

이기종 무선센서 네트워크를 위한 원격 모듈 중개자*

민홍⁰¹, 이상호¹, 김봉재¹, 김석현¹, 조유근¹, 홍지만²

¹서울대학교 컴퓨터공학부, ²숭실대학교 컴퓨터 공학부

¹{hmin, shyi, bjkim, shkim, ykcho}@os.snu.ac.kr ²jiman@ssu.ac.kr

Remote module broker for heterogeneous wireless sensor networks

Hong Min⁰¹, Sangho Yi¹, Bongjae Kim¹, Seokhyun Kim¹, Yookun Cho¹, Jiman Hong²

¹School of Computer Sci. & Eng., Seoul National University,

²School of computing, Soongsil University

요 약

기존 무선 센서 네트워크 분야에서의 연구는 단일기종 환경이라는 가정하에서 진행되었다. 최근 정보 수집 기법이나 라우팅 프로토콜의 경우 클러스터링 기반의 이기종 환경에서 더 높은 에너지 효율을 보이고 있다. 네트워크의 수명이 길어지고, 이를 기반으로 동작하는 응용에 대한 요구가 다양해 짐에 따라 동적 재구성에 대한 요구도 높아지고 있다. 본 논문에서는 이기종 환경을 기반으로 보다 효율적인 동적 재구성 기법을 제안한다. 제안한 기법은 무선 센서 네트워크의 특성을 반영하며, 대규모의 네트워크를 지원할 수 있도록 설계했다. 시뮬레이션 결과를 통해, 제안한 기법이 기존의 기법에 비해 우수하다는 것을 검증한다.

본 논문에서는 이기종 무선 센서네트워크 기반의 동적 재구성 기법을 제안한다. 기존의 센서 운영체제에서 제공하는 기법들은 동일한 플랫폼을 가정하고 있기 때문에, 일부분을 수정할 때에도 네트워크 내의 모든 노드들의 실행환경을 새롭게 갱신해주어야 하는 문제가 발생한다. 새로 제안된 기법은 이기종 플랫폼의 계층관계를 활용하여, 일부의 변화가 네트워크 전체에 영향을 주지 않도록 제한한다. 시뮬레이션을 통한 성능 평가를 통해 기존 기법들 보다 에너지 효율성 측면에서 우수하다는 것을 보인다.

본 논문의 구성은 다음과 같다. 본 연구의 필요성과 연구 범위를 제시한 1장의 서론에 이어, 2장에서는 동적 재구성과 관련된 기존 연구들을 살펴본다. 3장에서는 기존 시스템의 문제점은 지적하고, 4장과 5장에서는 본 논문에서 제안하는 동적 재구성 기법의 요구 사항과 설계에 관련된 내용을 서술한다. 6장에서는 성능 평가를 통해 본 논문에서 제안한 시스템이 기존의 시스템에 비해 효율적이라는 것을 검증하고, 7장에서 결론 및 향후 연구 방향을 제시한다.

1. 서 론

최근 유비쿼터스 컴퓨팅에 대한 연구가 활발하게 진행됨에 따라, 무선 센서 네트워크가 자연 환경과 가장 밀접하게 상호작용하면서 정보를 수집하는 인프라의 역할을 수행할 것으로 기대된다[1]. 또한 무선 센서 네트워크의 연구 분야도 다양해 지면서, 이론적 연구를 위한 테스트 베드에서 벗어나 실제 생활에서 사용될 수 있는 다양한 응용들이 제안되고 있다[2]. 대규모 네트워크를 제한된 예산 내에서 구성하기 위해, 센서 노드들은 저사양의 프로세서, 제한된 크기의 메모리와 같은 자원의 제약을 받는다. 따라서 효율적인 시스템 설계와 운영이 주된 연구목표이다.

기존의 무선 센서 네트워크 연구들은 동일한 성능과 조건을 갖는 노드들을 기반으로 연구가 진행되었다. 네트워크의 제어와 운영은 관리자가 직접 관리할 수 있는 BS(Base Station)에게 맡기고, 같은 플랫폼과 소프트웨어를 탑재한 수 많은 노드가 자율적으로 네트워크를 구성하도록 설계되었다[3]. 그러나 에너지 효율적인 정보 수집 기법과 라우팅 알고리즘에 대한 연구가 진행되면서, 클러스터링 기반의 계층적 구조가 센서 네트워크의 대규모성을 지원하기에 적합하다는 주장이 설득력을 얻고 있다. 최근에는 단일기종을 가정하는 것보다는 무선 센서 네트워크를 이기종 플랫폼으로 구성하는 것이 보다 현실적이고, 효율적이라는 주장들이 제기되고 있다[4].

2. 관련연구

디버깅이나 응용 변경 등의 목적을 위해 실제 환경에 배치되어 운영되고 있는 센서 노드에, 동적으로 소프트웨어를 업데이트 하는 연구가 많이 진행되었다[5]. 대부분의 연구들은 동일 플랫폼을 가정하고 있으며, 전송해야 할 코드 크기의 최소화를 목표로 하고 있다.

2.1. Deluge

Deluge는 TinyOS[11]에서 제공하는 원격 재구성 기

*본 연구는 서울대학교 BK21과 컴퓨터 연구소의 지원으로 이루어 졌음.

법으로 네트워크를 구성하고 있는 노드들 사이에서 무선 통신을 통해 프로그램 코드를 전송하고 이를 수행할 수 있도록 도와준다[6]. 커널 코드와 응용 프로그램을 포함한 전체 이미지를 전송하기 때문에 통신 비용이 크고, 재구성 과정이 완료될 때까지 네트워크 동작이 중지된다는 단점이 있다.

2.2. FlexCup

FlexCup은 TinyOS에서 컴포넌트 단위로 코드 갱신을 할 수 있도록 도와 주는 기법으로, 컴포넌트의 메타 데이터와 실제 코드를 분리하여 관리하는 것이 특징이다 [7]. 수행 시간에 새로운 컴포넌트를 링크해주는 알고리즘을 제공하기 때문에 하드웨어의 재부팅 없이 재구성 작업이 완료된다.

2.3. 점증적 코드 갱신 기법

점증적 코드 갱신 기법(Incremental Code Updating)은 프로그램 이미지를 블록단위로 나누고, 각 블록의 체크섬값을 해시 테이블에 등록한다[8]. 새로운 프로그램 이미지로 갱신할 경우, 이전에 설정해놓은 해시 테이블을 참조하여 일치하는 블록은 저장해 놓은 이전 프로그램에서 복사하고, 불일치하는 부분만 무선 통신을 통해 다운 받음으로써 전송해야 할 코드의 양을 줄인다. 그림 1은 점증적 코드 갱신 기법의 동작 방식을 보여주고 있다.

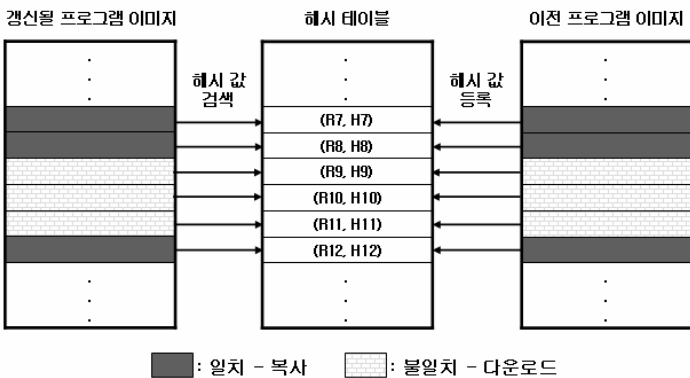


그림 1. 점증적 코드 갱신 기법

2.4. 가상 머신

무선 센서 네트워크에서도 JVM과 같이 가상 머신 기반의 동적 재구성 기법이 연구되고 있다[9][10]. 모든 센서 노드에 가상 머신을 탑재하고, 명령어가 포함된 패킷을 네트워크를 통해 전송하면, 가상 머신이 명령어를 해석하면서 수행시키는 방식으로 동작한다. 기존 응용의 수행을 방해하지 않고, 새로운 프로그램을 수행할 수 있다는 장점이 있지만, 가상 머신이 수행할 수 있는 명령어로 프로그램 로직을 재구성해야 하는 부담이 발생한다.

3. 기존 동적 재구성 기법의 문제점

기존의 동적 재구성 기법들은 통신 비용을 줄이기 위하여 다양한 접근 방법을 제시하고 있지만 동일 플랫폼을 가정하고 있기 때문에, 그림 2와 같은 방법으로 프로그램 코드의 전송이 이루어진다. 클러스터링에 기반한 계층적 구조의 라우팅 프로토콜을 사용하여 코드 전송에 대한 부하를 줄일 수 있지만, 특정 노드에 대한 부담이 늘기 때문에 이로 인한 네트워크 분리(partition) 및 지연 시간, 재전송 비용 증가와 같은 문제가 발생하게 된다. 특히 대규모의 네트워크를 구성하게 되면 동적 재구성으로 인한 지연문제와 자원 낭비가 급속도로 악화된다. 따라서 안정적인 네트워크 관리와 효율적인 운영을 위해 이기종 기반의 네트워크를 구성하고, 이를 활용한 새로운 동적 재구성 기법이 필요하다.

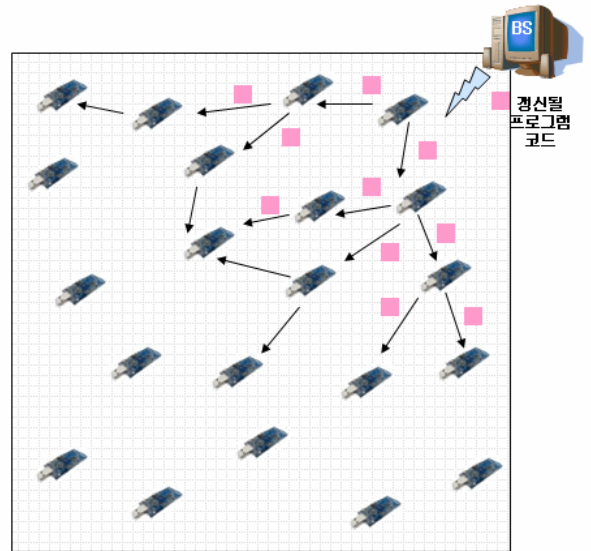


그림 2. 동일 플랫폼 기반의 동적 재구성 기법

4. 시스템 요구 사항

이전 기법들의 문제를 해결하고 효율적인 동적 재구성 기법을 설계하기 위해서 다음과 같은 요구 사항을 만족 시켜야 한다.

- 저비용 네트워크 구성: 대규모 센서 네트워크를 구성하기 위해서 가장 먼저 고려해야 할 부분은 경제적인 측면이다. 센서 노드의 기능이 제한적인 것도 생산비용을 최소화 하기 위해서이다. 따라서 시스템 설계 시에 비용의 문제를 고려해야 한다.
- 다양한 플랫폼 지원: 센서 응용이 다양해짐에 따라, 다양한 형태와 성능을 갖는 플랫폼이 개발되고 있다. 이러한 이기종 간의 호환성 문제도 해결해야 한다.
- 대규모 네트워크 지원: 기존의 기법들의 가장 큰 문제는 대규모 네트워크 적용 시 이를 효과적으로 지원하지 못해 지연 시간이 급격하게 증가한다는 것이다. 새로 제안될 시스템은 대규모 네트워크를

안정적으로 지원할 수 있어야 한다.

- 통신 비용의 최소화: 통신 비용은 연산 작업보다 많은 에너지를 소모한다. 센서 노드의 자원 제약 조건을 만족할 수 있도록 통신 비용을 최소화 해야 한다.
- 적용 가능성: 무선 센서 네트워크의 운영에 변화를 주지 않고 쉽게 적용할 수 있어야 한다.

본 논문에서는 위에서 언급한 요구 사항을 고려하여, 새로운 동적 재구성 기법을 설계하였다.

5. 원격 모듈 중개자의 설계

이기종 네트워크 제어와 응용 프로그램 수행을 효율적으로 하기 위해서, 시스템에서 사용할 다양한 모듈들을 관리해 줄 수 있는 원격 모듈 중개자 개념을 사용한다. 원격 모듈 중개자는 말단 노드들에게 서비스를 제공하고 BS(Base Station)의 명령을 전달하는 인프라를 제공한다.

5.1. 네트워크의 구성

네트워크 관리의 효율성을 위해서 각 노드는 성능에 따라 다음과 같은 역할을 수행한다.

- BS: 네트워크를 제어 및 운영을 담당하는 노드로 관리자가 직접 관리하며, 성능이 뛰어나고 전력 공급이 안정적이다.
- 원격 모듈 중개자: BS로부터 명령을 받고, 말단 노드들로부터 수집된 정보를 BS로 전달한다. 말단 노드들 보다 성능이 우수하고, 자가 발전 장치를 장비하여 전원을 공급 받을 수 있다.
- 말단노드: 저사양의 프로세서와 작은 크기의 메모리, 무선 통신 모듈로 구성되어 있으며, 필요에 따라 센서를 장비하고 있다.

본 논문에서는 계층적 구조를 통해 네트워크를 관리하며, 각각의 역할에 따라 서로 다른 성능 및 요구사항을 가정한다. 특히 원격 모듈 중개자는 다양한 센서 응용을 수용할 수 있을 정도의 성능과 통신 능력을 갖는다.

5.2. 시스템 개요

새로 제안한 시스템은 모듈기반으로 동작하는 SOS[12] 센서 운영체제 위에서 설계되었다. SOS는 미국의 UCLA에서 개발한 센서 운영체제로 TinyOS의 컴포넌트와 유사한 모듈단위로 시스템의 각 부분을 나누고 이들간의 상호 작용을 통해 응용 작업을 수행한다.

원격 모듈 중개자를 사용하여 구현한 시스템의 특징은 말단노드들은 네트워크의 변경이나 데이터 처리를 위한 연산 작업에 관여하지 않는다. 단지 측정된 정보를 원격 모듈 중개자에게 전달하고, 정보 처리를 위한 모듈을 요청하면 된다. 원격 모듈 중개자는 말단 노드들로부터의 요청을 메시지 큐에 저장해 두고, 하나씩 처리하여

필요한 경우에는 결과 값을 말단 노드에게 전달한다. 그림 3은 원격 모듈 중개자와 말단 노드 사이의 모듈 요청과 처리 과정을 보여주고 있다.

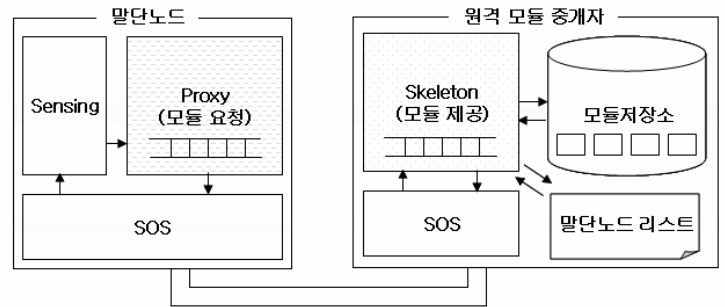


그림 3. 시스템 개요

모듈 요청과 제공을 위해서 이기종간의 통합응용개발을 지원해주는 CORBA[13]의 Proxy-Skeleton 방식을 무선 센서 네트워크에 맞게 적용했다. 말단노드는 측정된 데이터 값을 처리하거나 BS에게 전송할 필요가 있을 경우 Proxy를 통해 원격모듈을 호출한다. Proxy는 원격 모듈 중개자가 모듈 실행에 필요한 정보를 인터페이스에 맞게 배치해서 버퍼에 저장해두고, SOS의 통신 모듈을 통해 Skeleton에게 처리를 요청한다. 원격 모듈 중개자는 말단노드 리스트를 통해 자신이 관리하는 노드인지를 확인하고, 모듈저장소로부터 데이터 처리에 필요한 모듈을 제공받아 작업을 처리한다. 센서 응용의 대부분이 주변 환경을 측정하고, 이를 BS에 전달하는 방식이 많기 때문에 원격 모듈 중개자에서 처리된 내용을 말단노드로 전송하는 경우는 자주 발생하지 않는다.

5.3. 동적 재구성 과정

제안한 시스템에서 동적 재구성은 다음과 같은 방식으로 이루어진다.

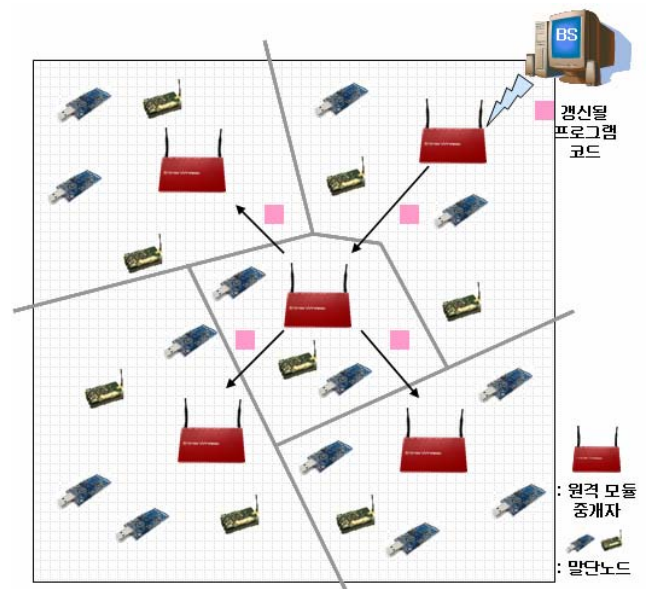


그림 4. 원격 모듈 중개자를 통한 동적 재구성

그림 4에서 보는 바와 같이 동적 재구성 과정은 원격 모듈 중개자만 참여한다. BS는 새로 갱신할 모듈 코드를 원격 모듈 중개자에게 전달하고, 원격 모듈 중개자는 주변의 원격 모듈 중개자 노드에게 BS로부터 받은 패킷을 전달한다. 전송이 완료되면 모듈저장소에 새로운 모듈을 등록한다. 말단노드는 원격 모듈 중개자가 모든 작업을 처리하기 때문에 모듈 갱신에 대한 추가적인 작업이 필요 없어 기존의 작업이 방해 받지 않는다.

6. 성능 평가

본 논문에서 제안한 기법의 성능측정을 위해서 무선 센서 네트워크용 시뮬레이터인 센서메이커[14]를 사용하였다. 시뮬레이션을 위해 아래와 같이 원격 모듈 중개자와 말단노드를 각각 설정하였다.

표 1. 시뮬레이션 환경

	원격 모듈 중개자	말단노드
처리속도	533 MHz	8 MHz
RAM	32 MB	4 KB
FLASH	512 MB	128 KB
통신거리(가능)	50 m	15 m
통신거리(안정)	30 m	10 m
에너지 보유량	2000 J이상	187.5 J

원격 모듈 중개자 15개 말단 노드 200개를 1000m X 1000m 크기의 실험 지역에 배치했다고 가정한다. 먼저 동적 재구성 비용의 측정을 위해서 갱신할 코드의 크기를 50B에서 1KB까지 변화시켜 가면서 에너지 사용량을 측정하였다.

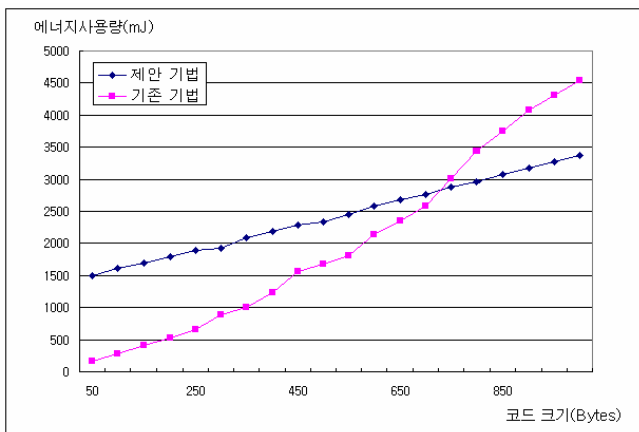


그림 5. 에너지 사용량

그림 5는 기존 기법과 논문에서 제안하고 있는 기법 사이의 에너지 사용량을 비교한 것이다. 기존 기법의 경우 전송 코드의 크기가 커짐에 따라 급격하게 에너지 사용량이 증가하는 것을 볼 수 있다. 이는 동적 재구성

에 참여하는 노드들의 숫자가 많고, 간섭으로 인한 재전송 비용이 급격하게 증가하기 때문이다. 그림 6은 제안 기법의 에너지 사용량을 원격 중개자와 말단노드를 각각 분리하여 보여주고 있다.

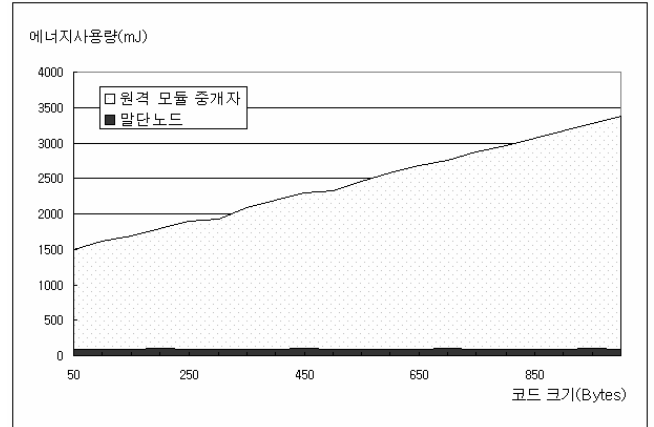


그림 6. 노드별 에너지 사용량 (제안기법)

그림에서 보는 바와 같이 말단 노드는 동적 재구성에 참여하지 않기 때문에 추가적인 에너지 소모가 없다. 이는 네트워크의 수명을 늘리는데 중요한 역할을 한다.

다음은 제안 기법과 기존 기법의 동적 재구성 완료 시간을 비교한 그래프이다.

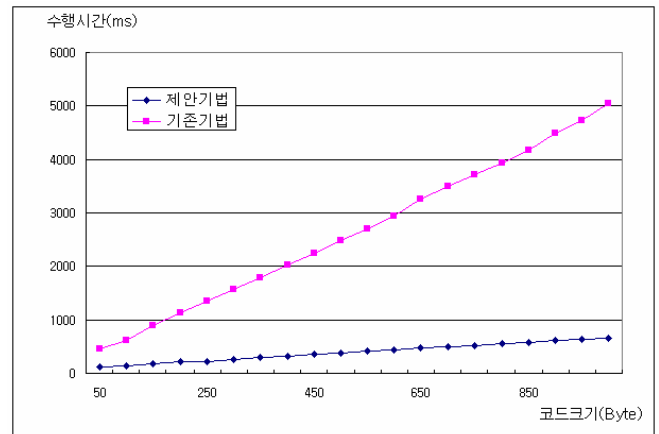


그림 7. 수행시간

말단노드와 원격 모듈 중개자 사이의 성능의 차이가 큰 차이의 원인이 될 수 있지만, 원격 모듈 중개자 사이의 코드 전송으로 동적 재구성이 이루어지는 시스템 구조의 특성도 다른 원인이 될 수 있다. 이를 통해 제안한 기법이 기존의 기법에 비해 대규모 네트워크에 더 적합하다는 것을 알 수 있다.

7. 결론

무선 센서 네트워크의 플랫폼 및 응용에 대한 요구가 다양해짐에 따라, 이기종 네트워크에 대한 연구의 필요성이 증대되고 있다. 특히 아직 수명이 남은 네트워크를

활용하여 새로운 기능을 할 수 있도록 하는 동적 재구성 기법의 경우, 다양한 플랫폼에 대한 고려가 필요하다.

본 논문에서는 단일기종만을 가정하여 동적 재구성 기법을 제안한 시스템의 문제점을 분석하고, 이기종 환경을 통해 이를 해결하는 기법을 제안했다. 새로운 기법은 무선 센서 네트워크의 저비용, 자원 제약의 특성을 이해하면서, 대규모의 네트워크를 에너지 효율적인 방법으로 지원할 수 있도록 설계했다. 시뮬레이션 결과를 통해, 새로 제안한 기법이 기존의 시스템에 비해 대규모 네트워크에서 에너지 사용량과 수행 시간 측면에서 우수하다는 것을 검증했다. 지금까지는 SOS라는 플랫폼에 의존하는 부분이 있기 때문에, 향후에는 이러한 제한을 받지 않도록 통합된 플랫폼을 제공할 수 있는 시스템의 설계를 연구할 계획이다.

참고문헌

- [1] W. Mingfei, C. Linlin, and Z. Ping, "Design Issues of Wireless Sensor Networks in Ubiquitous Learning", International Conference on Machine Learning and Cybernetics, vol. 7, p.4149-4153, 2007.
- [2] D.Cuff, M. Hansen, and J. Kang, "Urban sensing: out of the woods", Communications of the ACM, vol. 51, p.24-33, 2008.
- [3] Y. Yu, B. Krishnamachari, and V. Prasanna, "Issues in designing middleware for wireless sensor networks", IEEE Network, vol. 18, p.15-21, 2004.
- [4] X. Du, and F. Lin, "Improving routing in sensor networks with heterogeneous sensor nodes", IEEE the 61st Vehicular Technology Conference, vol. 4, p. 2528- 2532, 2005.
- [5] S. Brown, "Updating Software in Wireless Sensor Networks: A Survey", National University of Ireland, Technical Report, 2006.
- [6] J. Hui, and D. Culler, "The Dynamic Behavior of a Data Dissemination Protocol for Network Reprogramming at Scale", ACM the 2nd International Conference on Embedded Networked Sensor Systems, p.81-94, 2004.
- [7] P. Marron, M. Gauger, and D. Minder, "FlexCup: A Flexible and Efficient Code Update Mechanism for Sensor Networks", The 3rd European Workshop on Wireless Sensor Networks, p. 212-227, 2006.
- [8] J. Jeong, and D. Culler, "Incremental Network Programming for Wireless Sensors", IEEE Sensor and Ad Hoc Communications and Networks, p.25-33, 2004.
- [9] P. Levis, and D. Culler, "Mate: a tiny virtual machine for sensor networks", The 10th international conference on Architectural support for programming languages and operating systems, p.85-95, 2002.
- [10] S. Michiels, W. Horre, W. Joosen, and P. Verbaeten, "DAViM: a dynamically adaptable virtual machine for sensor networks", ACM International workshop on Middleware for sensor networks, p.7-12, 2006.
- [11] TinyOS, <http://www.tinyos.net/>
- [12] SOS, <https://projects.nesl.ucla.edu/public/sos-2x/doc/>
- [13] CORBA, <http://www.corba.org/>
- [14] 이상호, 이승우, 민홍, 조유근, 홍지만, "센서메이커: 세밀하고 확장성 있는 실험을 위한 무선 센서 네트워크 시뮬레이터", 한국정보과학회 정보통신 연구회 하계컴퓨터통신워크샵, 2007.