

<https://doi.org/10.7236/JIIBC.2020.20.1.135>  
JIIBC 2020-1-19

## 멀티코어 이기종메모리 환경에서의 유전 알고리즘 기반 실시간 전력 절감 스케줄링

### Real-Time Power-Saving Scheduling Based on Genetic Algorithms in Multi-core Hybrid Memory Environments

류수현\*, 조예원\*, 조경운\*\*, 반효경\*\*\*

Suhyeon Yoo\*, Yewon Jo\*, Kyung-Woon Cho\*\*, Hyokyung Bahn\*\*\*

**요약** 최근 사물인터넷, 지능형 시스템 등의 활성화로 실시간 임베디드 시스템의 전력 절감 기술이 중요해지고 있다. 본 논문은 멀티코어 이기종메모리 환경에서 실시간 시스템의 전력 소모량을 절감하는 P-GA (parallel genetic algorithm) 스케줄링 알고리즘을 제안한다. P-GA는 멀티코어를 위한 PF (proportional fairness) 알고리즘에 기반한 프로세서의 전압 및 주파수 동적 조절 기법에 차세대 비휘발성메모리 기술을 결합하여 시스템의 전력 소모를 더욱 줄인다. 특히, 유전 알고리즘을 사용하여 태스크별 수행 프로세서의 전압 및 주파수 모드와 메모리의 종류를 최적화하여 태스크 집합의 전력 소모량을 최소화한다. 시뮬레이션 실험을 통해 P-GA가 기존 방식 대비 최대 2.85배의 전력 소모량을 감소할 수 있음을 보인다.

**Abstract** Recently, due to the rapid diffusion of intelligent systems and IoT technologies, power saving techniques in real-time embedded systems has become important. In this paper, we propose P-GA (Parallel Genetic Algorithm), a scheduling algorithm aims at reducing the power consumption of real-time systems in multi-core hybrid memory environments. P-GA improves the Proportional-Fairness (PF) algorithm devised for multi-core environments by combining the dynamic voltage/frequency scaling of the processor with the nonvolatile memory technologies. Specifically, P-GA applies genetic algorithms for optimizing the voltage and frequency modes of processors and the memory types, thereby minimizing the power consumptions of the task set. Simulation experiments show that the power consumption of P-GA is reduced by 2.85 times compared to the conventional schemes.

**Key Words** : DVFS, Genetic Algorithm, Hybrid Memory, Multi core, Real Time, Scheduling.

\*비회원, 이화여자대학교 컴퓨터공학과  
S. Yoo and Y. Jo contributed equally to this work.  
\*\*준회원, 이화여자대학교 임베디드소프트웨어연구센터  
\*\*\*정회원, 이화여자대학교 컴퓨터공학과(교신저자)  
접수일자: 2019년 12월 30일, 수정완료: 2020년 1월 26일  
게재확정일자: 2020년 2월 7일

Received: 30 December, 2019 / Revised: 26 January, 2020 /  
Accepted: 7 February, 2020

\*Corresponding Author: bahn@ewha.ac.kr  
Dept. of Computer Engineering, Ewha University, Korea

## I. 서 론

최근 사물인터넷, 스마트 빌딩 등 다양한 분야에서 5세대 이동통신, 인공지능 기술 등이 적용되면서 프로세서의 연산요구량이 급격히 증가하고 있다<sup>[1][6]</sup>. 이를 위해 모바일 환경에도 멀티코어, 매니코어 등의 도입으로 프로세서의 연산 능력을 향상시키는 추세이다<sup>[2]</sup>. 이러한 시스템들은 주어진 시간 내에 특정한 일을 수행해야 하는 실시간 임베디드 시스템으로 한정된 배터리를 전력 공급의 수단으로 사용한다. 따라서, 전력 절감 기술의 중요성이 더욱 강조되고 있다.

본 논문에서는 (1) 경성 태스크 집합에 대해 멀티코어 시스템의 이용률을 최대로 높일 수 있는 PF 스케줄링 알고리즘을 바탕으로, (2) DRAM과 비휘발성 차세대 메모리를 함께 사용하여 메모리의 전력 소모를 줄이는 방법(이하 HM: Hybrid Memory), (3) 프로세서의 전압을 동적으로 조절하여 프로세서의 전력 소모를 낮추는 방법(이하 DVFS, Dynamic Voltage/Frequency Scaling), (4) 메타휴리스틱 알고리즘의 일종인 유전알고리즘을 이용하여 각 태스크에 할당될 프로세서의 전압 및 주파수와 메모리 종류를 결정하는 방법을 결합하여 시스템의 전력 소모를 최소화하는 P-GA(Parallel Genetic Algorithm)를 제안한다.

## II. 본 론

### 1. 프로세서의 전압 및 주파수 조절 방법

프로세서의 전력 절감을 위해 사용되는 대표적인 방법으로는 CMOS 회로 기술을 바탕으로 프로세서의 전압과 주파수를 동적으로 조절하는 DVFS 기법이 있다<sup>[3]</sup>. 프로세서의 소비 전력은 태스크를 수행할 때 소비되는 '동적 소비 전력'과 유휴 상태에 소비되는 '정적 소비 전력'으로 나눌 수 있다<sup>[1]</sup>. 동적 소비 전력은 식 (1)에서 보는 것처럼 전압의 제곱에 비례한다. 따라서, 전압을 낮춤으로써 동적 소비 전력을 크게 감소시킬 수 있다. 하지만 이를 낮추면 프로세서가 단위 시간당 수행하는 명령어의 수가 줄기 때문에, 태스크의 수행시간은 증가한다. 그러므로, 태스크의 데드라인을 위반하지 않는 범위에서 주파수와 전압을 적절히 조절하는 것이 중요하다.

$$E \cong E_{dynamic} = C \times f \times V^{2xt} \quad (C \text{는 상수}) \quad (1)$$

### 2. 비휘발성 차세대 메모리를 이용하는 방법

메인메모리 시스템에 사용되는 DRAM은 휘발성 매체로 데이터의 유지를 위해 유휴 상태에서도 지속적인 전력 재공급 연산(refresh)이 필요하다<sup>[5]</sup>. 반면 PRAM, STT-MRAM 등 차세대 저전력 메모리(이하 LPM, Low Power Memory)는 비휘발성이므로, 전력 재공급 연산이 필요하지 않아 전력 소모를 줄일 수 있다. 그러나 표 1과 같이 LPM의 성능이 DRAM보다 떨어지므로, 메모리 매체를 LPM으로 전면 대체하는 것은 어려운 일이다. 본 논문은 DRAM과 LPM을 함께 사용하는 HM 방식으로 메모리의 전력 소모를 줄인다.

표 1. DRAM과 LPM의 성능 및 전력 소모량

Table 1. Latency and power consumption of Memories

	DRAM	LPM
Read latency	50 (ns)	100 (ns)
Write latency	50 (ns)	350 (ns)
Read energy	0.1 (nJ/bit)	0.2 (nJ/bit)
Write energy	0.1 (nJ/bit)	1.0 (nJ/bit)
Idle power	1 (W/GB)	0.1 (W/GB)

### 3. 유전 알고리즘을 이용하는 방법

본 논문에서는 각 태스크가 수행될 프로세서의 전압 및 주파수 모드와 메모리 종류를 미리 지정하여 전력 소모를 절감한다. 이때, 전역 최적해를 찾기 위해서는 가능한 모든 조합에 대해 태스크 집합의 전력 소모량을 계산하는 지수 시간 복잡도를 필요로 한다. 따라서, 본 논문은 유전 알고리즘을 적용하여 주어진 시간 내에 최적해의 근삿값을 구한다.

유전 알고리즘은 생물의 진화과정을 모방하여 근사 최적해를 찾아내는 방법으로, 문제의 해를 염색체로 인코딩하는 과정이 필요하다. 본 논문은 하나의 해가 프로세서의 전압 및 주파수 모드와 메모리의 종류를 나타내는 "두 개의 문자열"로 인코딩된다. 이때, 각 문자열의 길이는 태스크 집합에 속하는 태스크의 수가 된다. 문자열의  $i$ 번째 원소는 태스크  $i$ 의 전압 및 주파수 모드와 메모리 정보를 나타낸다. 예를 들어, 4개의 프로세서 모드(1, 0.5, 0.25, 0.125)와 2종의 메모리(DRAM, LPM)가 있는 시스템에 8개의 태스크가 존재하는 경우 그림 1과 같이 인코딩될 수 있다. 그림에서 태스크 3은 메모리 1번에 적재되어 프로세서 모드 0번으로 실행된다는 정보를 담고 있다. 본 논문에서는 이러한 인코딩 방식에 따라 100개의 초기 해집합을 무작위로 생성한다.

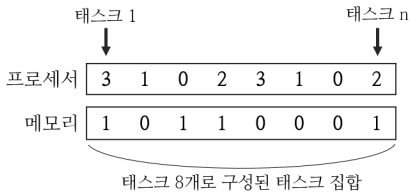


그림 1. 유전알고리즘 인코딩 방식  
 Fig. 1. Encoding of the proposed genetic algorithm

유전 알고리즘에서는 해의 우수성을 판별하기 위한 적합도 함수의 정의가 필요하다. 본 연구는 태스크의 데드 라인을 위반하지 않는 범위에서 전력 소모가 최소인 프로세서 모드와 메모리 배치 조합을 찾는 문제로, 태스크 집합의 총 전력 소모량을 적합도를 위한 비용 함수로 정의한다. 만약 해가 스케줄링 불가능하다면 데드라인을 위반한 정도를 반영한 페널티를 비용에 추가한다.

교배할 해의 선택에는 룰렛-휠 방식을 사용한다. 해의 적합도  $f$ 는 식 (2)와 같이 계산한다.  $C_i$ 는 해당 해의 비용,  $C_w$ 는 해집합 내에서 가장 비용이 큰 해의 비용이고,  $C_b$ 는 비용이 가장 작은 해의 비용이다. 해  $i$ 가 선택될 확률은 식 (3)에 의해 계산된다. 결론적으로, 해집합 내에서 가장 좋은 해가 선택될 확률은 가장 나쁜 해가 선택될 확률의  $k$ 배가 된다.

$$f_i = (C_w - C_i) + \frac{(C_w - C_b)}{(k - 1)} \quad (2)$$

$$P_i = \frac{f_i}{\sum_{j=0}^{j=n} f_j} \quad (3)$$

선택된 두 해의 교배는 프로세서의 정보를 나타내는 염색체와 메모리 종류를 나타내는 염색체에 대해 각각 진행한다. 그림 2와 같이 각 염색체에 무작위로 교배점을 선택한 후, 그 지점을 기준으로 두 해를 합병한다. 해의 다양성을 높이기 위해 교배 시마다 일정 확률로 돌연변이를 발생시킨다. 그림 3과 같이 돌연변이는 프로세서를 나타내는 염색체와 메모리를 나타내는 염색체 중 하나에만 영향을 미치며, 무작위로 선택된 서로 다른 두 태스크의 속성(유전자)이 교환된다.

#### 4. 시스템 모델

실시간 임베디드 시스템에서 수행되는 태스크 집합  $J = \{j_1, j_2, j_3, \dots, j_n\}$ 이 있다고 가정한다. 각각의 태스크  $j_i$ 는  $\langle t_i, p_i, m_i \rangle$ 로 나타낼 수 있으며,  $t_i$ 는 최약수행시간,  $p_i$ 는

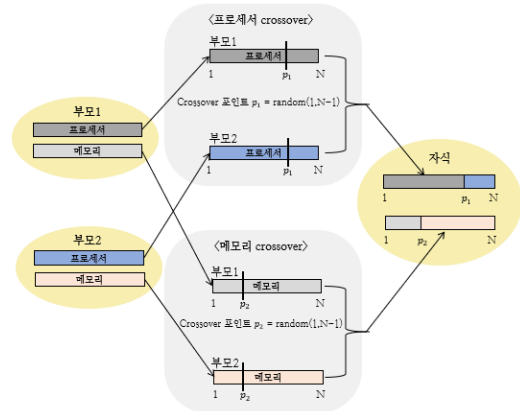


그림 2. 유전 알고리즘의 교배연산  
 Fig. 2. Crossover process of genetic algorithms



그림 3. 유전 알고리즘의 돌연변이 연산  
 Fig. 3. Mutation process of genetic algorithms

주기,  $m_i$ 는 메모리 용량이다. 모든 태스크는 주기성을 가지며, 각 주기  $p_i$ 가 태스크의 데드라인을 결정한다. 대부분의 실시간 스케줄링 연구와 같이 본 논문의 모델에서도 아래와 같은 가정을 바탕으로 스케줄링을 수행한다 [1][4].

- 가정 1. 각 태스크는 독립적이다.
- 가정 2. 코어의 수는  $M$ 개이며, 성능은 모두 같다.
- 가정 3. DVFS 방법을 이용하여 프로세서의 전압과 주파수를 코어별로 조정할 수 있다.
- 가정 4. 시스템의 메모리는 DRAM과 LPM이 혼합된 형태이다.
- 가정 5. DRAM의 크기는 모든 태스크 집합을 할당할 수 있는 크기이지만, LPM에 배치된 태스크의 경우 DRAM의 해당 부분은 절전모드로 전환하여 전력제공 연산을 없앤다.
- 가정 6. 프로세서의 문맥교환의 오버헤드는 무시한다.
- 가정 7. 프로세서의 전압 및 주파수 모드를 변경할 때

발생하는 오버헤드는 무시한다.

가정 8. 한 태스크는 동시에 여러 코어에서 실행되지 않는다.

본 모델에서 태스크의 최악수행시간은 태스크가 수행될 코어의 전압 및 주파수 모드와 메모리 종류에 의해 결정된다. 만약, 태스크가 LPM에서 수행되며 DVFS 방법을 적용한 경우, 프로세서의 전압 및 주파수를 낮춤에 따라 늘어나는 수행시간과 LPM에서 수행됨에 따라 늘어나는 수행시간에는 오버랩되는 부분이 존재할 수 있다. 이를 고려하여 태스크의 최악수행시간은 태스크가 수행될 프로세서 모드와 메모리 종류 중 더 큰 영향을 미치는 요소에 의해 결정된다. 즉, 식 (4)와 같이 태스크의 최악수행시간  $f(t_i)$ 는 DVFS와 LPM의 지연시간 중 최댓값에 의해 결정된다.

$$f(t_i) = \max \{f_{DVFS}(t_i), f_{LPM}(t_i)\} \quad (4)$$

멀티코어 환경에서의 실시간 스케줄링 알고리즘 중 PF 알고리즘은 모든 태스크의 데드라인을 위반하지 않으면서도 이론적으로 100%의 이용률을 가능하게 하는 알고리즘이다. PF 알고리즘에서 한 태스크  $j_i$ 의 비중  $W_i$ 는 (5)의 식으로 정의할 수 있으며, 이는 태스크의 최악수행시간을 주기로 나눈 값이다. 전체 시스템 이용률은 비중  $W_i$ 의 합이며, (6)의 식으로 정의한다. 태스크의 스케줄링 가능 여부는 (7)의 조건식 만족 여부에 따라 결정되며,  $M$ 개의 코어를 가진 시스템의 태스크 집합에 대해 전체 시스템 이용률  $U$ 는  $M$ 을 넘지 않아야 한다<sup>[4]</sup>.

$$W_i = \frac{f(t_i)}{p_i} \quad (5)$$

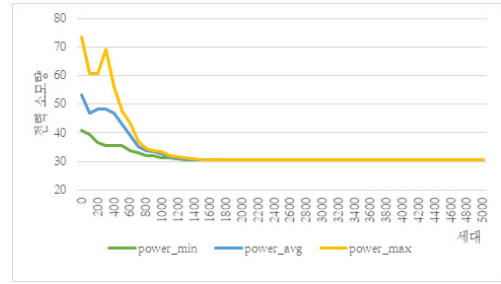
$$U = \sum W_i \quad (6)$$

$$U \leq M \quad (7)$$

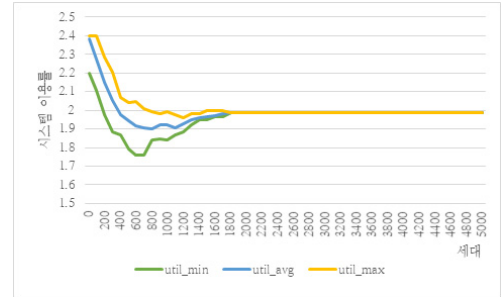
### III. 성능 평가

본 논문에서는 코어가 2개인 프로세서 환경을 가정하며, 프로세서의 이용률이 25%, 50%, 75%, 90%로 변경됨에 따른 성능을 평가하였다. 여기서 언급된 프로세서의 이용률은 LPM과 DVFS를 이용하지 않는 방식에서의 이용률을 의미한다. 이용률마다 20개의 태스크 집합을 생성하였으며, 하나의 태스크 집합은 태스크 100개로 구성된다.

비교 대상으로는 (1) 저전력 기법을 이용하지 않는 방



(a) 전력 소모량



(b) 시스템 이용률

그림 4. 유전알고리즘 진행에 따른 해의 수렴 결과  
Fig. 4. Convergence of solutions as GA progresses

식(ORIGINAL), (2) 비휘발성 메모리와 DRAM을 함께 사용하는 방식(HM), (3) 프로세서의 전압 및 주파수 조절 기법만을 적용한 방식(DVFS-DRAM), (4) 앞의 (2)와 (3)을 모두 사용하는 방식(DVFS-HM)과 본 논문이 제안한 P-GA를 비교하였다.

각 메모리의 소비 전력은 표 1에 정의된 값을 사용하였으며, 프로세서의 주파수 모드는 1, 0.5, 0.25, 0.125의 4가지를 사용하였다. 유전 알고리즘의 세대 수는 100,000으로 설정하였으며, 해집단의 크기는 100으로 하였다. 돌연변이 발생확률은 0.001%로 했으며, 해의 선택시 식 (2)의  $k$ 값은 4로 하였다.

그림 4는 유전 알고리즘의 세대 진행에 따른 전력 소모량과 시스템 이용률을 보여주고 있다. 그림에서 보는 것처럼 초기 해집단은 다양한 분포를 가지며 전력 소모량의 평균이 54인 반면, 세대를 진행할수록 30에 가깝게 수렴함을 확인할 수 있다. 또한, 그림 4(b)에서 보는 것처럼 초기 해집단에는 코어의 수인 2를 초과하는 해가 존재하지만, 페널티 부여를 통해 결국 비용이 적은 해, 즉 스케줄링 가능한 해로 수렴함을 확인할 수 있다. 결론적으로, 100,000 세대를 거쳐 진화함에 따라 해집단은 전력 소모량 및 이용률 측면에서 모두 수렴함을 확인할 수 있다.

그림 5는 시스템의 이용률 변화에 따른 제안 기법과

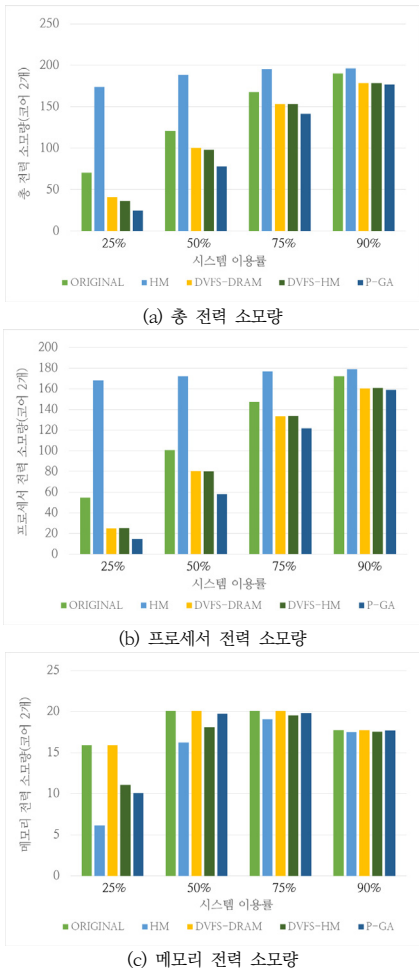


그림 5. 시뮬레이션 결과 전력 소모량의 비교  
 Fig. 5. Comparison of power consumptions

기존 기법 4가지의 전력소모량을 비교한 그래프이다. 그림 5(a)는 시스템의 총 전력 소모량을 나타낸 그래프로, 제안하는 기법이 모든 경우에 있어 가장 좋은 결과를 나타내었다. ORIGINAL 방식 대비 제안 기법의 전력소모량은 시스템 이용률 25%, 50%, 75%, 90%인 경우에 대해 각각 2.84배, 1.55배, 1.18배, 1.07배 개선된 것을 확인할 수 있다.

그림 5(b)는 프로세서의 전력 소모량을 나타낸 그래프이다. DVFS 기술을 사용하는 DVFS-DRAM, DVFS-HM, P-GA 방식이 DVFS를 사용하지 않는 방식보다 프로세서의 전력 소모량이 낮은 것을 확인할 수 있다. P-GA의 프로세서 전력 소모량은 ORIGINAL 방식 대비 시스템 이용률 별로 각각 3.70배, 1.72배, 1.21배,

1.08배 감소하였다.

그림 5(c)는 메모리 전력 소모량을 나타낸 그래프이다. LPM을 사용하는 HM, DVFS-HM, P-GA 방식이 그렇지 않은 방식보다 더 많은 메모리 전력 절감을 나타내었다. 그러나, LPM은 DRAM보다 읽기/쓰기 속도가 느리므로 태스크의 수행시간이 길어져 전력 소모를 증가시키는 결과를 초래할 수 있다. 본 실험에서도 LPM만을 사용하는 HM 방식이 가장 낮은 메모리 전력 소모를 나타내었지만, 프로세서의 전력 소모량을 크게 증가시켜 결국 가장 높은 전력 소모량을 발생시켰다. 결론적으로 LPM과 DVFS 방법의 균형점을 찾는 것이 전력 절감의 극대화에 중요한 요소이며, 제안하는 방법은 유전 알고리즘을 통해 LPM과 DVFS를 적절히 조합하여 전력 소모량을 최소화시켰다.

또 한가지 주목할 점은 시스템의 이용률이 낮을수록 P-GA의 전력 절감 효과가 좋다는 점이다. 이는 프로세서와 메모리의 이용률이 적은 상태, 즉 유휴 상태가 길수록 DVFS와 LPM의 적용 가능성이 높기 때문이다. 즉, 프로세서와 메모리의 이용률이 매우 높은 태스크 집합에 DVFS와 LPM 기술을 적용할 경우 데드라인을 위반할 가능성이 크기 때문이다.

#### IV. 결 론

본 논문은 멀티코어 환경의 스케줄링 알고리즘인 PF 알고리즘을 기반으로, 프로세서의 DVFS 기술, 메모리의 저전력 LPM 기술, 유전 알고리즘 등을 통해 실시간 태스크 집합의 전력 소모를 절감하는 P-GA 기법을 제안하였다. 이는 프로세서의 주파수 및 전압이 낮은 상태에서는 성능이 낮은 LPM을 이용하더라도 여전히 스케줄링이 가능할 수 있다는 점에 착안한 것이다. 유전 알고리즘을 통해 태스크별 프로세서의 전압 및 주파수 모드와 메모리를 전력 소모 측면에서 가장 적합하게 결정하여 DVFS 및 LPM 각각의 기술 대비 1.65배와 7.01배의 전력 소모량 절감효과를 확인하였다. 또한, DVFS와 LPM을 모두 사용하지 않는 기법 대비 2.84배, 모두 사용하는 기법 대비 1.45배의 전력 절감 효과를 확인하였다. 제안한 기법은 사물인터넷과 같은 저전력-배터리 기반 시스템에서 에너지 효율성을 제공하는 데에 도움이 될 것으로 기대된다.

## References

- [1] S. Nam, H. Bahn, "Real-time Task Scheduling Methods to Incorporate Low-power Techniques of Processors and Memory in IoT Environments," The Journal of The Institute of Internet, Broadcasting and Communication, Vol. 17, No. 2, pp. 1-5, 2017.  
DOI: <https://doi.org/10.7236/IIBC.2017.17.2.1>
- [2] S. Paik, K. Kim, J. Lee, C. Lee, "Design of Parameter Decision Genetic Algorithm for Multi-parameter Real-time Task Scheduling in Multiprocessor System," Proc. of KIISE, pp. 1338-1340, 2015.
- [3] J. Yoo, S. Heo, S. Hong, "Limitations on DVFS-Based Power Saving Mechanisms of Android Smartphones," Commun. of KIISE, Vol.30 No.7, pp. 9-16, 2012.
- [4] Y. Lee, Y. Doh, C. Krishna, "EDF scheduling using two-mode voltage clock scaling for hard real-time systems," Proc. ACM CASES Conf., pp. 221-228, 2001.  
DOI: <https://doi.org/10.1145/502217.502255>
- [5] M. Cho, K. Han, and W. Jang, "A Pre-Refresh Technique for Improving DRAM Performance," The Journal of KIIT, Vol.16, No.10, pp.1027-1035, 2018.  
DOI: <http://doi.org/10.14801/jkiit.2018.16.10.27>
- [6] H. Bahn, "Efficient scheduling of sensor-based elevator systems in smart buildings," Journal of the Korea Academia-Industrial cooperation Society, Vol.17, No.10, pp.367-372, 2016.  
<http://dx.doi.org/10.5762/KAIS.2016.17.10.367>

## 저 자 소 개

### 류 수 현(비회원)



- 2017년 3월 ~ 현재 : 이화여자대학교 컴퓨터공학 학사과정
- 관심분야 : 운영체제, 임베디드시스템

### 조 예 원(비회원)



- 2017년 3월 ~ 현재 : 이화여자대학교 컴퓨터공학 학사과정
- 관심분야 : 운영체제, 임베디드시스템

### 조 경 운(준회원)



- 1995년 2월 : 서울대학교 계산통계학과 학사
- 1997년 2월 : 서울대학교 전산학과 석사
- 2012년 2월 : 서울대학교 컴퓨터공학부 박사
- 2000년 ~ 2016년 : ㈜클루닉스 연구소장
- 2016년 4월 ~ : 이화여자대학교 임베디드소프트웨어연구센터 수석연구원/연구교수

### 반 효 경(정회원)



- 1997년 2월 : 서울대학교 계산통계학과 학사
- 1999년 2월 : 서울대학교 전산학과 석사
- 2002년 2월 : 서울대학교 컴퓨터공학부 박사.
- 2002년 9월 ~ : 이화여자대학교 컴퓨터공학과 교수

※ This research was funded by the ICT R&D program of MSIP/IITP (2019-0-00074, developing system software technologies for emerging new memory that adaptively learn workload characteristics) and also by the Basic Science Research Program through the NRF grant funded by Korea Government (MSIP) (No. 2019R1A2C1009275).