

차세대 유비쿼터스 환경을 위한 적응적 센서 네트워크 노드 구조

박수용^o 김성수 이상학*

아주대학교 정보통신전문대학원, 전자부품연구원*

{cslab03^o, sskim}@ajou.ac.kr, shlee@keti.re.kr*

An Adaptive Sensor Network Node Structure for Next Generation Ubiquitous Environment

Suyong Park^o, Sungsoo Kim, Sanghak Lee*

Graduate School of Information and Communication, Ajou University

Ubiquitous Computing Research Center, Korea Electronics Technology Institute*

요 약

현재의 센서 네트워크를 구성하고 있는 센서 노드들은 배치 전에 미리 입력된 라우팅 및 감지 서비스만을 제공한다. 이러한 센서 네트워크 및 노드 구조는 기존의 고정화된 환경에는 적합하나 차세대 유비쿼터스 환경과 같이 시스템이 스스로 성장하며 서비스 추가 및 재설정 요구가 빈번하게 일어나는 환경에서는 그대로 적용하기가 힘들다. 따라서 본 논문에서는 유비쿼터스 환경에서 시스템 레벨의 정책 및 환경 변화에 따라 적응적으로 동작할 수 있는 센서 네트워크 노드 구조를 제안하며, 또한 제안된 구조에서 기존의 구조에 대한 에너지 소모 정도를 분석하여 제안의 현실성을 검증한다.

1. 서 론

센서 네트워크는 유비쿼터스 컴퓨팅에서 핵심 기술로 떠오르고 있는 분야로써, 각 노드들은 제한된 범위 내의 이벤트를 감지하고 감지된 정보를 RF(Radio Frequency) 통신 매체를 이용하여 이웃 노드들로 전송한 후 최종적으로 싱크 노드에 앞으로의 판단을 위한 원정보(Raw data) 들을 전달한다. 이러한 센서 네트워크에서의 각 노드는 한정된 하드웨어 자원으로 노드간 통신 기능을 제공해야 할 뿐만 아니라, 여러 이벤트에 반응하여 이를 실시간으로 처리할 수 있어야 한다. 특히 차세대 유비쿼터스 환경에서는 전체 시스템에서 결정된 정책 및 주변 환경 변화에 따라 실행시간 중에 시스템 구성요소들의 동작이 재설정되어야 할 필요가 있으나 기존의 센서 네트워크는 노드 배치 전에 입력된 동작 외에는 서비스 환경 변화에 대해 유연하게 대처하기 어렵다는 단점이 있으며, 이를 위해 노드 배치전에 모든 경우를 고려하여 시스템을 구현하는 것은 하드웨어적인 한계로 인하여 불가능하다.

이러한 문제를 해결하기 위해서 고려할 수 있는 방법으로는 “built-in” 구조의 센서 노드가 아닌 실행시 서비스 요구에 따라 모듈을 재설정할 수 있는 “plug-in” 형태의 센서 노드를 고려해 볼 수 있다[1]. 즉 한정된 하드웨어 환경에서 변화에 대한 적응성을 가지기 위해서는 시

스템 정책에 따른 라우팅 서비스 교체나 기타 추가 서비스 설치가 가능한 구조의 센서 노드가 요구된다. 이를 위해 본 논문에서는 오토노믹 유비쿼터스 환경의 센서 네트워크에 적합한 동작 프레임워크 제공을 위한 노드 구조를 제안한다. 또한 이러한 노드 구조가 시스템 및 서비스 재구성을 어떻게 지원하며 메모리 및 전력 등의 자원 제한을 어떻게 만족시키는지에 대해 설명한다.

2. 관련 연구

IBM이 제안한 오토노믹 컴퓨팅은 인간의 개입을 최소화하면서 스스로 관리가능한 컴퓨팅 시스템(Self-managed System) 개발을 위한 개념으로, 컴포넌트 혹은 하위 시스템 레벨에서 구현된 오토노믹 단위요소(Autonomic Element)의 모임으로 시스템을 구축하고 각 단위요소의 상호작용을 통하여 시스템 동작을 자율적으로 이끌어가게 된다[2]. 이를 센서 네트워크에 적용하기 위해서는 각 센서 노드를 오토노믹 단위요소로 취급할 수 있으며 이를 위해 오토노믹 관리자(M.A.P.E Loop) 개념을 노드 레벨에서 간단하게 구현할 필요가 있다.

또한 인터넷 환경에서 서비스 요구 변화에 적응적으로 동작하기 위한 기존 연구로서 Router-Plugins가 있다[3]. 이 Router-Plugins는 복잡한 서비스 요구를 충족시키기 위한 확장 가능한 라우터 구조를 제안한다. 이러한 확장 가능한

본 연구는 과학기술부 21세기 프론티어 연구 개발 사업의 일환으로 추진되고 있는 유비쿼터스 컴퓨팅 및 네트워크 원천기반 기술 개발 사업의 지원에 의한 것이다. 이 논문은 2004년도 두뇌한국21 사업에 의하여 지원되었음

라우터 구조는 비교적 간단하나 자원의 제약으로 인하여 센서 네트워크에 그대로 적용하기에는 효율적이지 못하며 이를 고려한 개방형 노드 구조가 필요하다. 또한 이러한 센서 네트워크의 제약사항을 극복하기 위해서 Mate 연구와 같이 작은 명령어 집합과 가상 머신을 사용하여 메모리와 전력 소모를 줄이는 방법을 적용할 수 있다[4].

3. 적응적 센서 네트워크 노드 구조

컴포넌트 및 하위 시스템 레벨의 오토노믹 단위요소로 구현된 오토노믹 시스템의 기본적인 개요는 그림 1과 같다[5]. 이러한 구조는 P2P 기반의 에어젯 시스템 혹은 수많은 노드들로 이루어진 센서 네트워크의 토폴로지와 매우 유사하며, 그렇기 때문에 이러한 개념을 실제 노드 구조를 위해 그대로 적용하는 것이 가능하다. 물론 서비스 추가 및 재설정이 가능하고 적응적으로 동작할 수 있는 노드 구조는 현재의 센서 네트워크 환경하에서는 크게 요구되지 않으나 차세대 유비쿼터스 컴퓨팅 환경하에서는 사용자의 서비스 요구에 능동적으로 반응하기 위해 유비쿼터스 시스템 하부의 센서 노드들이 특정 동작을 수행할 필요가 있으며, 이 때 제안된 구조가 기존 구조보다 더 적은 수의 센서 노드로 더 나은 성능을 보일 수 있다.

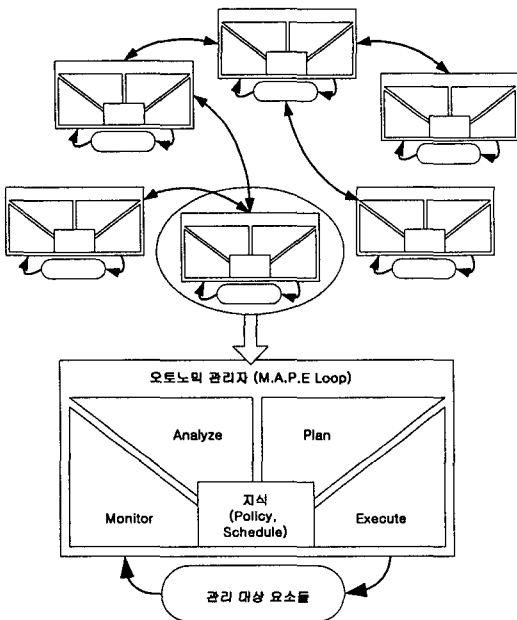


그림 1 오토노믹 시스템의 구조 개요

이러한 시스템 구조를 센서 네트워크에 적용할 때 오토노믹 컴퓨팅 개념을 위한 오토노믹 관리자(M.A.P.E. Loop)의 삽입으로 인한 노드 내부 구조의 변화는 대략 그림 2와 같다. 이러한 변화는 관리 대상과 관리 부분의 확실한 분화를 위한 것으로, 정책 및 환경 변화에 따라 관리 대상이 관리 부분에 의해 적응적으로 재설정된다.

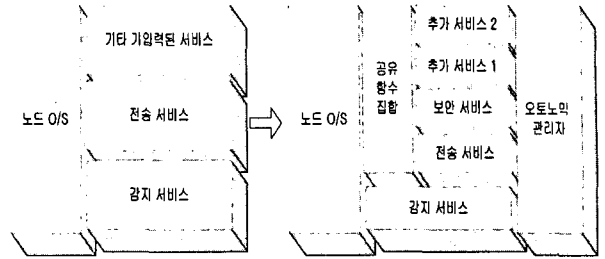


그림 2 변화된 센서 노드의 구조 개요

이렇게 노드 구조를 변경할 때 서비스 재설정 관련 부분들이 포함되기 때문에 이에 대한 추가적인 코드나 실행의 전력 소모가 문제가 될 수 있다. 이를 최소화하기 위한 방법으로 각 부분들에서 사용되는 공통 함수를 묶어 하나의 공통 라이브러리 형태로 만들어 서비스 재설정 시 이웃 노드 및 싱크 노드로부터 전송받는 코드의 양을 줄이는 행위가 필수적이다. 또한 이러한 라이브러리 코드들은 각 서비스가 시작되기 전에는 메모리에 적재되지 않으므로 실제 동작하는 서비스 코드들도 감소시킬 수 있다. 또한 전체 시스템에서 결정된 정책에 따라 센서 노드의 유휴-휴면-활성(Idle-Sleep-Active) 상태 전이를 동적으로 수행하여 시스템이 필요로 할 때에만 동작함으로써 불필요한 데이터의 전송과 전력소모를 줄여 센서 노드의 수명을 증가시킬 수 있다.

4. 성능 평가

실제 센서 노드의 대략적인 성능 모델은 아래 수식과 같으며 이 수식에서 보듯 실제 노드의 성능에 영향을 주는 요인은 크게 노드의 기종별 특성, 처리 데이터의 크기, 버스 및 프로세서의 주파수 등을 들 수 있다[6].

$$\text{전력소모량} \approx \text{실행시간} \approx \frac{a+b \cdot [\text{데이터량/블록 크기}]}{\text{프로세서 주파수} \cdot \text{버스 크기}}$$

(a, b: 각 센서 노드의 기종 및 급행 연산능력 등에 의한 보정값)

$$a = a_{\text{BASE}} + a_{\text{Multiplication}} + a_{\text{RISC}}$$

$$b = b_{\text{BASE}} + b_{\text{Multiplication}} + b_{\text{RISC}}$$

(a_{BASE} , b_{BASE} 는 각 센서 노드별 서비스에 대한 기본 파라미터, $a_{\text{Multiplication}}$, $b_{\text{Multiplication}}$ 는 대상 서비스의 급행 연산 유무에 따른 보정값, a_{RISC} , b_{RISC} 는 각 센서 노드의 RISC/CISC 구조에 따른 보정값을 의미)

이에 기반하여 성능 비교는 노드의 실제 서비스 실행 시간과 전력 소모가 정비례한다는 가정하에서 수행되었으며 이 때 파라미터들은 표 1과 같다. 또한 성능 평가를 위한 대상 센서 노드의 하드웨어 플랫폼은 UltraSparc2 440MHz로 가정하였으며 사용자 요구에 의한 서비스 재설정 및 감지 이벤트 발생은 균일 분포(uniform distribution)를 따르며 매우 빈번하게 발생한다고 가정하였다. 추가적으로 감지 이벤트는 각 경우의 비교를 위해 동일 시간대에 일어난다. 본 성능 평가에서 사용한 서비스들은 모두 암호화 알고리즘으로서 대상 센서 노드에서의 실행 시간과 소모 전력은 기존 연구에서 검증된 값들을 이용하였다.

표 1 성능평가를 위한 추가 서비스 및 파라미터들

서비스	보정값 a	보정값 b	블록 크기
MD5	203656	86298	512
SHA1	77337	233082	512
RC4	69240	13743	8

성능 평가를 위해 기존 구조에 서비스가 추가된 경우를 함께 테스트하였으며 이는 노드 배치시 미리 서비스를 추가하는 경우와 동적으로 서비스를 추가하는 경우에 소모되는 추가적인 비용을 비교하기 위한 것으로, 본 시뮬레이션에서 적응적 노드 구조가 기존 전송 및 감지 서비스 외에 3가지의 추가적인 서비스를 지원할 수 있다고 가정하였으므로 최대 3개를 기존 노드에 미리 배치한 경우까지 가정하였다.

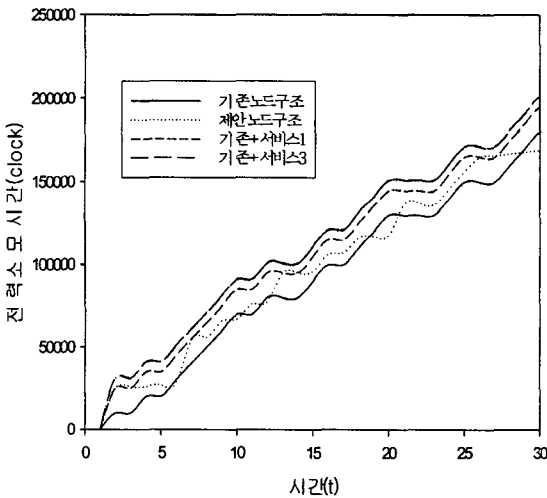


그림 3 제안된 노드 구조 및 기존 구조 성능 비교

표 2 시간당 각 구조의 평균 전력 소모 (clock/t)

구조 \ 시간	0~10	10~20	20~30
기존구조	31000	96000	149000
제안구조	37551.37	100383	154610.6
기존+서비스1	44440.54	110933.9	163933.9
기존+서비스3	50097.04	117218.9	170218.9

그림 3의 결과 그래프를 보면 노드가 시간이 흘러갈 때 따라 동작을 위해 소모한 평균적인 전력이 가장 적은 구조는 기존의 단순한 노드 구조이며, 본 논문에서 제안한 적응적 노드 구조, 기존 노드 구조에 추가적인 서비스가 1개 증가한 경우, 기존 노드 구조에 추가적인 서비스가 3개 증가한 경우 순으로 이루어진다. 또한 표 2의 결과에서 볼 수 있듯이 적응적 노드 구조는 기존의 단순한 노드 구조에

비해 전체적으로 전력소모가 증가하였다. 그러나 이런 추가 전력소모가 비교적 작다고 할 수 있으며, 또한 이는 기존 노드가 더 적은 수의 제한된 서비스만을 제공하기 때문으로, 기존 노드 구조에서 추가적인 서비스가 요구될 경우에는 그 래프에서 보듯이 제안된 노드 구조가 기존 구조에 서비스를 추가하는 경우에 비해 전력 소모 및 실행시간을 줄일 수 있다. 현실적으로 예상되는 모든 서비스를 노드 배치전에 설치한다는 것은 전력 소모 및 메모리 제약으로 인해 불가능하므로 제안된 적응적인 노드 구조가 실질적으로 더 나은 성능을 보여줄 수 있다.

5. 결론

본 논문에서는 오토노믹 유비쿼터스 환경하에서 전체 시스템 레벨의 정책 및 환경 변화에 따라 적응적으로 동작할 수 있는 센서 네트워크 노드의 구조를 제안하였으며, 또한 제안된 적응적 센서 네트워크 노드 구조와 기존 노드 구조의 에너지 소모를 비교하여 제안된 구조의 현실성을 검증하였다. 향후 연구에서 제안된 노드 구조를 실제 어플리케이션에서 적용하기 위해서는 전체 센서 네트워크에서 전송 서비스 재설정시 라우팅 기법을 노드 레벨에서 동조화하는 방법, 서비스 재설정에 따른 에너지 소모 및 오류, 서비스 재설정을 위한 결정 기준을 위한 정책의 정의, 재설정시 네트워크 확장성에 대한 고려 등이 이루어져야 하며 이는 차세대의 정책 기반 유비쿼터스 컴퓨팅 시스템 구현을 위해서 매우 중요한 역할을 수행할 것이다.

6. 참고 문헌

- [1] Y. He, et. al., " A Programmable Routing Framework for Autonomic Sensor Networks," Proceedings of IEEE International Autonomic Computing Workshop on Active Middleware Services, pp. 60-68, 2003.
- [2] IBM Autonomic Computing Research, <http://www.research.ibm.com/autonomic/>
- [3] D. Decasper, et. al., " Router plugins: A Software Architecture for Next Generation Routers," IEEE/ACM Transactions on Networking, Vol. 8, No. 1, pp. 2-15, 2000.
- [4] P. Levis and D. Culler, " Maté: A Tiny Virtual Machine for Sensor Networks," Proceedings of International Conference on Architectural Support for Programming Languages and Operating Systems, pp. 85-95, 2002.
- [5] J. Kephart, et. al., " The Vision of Autonomic Computing," IEEE Computer Magazine, Vol.36, No. 1, pp. 41-50, 2003.
- [6] P. Ganesan, et. al., " Analyzing and Modeling Encryption Overhead for Sensor Network Nodes," Proceedings of ACM Workshop on Wireless Sensor Networks and Applications, pp. 151-159, 2003.