

커뮤니티 컴퓨팅 환경에서의 멤버 생존시간 최적화 알고리즘 연구

論 文

57-7-23

Study on the Optimization Algorithm for Member Lifetime in Community Computing Environments

金基英** · 朴慧誠* · 盧暻佑* · 金錫潤†

(Ki-Young Kim · Hyae-Seong Park · Kyung-Woo Noh · Seok-Yoon Kim)

Abstract - In community computing environments, various members cooperate with each other systematically for attaining each community's goals. Because community computing environments are organized on the basis of PAN (Personal Area Network), each member commonly uses the power of batteries. If one member in community uses up the power of battery and does not operate normally, the community will not be able to provide the ultimate service goals for its users and be terminated finally. Therefore, it is necessary for accurate community operation to prevent a specific member's lifetime from terminating, as checking each member's power consumption in real-time. In this paper, we propose WEL (WEighted Leach) algorithm for optimizing lifetime of the members in community.

Key Words : Community Computing, Lifetime

1. 서 론

현재의 컴퓨팅 환경을 대표하는 단어로는 '유비쿼터스'가 있다. 사용자의 제어 없이도 전자 기기들 간의 통신을 수행하고 여러 가지 다양한 서비스를 제공하는 것을 목표로 삼고 있는 환경이다. 유비쿼터스 컴퓨팅 환경 상의 기기들은 모두 제각각의 특성들을 가질 수 있기 때문에, 서비스를 제공하는 과정 중 서로 간에 데이터를 주고받거나 하는 등의 커뮤니케이션 동작에 주의를 기울여야 한다.

이와 같은 유비쿼터스 컴퓨팅 환경 내에서, 다양한 컴퓨팅 장치 (Computing Device), 응용 서비스 (Application Service) 등이 어떤 공통의 서비스 목표를 달성하기 위해서 '커뮤니티 (Community)'라는 개념으로 결속되고 이들 간의 협력을 수행하게 된다. 따라서, 처음부터 새롭게 설계해서 개발하는 과정을 수행하지 않고도 기존의 서비스나 기기들을 동적으로 결합시킴으로써 어떤 새로운 서비스 혹은 발전된 형태의 서비스를 개발하는 것이 가능해질 수 있다. 이것이 바로 '커뮤니티 컴퓨팅'의 개념이다 (그림 1).

위에서 언급한 바와 같이, 커뮤니티 컴퓨팅 환경은 각 커뮤니티를 형성한 목표를 달성하기 위해 복수 개의 컴퓨팅 장치 (이후 '멤버'라는 용어로 사용함)들이 유기적으로 협력하게 된다. 현재 출시되어 있는 주요 휴대형 멀티미디어 기기들은 차세대 PC로서 정보기기 기능을 가진 PDA, 휴대형

오디오 플레이어인 MP3플레이어, 동영상 재생 및 네비게이션 등 멀티미디어 기능을 가진 PMP 등이 있다. 시간이 지날수록 대부분의 기기들은 원래 가진 기능 이외에도 인터넷을 비롯한 다양한 부가기능들을 그 안에 담게 될 것이고, 멀티 기능을 통합적으로 제공하는 장비인 PMP를 중심으로 통합될 것으로 전망된다. 그리고 이러한 PMP에 가정 또는 교육기관 내의 네트워크 환경에서 서비스를 제공하는 기능적 제공자 역할이 바로 개인형 미디어 서버 및 게이트웨이 기술이라 할 수 있다. 이와 같은 역할을 담당하는 차세대 개인형 미디어 게이트웨이를 Pway (Personal gateway)라고 정의하고, 본 연구에서는 커뮤니티 컴퓨팅 환경을 운영하는 중앙 관리의 역할을 Pway가 맡는 것으로 가정한다.

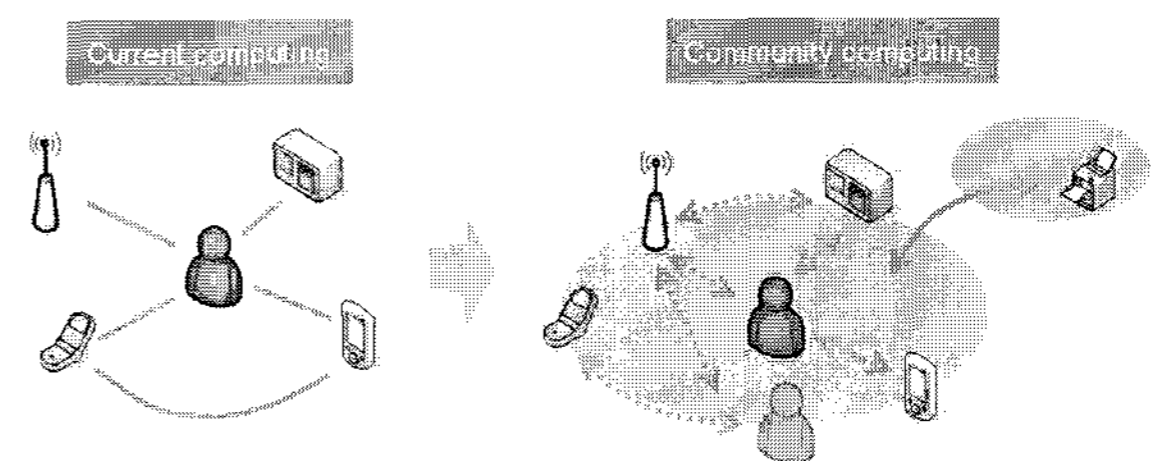


그림 1. 커뮤니티 컴퓨팅의 개념[1]

Fig. 1 The concept of community computing[1]

Pway는 Zigbee/WLAN/UWB 인터페이스를 모두 지원하며 주변에 연결된 멤버들 간의 자원 공유 및 데이터 포맷을 변환한 미디어를 전송해 주는 역할을 수행한다. 그림 2는 Pway를 이용한 커뮤니티 컴퓨팅 환경의 구성 예를 보여주고 있다. Pway는 다양한 영역의 데이터 전송 속도 및 소모 전력 요구사항을 충족할 수 있도록 Zigbee, UWB 및 WLAN인터페이스를 제공한다. Zigbee는 저속, 저전력 단말

† 교신저자, 正會員 : 崇實大學 컴퓨터學科 教授 · 工博

E-mail : ksy@ssu.ac.kr

* 學生會員 : 崇實大學 컴퓨터學科 碩士課程

** 正會員 : 崇實大學 컴퓨터學科 博士課程

接受日字 : 2008年 4月 4日

最終完了 : 2008年 5月 28日

에 적합하며, UWB는 고속 고사양의 단말에 적합한 w-PAN인터페이스이다. WLAN은 현재 상용화된 무선 네트워크 기술 중 가장 보편화된 기술로서 구조가 복잡하고 전력소모가 많다는 단점이 있으나 기존 어플리케이션과의 호환성을 고려하여 w-PAN인터페이스에 포함시켰다.

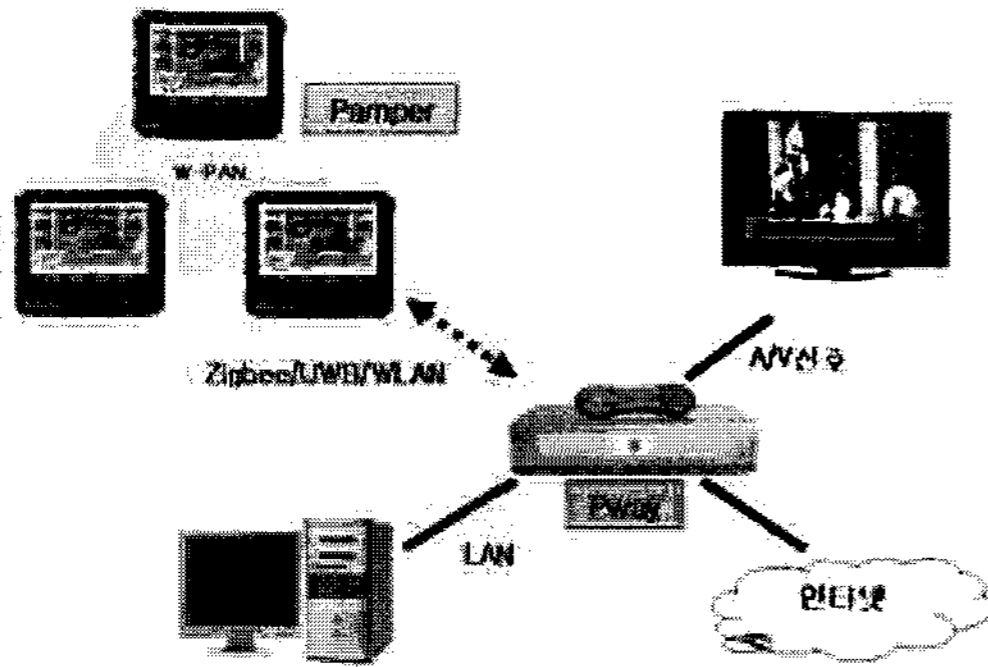


그림 2. 커뮤니티 컴퓨팅 환경의 구성 예
Fig. 2 An example of community computing environments

커뮤니티 컴퓨팅 환경은 기본적으로 PAN (Personal Area Network)을 기반으로 구성되기 때문에, 각 멤버들은 이동성을 보장받기 위해서 배터리를 통하여 전력을 공급받게 된다. 만약, 커뮤니티를 구성하는 어느 하나의 멤버가 배터리를 전부 소모하여 정상적인 동작을 하지 못하게 된다면, 커뮤니티가 구성된 궁극적인 목표 서비스를 사용자에게 제공하지 못한 채 커뮤니티 동작이 종료될 것이다. 따라서, 멤버들의 전력 소모 상황을 실시간으로 확인하여 특정 멤버의 생존시간이 종료되지 않도록 하는 것은, 커뮤니티의 원활한 동작을 위해 필수적인 작업이다.

그림 2와 같이 정의된 환경에서 커뮤니티 컴퓨팅을 수행할 때, 커뮤니티를 구성하는 여러 가지 형태의 멤버들의 생존시간을 최적화시키기 위한 알고리즘을 제안하는 것이 본 연구의 목표이다.

본 논문에서는, 커뮤니티를 구성하는 멤버들의 생존시간 (lifetime)을 최적화시키기 위하여 다음 2가지 방안을 커뮤니티 컴퓨팅 환경에 적용하는 것을 제안한다.

첫째, 커뮤니티내의 멤버들을 몇 개의 클러스터로 분리하며, 각 Sender 및 Receiver 헤더를 가지는 독립적인 클러스터를 통해 데이터 전송을 수행하는 것이다. 센서 네트워크 환경에 대한 기존 연구[2]를 확장한 것으로, 모든 멤버가 비교적 긴 거리까지 데이터를 전송하기 위해 전력을 소모하는 것보다는 특정 멤버인 Sender와 Receiver를 중계점으로 사용하여 데이터를 전송하는 것이 효율적이라는 가정에서 시작한다. 기존의 연구에서는 클러스터 헤드를 하나만 선정하여 데이터 전송을 수행함으로써 인해 전송 효율이 떨어질 수 있기 때문에, 본 연구에서는 데이터를 송신하는 헤드와 수신하는 헤드를 분리하는 방법을 사용하기로 한다. 따라서, 동적으로 분리된 각 클러스터 내의 멤버들이 데이터를 송신하고 싶을 때는, 선출된 각 Sender들에게 데이터를 보내고 그 Sender는 데이터를 취합하여 다른 곳 (다른 클러스터의 Receiver, 중앙 서버, 커뮤니티 통신을 관리하는 게이트웨이 등)으로 전송하게 된다. 데이터를 수신할 때도 마찬가지로 다른 곳 (다른 클러스터의 Sender, 중앙 서버, 커뮤니티 통

신을 관리하는 게이트웨이 등)으로부터의 데이터를 각 클러스터의 Receiver가 취합하여 자신이 속한 클러스터의 멤버들에게 전송하게 된다.

둘째, 각 클러스터 내의 특정 Sender와 Receiver만이 계속적으로 클러스터 외부로의 데이터 전송을 전담하게 되면, 해당 Sender와 Receiver의 전력 소모는 급증하여 더 이상 송신 헤드나 수신 헤드로서의 역할을 수행하지 못하게 될 수도 있을 것이다. 따라서, 커뮤니티 내 모든 멤버들이 공평하게 송,수신 헤드로 선정될 수 있도록 해주는 것이 커뮤니티 전체의 생존시간을 관리하는 데 있어 중요하다. 이것을 위해 기존의 클러스터 헤드 선정 알고리즘인 LEACH 알고리즘[3]과 같은 것을 적용할 수도 있을 것이다. 하지만, LEACH 알고리즘은 단지 확률적으로 모든 멤버에게 동일한 기회만을 부여해주기 때문에, 초기 배터리량이 적은 멤버가 헤드로 선정된 경우에는 다른 멤버들에 비해 생존시간이 급격히 감소할 수 있는 단점을 지니고 있다. 센서 네트워크 분야에서도 이 문제를 해결하기 위한 다양한 알고리즘 [4]-[12]이 제안되었으나, 본 연구에서는 LEACH 알고리즘을 커뮤니티 컴퓨팅 환경에 좀 더 적합하도록 확장시킨 새로운 알고리즘 (WEighted Leach, WEL)을 제안한다.

본 논문은 다음과 같이 구성되었다. 서론에 이어 2장에서는 제안하는 멤버 생존시간 최적화 알고리즘을 적용시킬 커뮤니티 컴퓨팅 환경을 정의하고, 3장에서는 제안하는 알고리즘에 대해 구체적으로 설명하며, 4장에서 결론을 맺도록 한다.

2. 멤버 생존시간 최적화 알고리즘 (WEL)

2.1 배경

제안하는 알고리즘은, 다수의 멤버들이 특정 서비스를 제공하기 위해 동적으로 커뮤니티를 생성한 이후의 상황을 가정한다.

커뮤니티가 형성되면, 커뮤니티내에서 WEL 알고리즘이 동작을 시작한다. WEL 알고리즘은 '라운드' 단위로 동작하게 되는데, 하나의 라운드는 임의의 멤버들이 클러스터를 구성하는 부분과 각 클러스터의 헤드를 통해 데이터를 전송하는 부분으로 구성된다.

2.2 각 Sender와 Receiver 헤드 결정

멤버들이 클러스터를 구성하기 위해서, 일단 클러스터 헤드를 결정해야 한다. 클러스터 헤드를 결정하기 위해서 확률식[3]을 응용한다. 본 논문에서는 데이터를 보내기 위한 헤드인 Sender와 데이터를 받기 위한 헤드인 Receiver 결정을 위해 2개의 확률식을 독립적으로 적용하는 방식을 사용한다.

[3]에서는 모든 노드에게 동일한 기회를 부여하기 위해, 최근의 이전 라운드에서 헤드로 선출된 노드는 현재 라운드의 헤드 후보에서 제외 (확률식 값을 0으로 설정)하였다. 하지만, 본 연구에서는 각 멤버의 배터리량을 동적으로 고려하여 헤드를 선출할 것이므로 최근의 이전 라운드에서 헤드로 선출되었더라도 현재 라운드의 헤드 후보에 포함시키게 된

다. 멤버들에게 동일한 기회를 부여하는 것이 목적이 아니라, 남아 있는 배터리량이 충분한 멤버들을 주로 헤드로 결정하는 것이 목적이기 때문이다. 따라서, 식 (1)과 같은 Sender 결정을 위한 확률식과, 식 (2)와 같은 Receiver 결정을 위한 확률식을 유도할 수 있다. P_s 와 P_r 은 각각, 커뮤니티를 구성하는 전체 멤버들 중에 Sender와 Receiver의 개수를 결정하는 확률(멤버의 수에 따라 사용자가 임의 조정할 수 있는 변수로서, $\frac{1}{P_s}$ 와 $\frac{1}{P_r}$ 이 정수가 되는 값으로 설정한다), r 은 현재의 라운드, mod는 나머지 연산을 의미한다.

$$T_{sender}(N) = \frac{P_s}{1 - P_s \left(r \bmod \frac{1}{P_s} \right)} \quad (1)$$

$$T_{receiver}(N) = \frac{P_r}{1 - P_r \left(r \bmod \frac{1}{P_r} \right)} \quad (2)$$

또한, WEL 알고리즘에서는 커뮤니티가 생성될 때 멤버들 각자의 잔여 배터리량을 Pway로 전송하고, Pway는 멤버들의 배터리량 ($B(N)$)을 취합하여 모든 멤버들에게 전송한다. 이렇게 함으로써, 각 멤버들은 커뮤니티 내 모든 멤버들의 잔여 배터리량 정보와 가중치 ($W(N)$)를 테이블 (BT)로 가지고 있게 된다(예시 : 표 1).

표 1 BT 예시

Table 1 An example of BT

커뮤니티 멤버	배터리량	가중치 (최대 배터리량 기준)
1	$B(1) = 250$	$W(1) = 0.83$
2	$B(2) = 300$	$W(2) = 1$
3	$B(3) = 200$	$W(3) = 0.67$
...
N	$B(N) = 100$	$W(N) = 0.33$

위와 같은 테이블의 가중치를 이용하여, 각 멤버의 잔여 배터리량 정보를 클러스터 헤드 결정 확률에 반영시킬 수 있다. 즉, 앞서 유도했던 각 멤버의 확률식 $T_{sender}(N)$ 와 $T_{receiver}(N)$ 에, 각 멤버의 가중치 $W(N)$ 을 곱하여 최종 Sender 및 Receiver 결정 확률식 $WT_{sender}(N)$ 와 $WT_{receiver}(N)$ 를 식 (3), 식 (4)와 같이 표현할 수 있다.

$$WT_{sender}(N) = \frac{P_s \cdot W(N)}{1 - P_s \left(r \bmod \frac{1}{P_s} \right)} \quad (3)$$

$$WT_{receiver}(N) = \frac{P_r \cdot W(N)}{1 - P_r \left(r \bmod \frac{1}{P_r} \right)} \quad (4)$$

각 멤버들은 최근 $\left(\frac{1}{P_s} - 1 \right)$ 또는 $\left(\frac{1}{P_r} - 1 \right)$ 라운드동안 클러스터 헤드의 역할을 3회 미만 (본 수치는 본 논문의 실험 환경에서 최적화된 값이고, 네트워크 특성이나 응용분야에 따라 조정하여 적용 가능함) 수행하였다면 현재 라운드의 헤드 후보가 되고, 자신의 $WT_{sender}(N)$ 값과 $WT_{receiver}(N)$ 값을 0 ~ 1 사이의 랜덤 상수 (R_s , R_r)들과 비교하여 크다면 본인을 Sender 또는 Receiver로 결정하게 된다. 동시에 Sender와 Receiver로 결정되는 결과가 발생할 때는, $WT_{sender}(N)$ 값과 $WT_{receiver}(N)$ 값을 비교하여 큰 값을 갖는 쪽의 역할을 맡게 된다.

2.3 클러스터 구성

Sender 또는 Receiver 헤드로 결정된 멤버들은, 주변의 멤버들에게 클러스터 헤드로 결정되었음을 알려주기 위해 헤드 결정 메시지 (Head Decision Message, HD_MSG_s or HD_MSG_r)를 브로드캐스팅한다. 복수개의 헤드 결정 메시지를 수신한 멤버는 신호의 세기를 비교하여 가장 큰 쪽의 헤드에게 클러스터 가입 메시지 ($JOIN_MSG_s$ or $JOIN_MSG_r$)를 전송하고, 가입 메시지를 수신한 헤드는 가입 확인 메시지 ($JOIN_ACK_s$ or $JOIN_ACK_r$)를 클러스터 멤버들에게 전송함으로써 클러스터가 구성되었음을 알린다.

2.4 멀티미디어 데이터 전송

클러스터가 형성된 이후에, 각 클러스터 내의 멤버는 멀티미디어 데이터를 클러스터 외부로 보낼 경우 Sender 헤드를 이용하고, 멀티미디어 데이터를 클러스터 외부로부터 받을 때는 Receiver 헤드를 이용한다. Sender와 Receiver 헤드는 자신이 관리하는 클러스터 멤버들의 멀티미디어 데이터와 라우팅 정보 등의 추가 정보들을 패킷화하여 전송한다.

2.5 BT 갱신

BT 는 초기 배터리량을 기본으로 작성되기 때문에, 가중치가 높은 멤버에게는 헤드 역할이 과도하게 집중될 수가 있다. 이와 같은 문제를 해결하기 위해 WEL 알고리즘은, 추가적인 부하 분산 정책을 사용한다. 즉, 최근 $\left(\frac{1}{P_s} - 1 \right)$ 또는 $\left(\frac{1}{P_r} - 1 \right)$ 라운드가 종료되면, 각 Sender 헤드를 통해 클러스터 멤버들의 잔여 배터리량 정보를 Pway에게 전송하며, 이 정보를 수신한 Pway가 BT 를 갱신하여 각

Receiver 헤드들에게 전달하게 된다. 갱신된 BT를 이용하여 헤드를 재선출하면 특정 멤버에게 역할이 집중되는 것을 막을 수 있다. 기초로 삼은 LEACH 알고리즘과 본 논문에서 제안하는 WEL 알고리즘 모두 개별적인 노드(멤버)가 각각의 고유한 확률식을 기반으로 본인을 헤드로 결정할지를 판단하도록 설계되어 있으므로, Pway가 Sender와 Receiver를 일괄적으로 결정하는 방식을 적용하기는 어려웠다. 커뮤니티 컴퓨팅의 특성 상, 각각의 멤버들이 Pway의 제어 없이 자유롭게 커뮤니티를 구성하기도 하고 해체하기도 한다. 따라서 Pway는 가장 기본적인 역할만을 수행하도록 하고 멤버들이 독립적으로 헤드 결정 역시 수행하는 것이 적합하다고 볼 수 있다.

위의 알고리즘에 대한 순서도는 그림 4와 같다.

3. 실험결과

제안하는 WEL 알고리즘의 성능을 비교해보기 위해, Microsoft Visual C++를 이용하여 간단한 프로그램을 작성하여 실험을 하였다. 본 알고리즘의 적용 환경인 '멀티미디어 데이터 전송을 위한 커뮤니티 컴퓨팅 환경의 무선네트워크'를 가정하기 위해서 표 2와 같은 실험 조건을 구성하였다.

표 2. 실험 조건

Table 2. Experimental conditions

멤버의 수	20개
P_s	0.1 (10%)
P_r	0.1 (10%)
멤버들의 초기 배터리량	최대치를 1000Ah (100%), 최소치를 200Ah (20%)로 가정. (100%, 80%, 80%, 70%, 50%, 100%, 70%, 50%, 40%, 50%, 60%, 40%, 30%, 30%, 20%, 50%, 70%, 60%, 50%, 40%)
데이터 전송을 위한 전력 소모	한 라운드에 5번의 데이터 전송이 발생. 멤버와 클러스터 헤드 사이의 단거리 통신 (ex. Zigbee)을 위한 전력소모 20mAh. 클러스터 헤드와 서버와의 원거리 통신 (ex. WLAN)을 위한 전력소모 500mAh (송신), 250mAh (수신)
실험 종료의 기준	20개의 멤버 중에, 배터리량이 0이하가 되는 멤버 발생 시점

실험을 위한 커뮤니티는 멤버 20개로 구성되어 있으며, 커뮤니티 내의 Sender와 Receiver의 개수를 결정하는 확률은 각각 10% ($P_s=0.1$, $P_r=0.1$)로 설정하였다. 또한, 멤버들의 초기 배터리량을 최대 1000Ah에서 최소 200Ah까지 다양하게 초기화하였고, 하나의 Sender와 Receiver가 선출되면 (라운드의 개념) 5번의 데이터 전송을 하는 것으로 설정하였다.

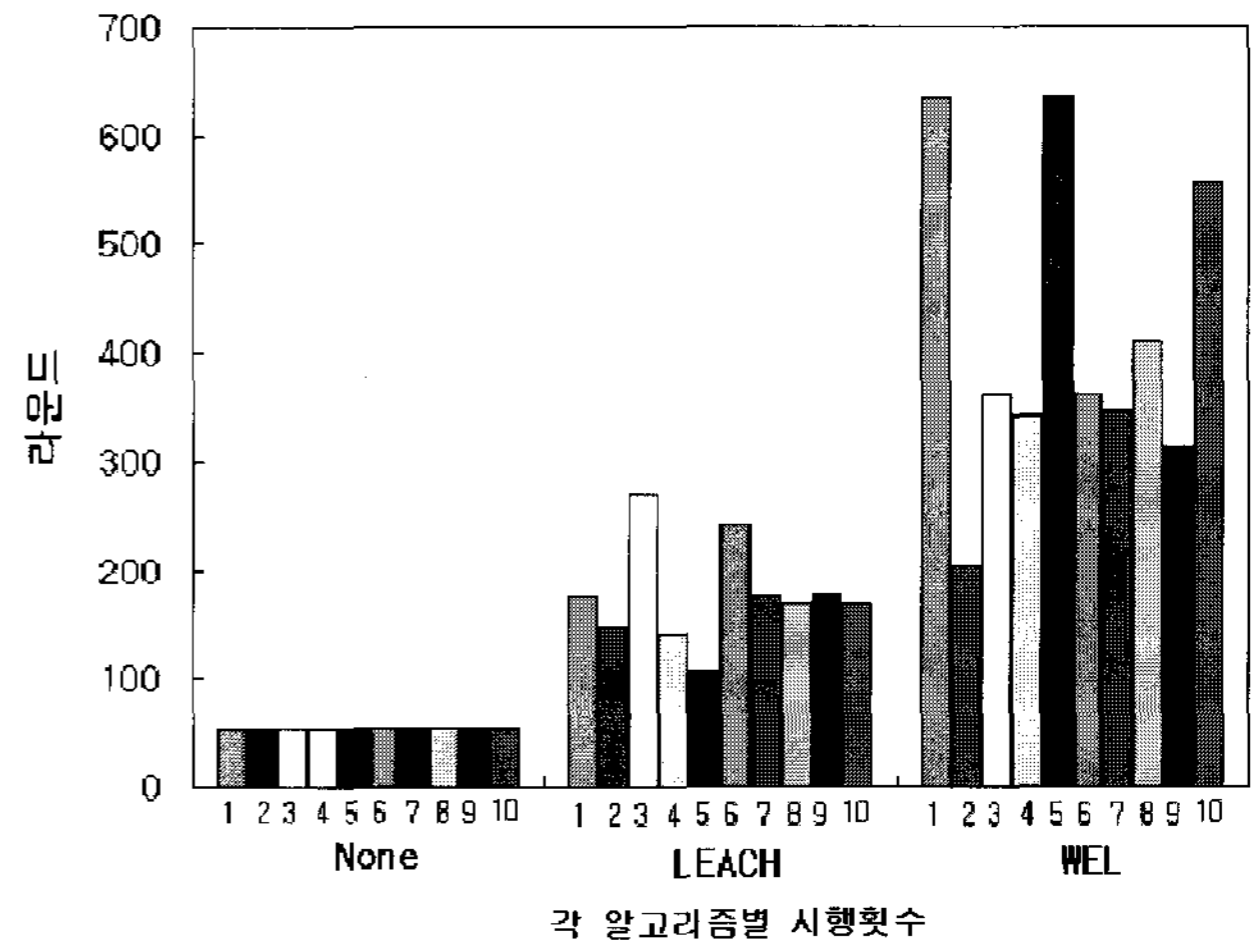


그림 3 실험 결과

Fig. 3 Experimental results

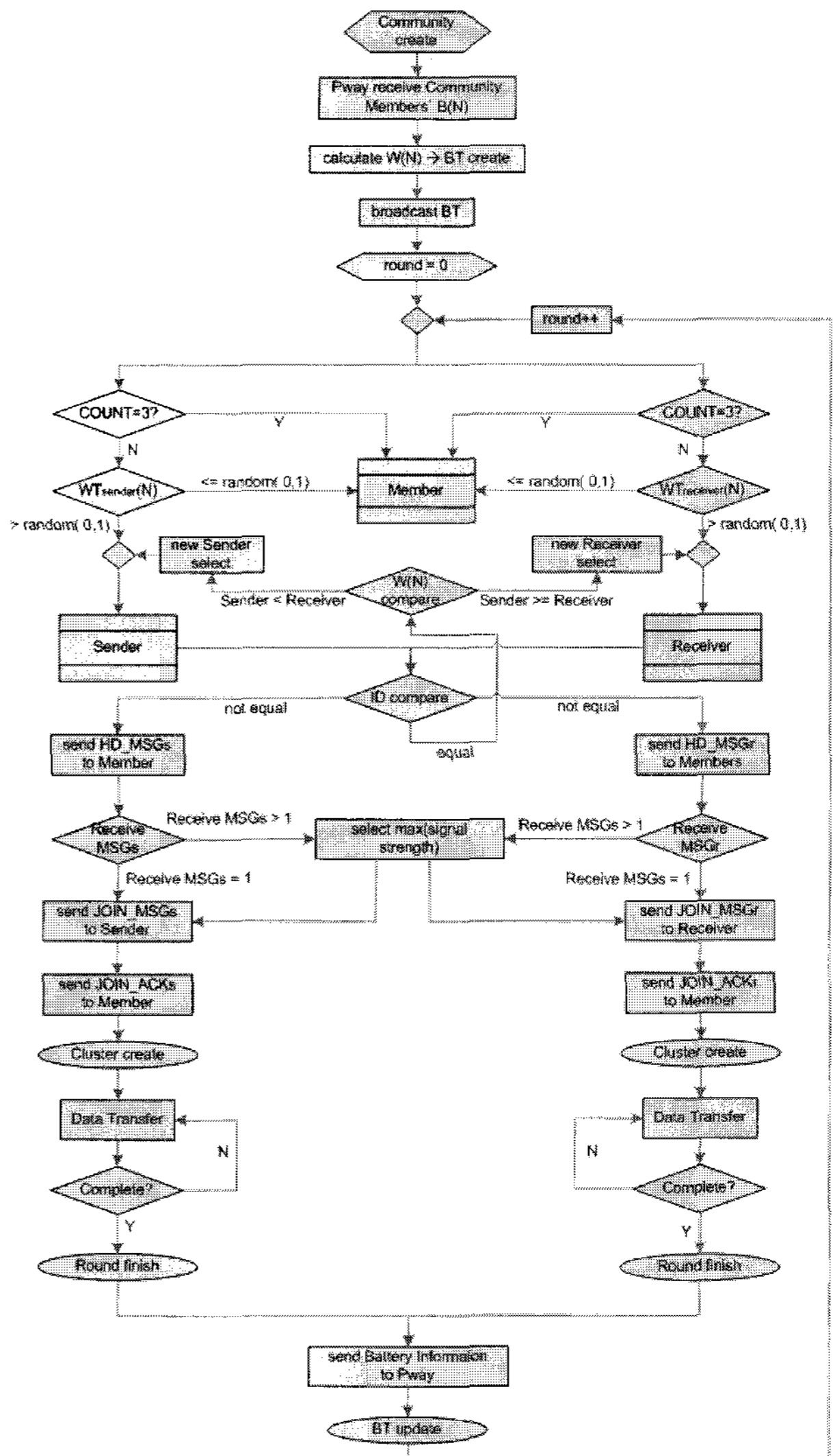


그림 4. WEL 알고리즘 순서도

Fig. 4 The flowchart of WEL algorithm

위의 실험조건 하에서, 클러스터 헤드를 사용하지 않는 데이터 전송 방법 (None), LEACH 알고리즘을 이용한 데이터 전송 방법 (LEACH), 제안하는 WEL 알고리즘을 이용한 데이터 전송방법 (WEL)에 대한 실험을 10회씩 시행하였고, 그 결과를 그림 4에 나타내었다. 그림 4의 세로축은 작성한 프로그램 상에서 각 알고리즘 별로 반복된 라운드 횟수이고, 이것은 커뮤니티의 생존시간을 의미하게 된다. 즉, 각 라운드 별로 5번의 데이터 전송이 일어나는 상황을 가정했으므로, 총 데이터 전송 횟수는 라운드 수에 5를 곱한 값으로 취할 수 있을 것이다.

실험 결과를 보면, 클러스터 헤드를 사용하지 않는 데이터 전송 방법의 경우에는 270회, LEACH 알고리즘을 이용한 데이터 전송 방법의 경우에는 10회 평균 884회, WEL 알고리즘을 이용한 데이터 전송방법의 경우에는 10회 평균 2080회의 데이터 전송을 수행한 후 커뮤니티가 종료되었음을 알 수 있다.

4. 결 론

본 논문에서는, 향후 그 영역이 점차 확대될 커뮤니티 컴퓨팅 환경을 지원하기 위한 알고리즘을 제안하였다. 커뮤니티를 구성하는 멤버들 중, 특정 멤버의 생존시간이 종료되어 커뮤니티의 목표 서비스를 제공하지 못하는 상황에 대처하기 위해서, 각 멤버들의 배터리량을 최대한 활용하여 각 멤버의 생존시간 및 전체 커뮤니티의 생존시간을 최적화시키는 것을 연구 목표로 삼았다. 기존의 연구를 활용하여, 클러스터 구성 후 특정 헤드에게 멀티미디어 데이터 전송을 전달시키는 방법을 도입하였고, 간단한 확률식으로 모든 멤버에게 동일한 기회를 부여하고자 하였다. 추가적으로, 각 멤버의 배터리량을 기준으로 확률식을 새롭게 구성하여, 배터리량이 많은 멤버들에게는 헤드 역할이 확률적으로 더 많이 돌아가도록 하여 전체 커뮤니티 내 멤버들의 생존시간이 균형을 이루도록 하였다.

기존의 다양한 클러스터 헤드 선출 알고리즘들[4]-[12]과 다르게 Sender와 Receiver로서 헤드를 구분하여서 데이터 전송의 부담을 분산시킴으로써, 대용량 멀티미디어 데이터 전송을 위한 무선네트워크 환경에서 성능이 극대화될 수 있도록 하였다. 또한, 다른 알고리즘들과 비교했을 때 상대적으로 간단한 LEACH 알고리즘을 기반으로 새로운 알고리즘을 구성함으로써 보다 쉽게 활용할 수 있도록 하였다.

제안한 알고리즘을 적절히 활용한다면, 저전력 동작이 가장 기본이 되는 유비쿼터스 환경 및 커뮤니티 컴퓨팅 환경의 구성에 큰 도움이 될 것이라 생각한다.

감사의 글

본 연구는 서울시 산학연 협력사업 연구비 지원과 숭실대학교 교내연구비 지원으로 이루어졌습니다.

참 고 문 헌

- [1] 조위덕, Ubiquitous Computing Paradigm, 유비쿼터스 컴퓨팅사업단, 2005년 5월
- [2] M. J. Handy, M. Haase, and D. Timmermann, "Low Energy Adaptive Clustering Hierarchy with Deterministic Cluster-Head Selection," International Workshop on Mobile and Wireless Communications Network, pp.368-372, Sep. 2002
- [3] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," Wireless Comm. IEEE, Vol. 1, No. 4, pp.660-670, 2002.
- [4] S. Lindsey and C. S. Raghavendra, "PEGASIS: Power Efficient Gathering in Sensor 95 Information Systems," Proc. of IEEE Aerospace Conference, pp.1125-1130, 2002.
- [5] A. Manjeshwar and D. P. Agrawal, "TEEN: a routing protocol for enhanced efficiency in wireless sensor networks," Proc. of 15th International Conference on Parallel and Distributed Processing Symposium, pp.2009-2015, 2001.
- [6] A. Manjeshwar and D. Agrawal, "APTEEN: A Hybrid Protocol for Efficient Routing and Comprehensive information Retrieval in Wireless Sensor Networks," International Parallel and Distributed Processing Symposium: IPDPS 2002 Workshops, pp.195-202, 2002.
- [7] V. Kawadia and P. R. Kumar, "Power Control and Clustering in Ad Hoc Networks," Proc. of IEEE INFOCOM, 2003
- [8] V. Mhatre and C. Rosenberg, "Design Guidelines for Wireless Sensor Network: Communication, Clustering and Aggregation," Elsevier Ad Hoc Networks, 2003
- [9] O. Tomoyuki, I. Shinji, K. Yosiaki and I. Kenji, "An Adaptive Multihop Clustering Scheme for Ad Hoc Networks with High Mobility," IEICE Tran. on Fundamentals, 2003
- [10] H. Chan and A. Perrig, "ACE: An Emergent Algorithm for Highly Uniform Cluster Formation," 2004 European Workshop on Sensor Networks, 2004
- [11] O. Younis and S. Fahmy, "Distributed Clustering in Ad-hoc Sensor Networks: A Hybrid, Energy-Efficient Approach," Proc. of IEEE INFOCOM 2004, 2004.
- [12] Lee, SangHak; Ham, KyungSun; Park, ChangWon, "Distributed Clustering for Wireless Sensor Networks," ISCIT '06, 2006

저 자 소 개



김기영 (金基英)

1979년 10월 29일생. 2002년 숭실대 컴퓨터학부 졸업. 2004년 숭실대 대학원 컴퓨터학과 석사과정 졸업. 2004년 3월 ~ 현재 동대학원 박사과정

Tel : 02-813-0682

E-mail : kky@ic.ssu.ac.kr



노경우 (盧暉佑)

1978년 10월 23일생. 2006년 숭실대 컴퓨터학부 졸업. 2008년 동 대학원 컴퓨터학과 석사과정 졸업



박혜성 (朴慧誠)

1981년 1월 23일생. 2005년 대전대 컴퓨터공학과 졸업. 2006년 ~ 현재 숭실대 대학원 컴퓨터학과 석사과정.



김석운 (金錫潤)

1958년 8월 12일생. 1980년 서울대 공대 전기공학과 학사. 1990년 University of Texas at Austin 전기, 컴퓨터학과 석사과정 졸업. 1993년 University of Texas at Austin 전기, 컴퓨터학과 박사과정 졸업. 1982년~1987년 한국전자통신연구소 연구원. 1993년~1995년 Motorola Inc., Senior Staff Engineer. 1995년~현재 숭실대 컴퓨터학부 교수.