기반으로 한 데이터를 생성하여 사용하였다[16]. 이 조사 자료를 바탕으로 시간대별 주요 10개 가전제품 사용 비율을 계산하였다. 그림 5는 주요 5개 가전제품의 시간대별 사용 비율을 보여주고 있다. 또한 전체 가전제품의 시간대별 사용 비율과 소비 전력을 사용하여 시장 가격을 기반으로 한 DR 가격을 예측하였다(그림 6). 이는 실제 한국 전력의 현재 과금 정책을 기반으로 하여 계산되었다. 평균 가격을 기준으로 수요 변화에 따라 제품 스케일의 인센티브를 부여하고 이를 $P(r_j)$ 로 정의하였다.

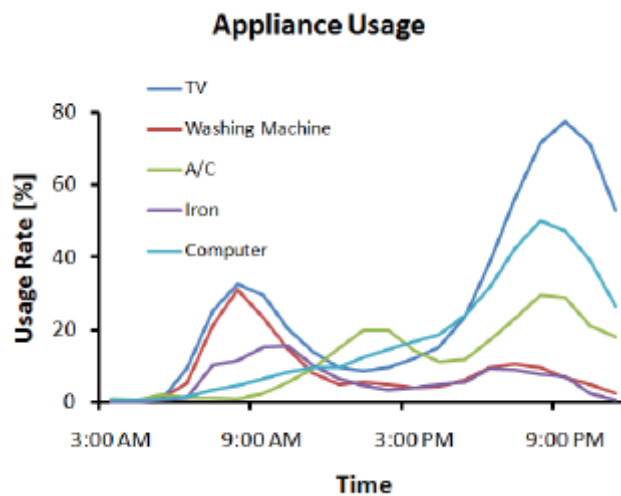


그림 5. 주요 5개 가전제품의 시간대별 사용 비율

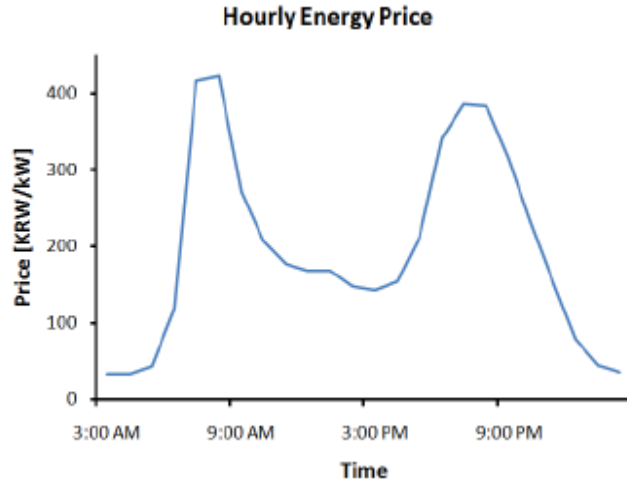


그림 6. 하루 중 시간대별 전력 가격 변화 예측

표 1. 가전제품 특성 분류 및 시뮬레이션 상수

Category	Appliance	Shiftable time	Energy rate (Watt/hr)	Task length
Fixed schedule	TV	0	200	3
	Cooker	0	1400	2
	Microwave Oven	0	100	1
Tight schedule	Computer	1	500	3
	Hair Drier	1	500	1
	Iron	1	900	1
Loose schedule	Audio	3	70	4
	Washing Machine	3	130	2
	Vacuum Cleaner	3	600	1
	A/C	4	1800	2

표 1은 10개 가전제품에 대한 특성 분류와 시뮬레이션 상수이다. 가전제품은 Fixed, Tight, Flexible 3개의 카테고리로 분류된다.

- **Fixed:** 사용자가 사용 시간대를 조정 할 수 없거나 조정하지 않아야만 하는 가전제품 군
- **Tight:** 사용자가 짧은 시간의 사용 시간대 조정은 허용 할 수 있는 가전제품 군
- **Flexible:** 몇 시간 정도의 사용 시간 조절을 허용 할 수 있는 가전제품 군

이러한 분류는 사용자에게 따라 달라질 수 있으며 본 연구의 시뮬레이션을 위해 조정 가능 시간과 작업 시간 등을 임의로 정하여 실험하였다.

시뮬레이션을 위해 생성된 데이터는 100 가구의 에너지 사용에 해당하는 데이터로 각 가정은 30에서 45개 정도의 작업을 스케줄링 하게 된다. 스케줄링은 각 가정 단위로 24시간마다 수행되며 이웃 가정의 사용 데이터는 서로 알 수 없다고 가정하였다. 스케줄링은 매일 새벽 3시 정각에 수행하였는데, 이는 전력 사용량이 매우 적은 새벽시간대가 배터리를 충전 하기에 최적의 시간대이기 때문이다.

그림 7은 100가구 각각에 대하여 minMax 알고리즘을 적용하고 이를 모두 누적시킨 결과를 보여주고 있다. 왼쪽 그림은 스케줄링 알고리즘을 전혀 사용하지 않았을 때의 에너지 가격을 시간대 별로 나타낸 것이다. 오전 시간과 저녁 시간에 한번씩 피크타임이 존재하는 것을 알 수 있다. 가운데 그림은 minMax 알고리즘을 적용 시킴으로써 오전 시간 대와 저녁시간 대에 각각 16%와 22% 최대 전력 수요가 낮아졌음을 보인다. 이는 오전 시간의 작업들은 저녁 시간의 작업보다 스케줄링이 어려운 fixed나 tight 카테고리의 작업이 더

많음을 나타낸다. 가장 오른쪽의 히스토그램은 최대 전력 수요 감소 비율에 따라 가구를 구분 한 것이다. 실제로 많은 수의 가정에서 최대 전력 수요가 크게 낮아지는는 않았음을 보이고 있다. 이는 실제 가전제품 사용 패턴을 기반으로 한 데이터를 사용한 결과로 실제 전력 사용자들의 행태를 어느 정도 반영 하고 있다고 볼 수 있다. 하지만 100가구 누적 결과를 보았을 때에는 가격적인 측면에서 하루 중 최대 전력 수요를 낮추는 효과를 가져옴을 알 수 있다.

그림 8은 시간대별 소비 전력에 대하여 minMax 알고리즘 사용 전후를 비교한 것이다. minMax 알고리즘만을 사용하였을 경우 수요의 분산이 이루어지고는 있으나 fixed, tight 한 작업에 따라 상당히 제한적으로 이루어지고 있음을 보여준다. 저녁 시간대의 에너지 수요가 minMax 알고리즘을 통해 밤 시간으로 많이 이동하였음을 볼 수 있다. 가장 오른쪽 그림은 minMax 스케줄링의 결과에 BatMax 알고리즘을 적용한 결과이다. 배터리 충전은 시간당 1.71 kW라고 가정하였다. 그 결과 스케줄링을 하지 않았을 때에 비하여 약 30%의 최대 전력 수요가 분산되었다.

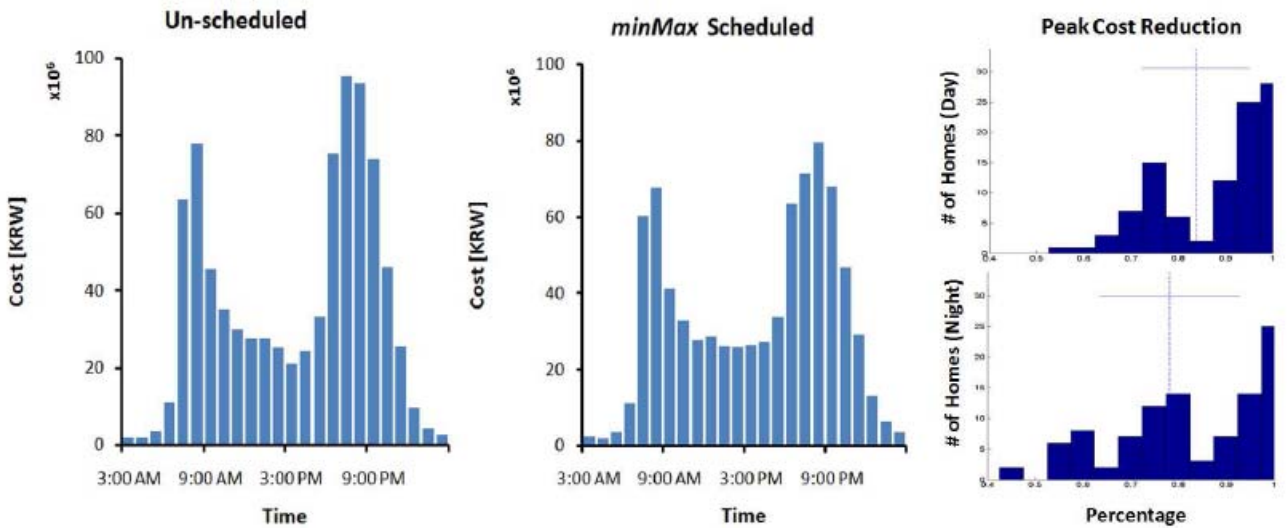


그림 7. 스케줄링 유무에 따른 100 가구의 시간대별 에너지 가격 누적 결과와 최대 전력 수요 분포 히스토그램

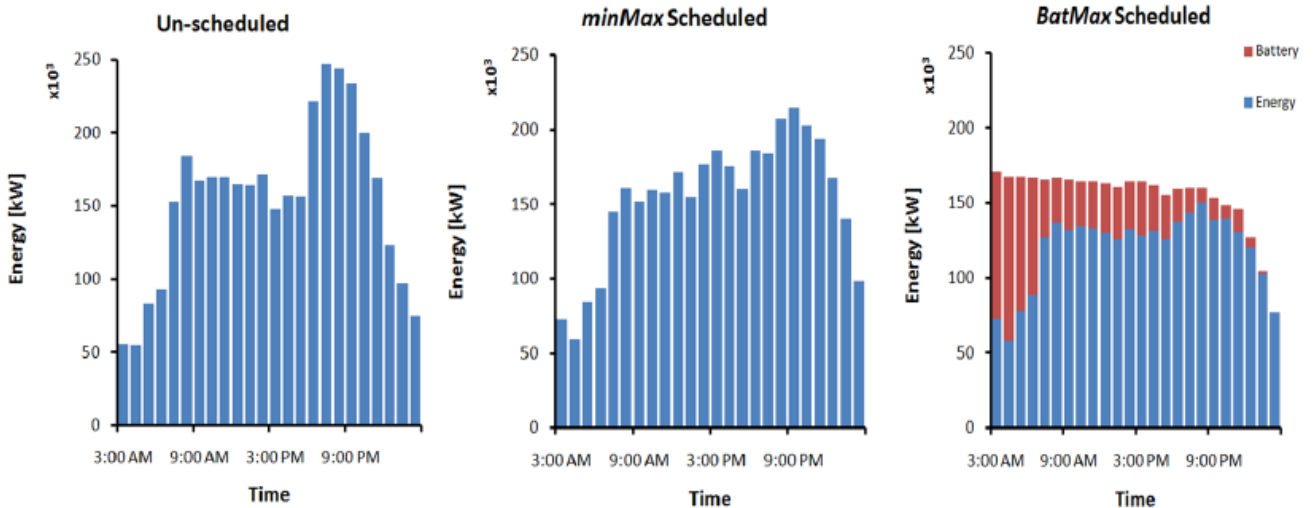


그림 8. 스케줄링 유무에 따른 100 가구의 시간대별 에너지 수요 누적 결과와 배터리 사용을 통한 전력 수요 분산 결과

6. 결론

본 논문에서는 수요 측면의 에너지 관리 문제를 작업 스케줄링 문제로 모델링 하였다. 또한 minMax 알고리즘을 사용하여 사용자가 지정한 시간 내에서 작업을 스케줄링 하여 에너지 소비 시간의 조정에 따른 불편함을 줄여주고, 가정용 고용량 배터리를 사용한 BatMax 스케줄링 알고리즘을 제안함으로써 보다 효과적으로 전력 수요를 분산 시킬 수 있음을 보였다. 특히 본 논문에서 제안하는 알고리즘은 전력 수요의 분산, 소비자의 에너지 사용 가격에 대한 하락, 사용자의 안정성 추구에 대한 요구를 만족시킬 수 있는 점에 대하여 탁월한 결과를 나타내었다. AOP를 기반으로 한 GHS 아키텍처는 가정용 에너지 관리 인터페이스로서 상호 운용성, 확장성, 효율성을 고려하여 설계하고 구현하였다.

작업 스케줄링 알고리즘을 검증하기 위하여 실제 한국 전력에서 조하산 데이터를 바탕으로 100 가구의 작업을 생성하였으며, 각 가구마다 개별적으로 스케줄링 작업 결과를 모두 누적하였을 경우에도 역시 에너지 수요가 효과적으로 분산되었음을 보였다.

앞으로 진행될 연구에서는 실제 가정에 GHS를 배포하여 필드 테스트 실험을 수행하고 기업, 빌딩 등 보다 큰 규모의 에너지 관리 환경에 적용 시킬 계획이다.

참고 문헌

- [1] G. C. Heffner, "Configuring load as a resource for competitive electricity markets - review of demand response programs in the us and around the world," in 14th Annual Conference of the Electric Power Supply Industry (CEPSI 2002), 2002.
- [2] S. Borenstein, "Dynamic pricing, advanced metering and demand response in electricity markets," The Theory of Demand-side Price Incentives, Hewlett Foundation Energy Series., 2002.
- [3] L. Ruff, "Economic principles of demand response in electricity," Report to the Edison Electric Institute., Washington D.C., 2002.
- [4] C. Goldman, N. Hopper, O. Sezgen, M. Moezzi, R. Bharvirkar, B. Neenan, D. Pratt, P. Cappers, and R. Boisvert, "Customer response to day-ahead wholesale market electricity prices: Case study of RTP program experience in new york," White Paper, LBNL-54761, Ernest Orlando Lawrence Berkeley National Laboratory, University of California Berkeley, 2004.
- [5] Y. Kojima, M. Koshio, S. Nakamura, H. Maejima, Y. Fujioka, and T. Goda, "A demonstration project in hachinohe: Microgrid with private distribution line," in IEEE International Conference on Systems Engineering, 2007.
- [6] A. Molderink, V. Bakker, M. Bosman, J. Hurink, and G. Smit, "Domestic energy management methodology for optimizing efficiency in smart grids," in IEEE conference on Power Technology, June 2009.
- [7] —, "Management and control of domestic smarg grid technology," IEEE Transactions on Smart Grid, vol. 1, no. 2, pp. 109–119, 2010.
- [8] K. Kok, Z. Derzsi, J. Gordijn, M. Hommelberg, C. Warmer, R. Kamphuis, and H. Akkermans, "Agent-based electricity balancing with distributed energy resources, a multiperspective case study," in the 41st Annual Hawaii International Conference on System Sciences, January 2007.
- [9] A.-H. Mohsenian-Rad, V. Wong, J. Jatskevich, and R. Schober, "Optimal and autonomous incentive-based energy consumption scheduling algorithm for smart grid," in Innovative Smart Grid Technologies 2010 (ISGT 2010), January 2010.
- [10] C. Wang, M. Groot, and P. Marendy, "A service-oriented system for optimizing residential energy use," in IEEE International Conference on Web Services, July 2009.
- [11] R. Rachamadugu, "Scheduling jobs with proportionate early/tardy penalties," IIE Transactions, vol. 27, no. 5, pp. 679–682, 1995.
- [12] S. Karnouskos, O. Tezidis, and P. Karnouskos, "An advanced metering infrastructure for future energy networks," in IFIP/IEEE 1st International Conference on New Technologies, Mobility and Security, April 2007.
- [13] Q. Dam, S. Hohagheghi, and J. Stoupis, "Intelligent demand response scheme for customer side load management," in IEEE Energy 2030, November 2008.
- [14] J. Xiao, J. Chung, J. Li, R. Boutaba, and J.-K. Hong, "Near-optimal demand-side energy management under real-time demand-response pricing," in IEEE Conference on Network and Services Management (CNSM 2010), mini-conference, October 2010.
- [15] P. Greenwood and L. Blair, "A framework for policy driven auto-adaptive systems using dynamic framed aspects," Lecture Notes in Computer Science (LNCS), vol. 4242, 2006.
- [16] Korean Power Exchange, "Survey of appliance penetration rate and domestic power usage," <http://epsisibook.kpx.or.kr/web/http/kpx/kpx/Gindex.php?cateCode=ABAAAG>, December 2009.



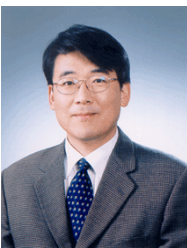
정재윤

2009 포항공과대학교, 컴퓨터공학과 학사
2009 ~ 현재 포항공과대학교, 컴퓨터공학과 석박사 통합 과정
<관심분야> 네트워크 트래픽 분석, 네트워크 관리, 스마트 그리드



리건

2007 연변과학기술대학교, 전자통신공학과 학사
2009 ~ 현재 포항공과대학교, 컴퓨터공학과 통합 과정
<관심분야> 클라우드 컴퓨팅, 스마트 그리드 네트워크 관리



홍원기

1983 Univ. of Western Ontario, BSc in Computer Science
1985 Univ. of Western Ontario, MS in Computer Science
1985 ~ 1986 Univ. of Western Ontario, Lecturer
1986 ~ 1991 Univ. of Waterloo, PhD in Computer Science
1991 ~ 1992 Univ. of Waterloo, Post-Doc Fellow
1992 ~ 1995 Univ. of Western Ontario, 연구교수

1995 ~ 현재 포항공과대학교 컴퓨터공학과 교수
2007 ~ 2009 포항공과대학교 정보통신연구소 연구소장
2008 ~ 2009 포항공과대학교 컴퓨터공학과 주임교수
2007 ~ 현재 포항공과대학교 정보통신대학원장
2008 ~ 현재 포항공과대학교 정보전자융합공학부장
<관심분야> 네트워크 트래픽 모니터링, 네트워크 및 시스템 관리