

RGG/WSN을 위한 분산 저장 부호의 성능 분석

정호영*

A Performance Analysis of Distributed Storage Codes for RGG/WSN

Ho-Young Cheong*

요약 본 논문에서는 IoT/WSN을 랜덤 기하 그래프를 이용하여 모델링하고 WSN에서 발생하는 데이터를 효율적으로 저장하기 위해 사용되는 지역 부호의 성능을 고찰하였다. 노드 수가 $n=100, 200$ 인 무선 센서 네트워크를 랜덤 기하 그래프로 모델링하여 분산화된 저장 코드의 부호 성능을 시뮬레이션을 통해 분석하였다. 네트워크의 총 노드 수가 $n=100$ 일 때와 200 일 때 복호율 η 에 따른 부호 성공률은 노드 수 n 보다는 소스 노드 수 k 값에 따라 좌우됨을 알 수 있었다. 특히 n 값에 관계없이 $\eta \leq 2.0$ 일 때 부호 성공 확률은 70% 를 상회함을 알 수 있었다. 복호율 η 에 따른 부호 연산량을 살펴본 바, BP 부호 방식의 부호 연산량은 소스 노드 수 k 값이 증가함에 따라 기하급수적으로 증가함을 알 수 있었다. 이는 소스 노드의 수가 증가할수록 LT 부호의 길이가 길어지고 이에 따라 부호 연산량이 크게 증가하는데 원인이 있는 것으로 생각된다.

Abstract In this paper IoT/WSN(Internet of Things/Wireless Sensor Network) has been modeled with a random geometric graph. And a performance of the decentralized code for the efficient storage of data which is generated from WSN has been analyzed. WSN with $n=100$ or 200 has been modeled as a random geometric graph and has been simulated for their performance analysis. When the number of the total nodes of WSN is $n=100$ or 200 , the successful decoding probability as decoding ratio η depends more on the number of source nodes k rather than the number of nodes n . Especially, from the simulation results we can see that the successful decoding rate depends greatly on k value than n value and the successful decoding rate was above 70% when $\eta \leq 2.0$. We showed that the number of operations of BP(belief propagation) decoding scheme increased exponentially with k value from the simulation of the number of operations as a η . This is probably because the length of the LT code becomes longer as the number of source nodes increases and thus the decoding computation amount increases greatly.

Key Words : Random Walk, Random Geometric Graph, Internet of Things, Decoding Ratio, Erdős-Rényi Model

1. 서론

각 종 센서와 통신 장치를 부착한 수많은 전자 기기들이 거대한 인터넷에 연결되어 동작하는 현

대의 IoT 네트워크 시스템은 기존의 무선 센서 네트워크(wireless sensor network, WSN)가 진화된 형태로 볼 수 있다. 센서mfm 통해 데이터를 발생시키기도 하고 발생된 데이터를 다른 지점으로 전달

Funding for this paper was provided by Namseoul university in 2016.

*Corresponding Author : Department of Information and Communication Engineering, Namseoul University (hycheong@nsu.ac.kr)

Received October 14, 2017

Revised October 16, 2017

Accepted October 17, 2017

하는 전자 단말기를 노드(node)라고 한다. 특히 온도나 기후 등을 측정하고 화재 등 여러 가지 재난 현상들을 감시하거나 측정하여 데이터를 발생시키는 작은 규모의 전자 센서 장치들을 소스 노드(source node)라고 한다. 이들은 대부분 근접하기가 어렵거나 열악한 환경에 설치되어 있고 에너지, 메모리 용량, 연산 능력, 배터리 수명, 통신 대역폭 등의 자원이 한정되어 있다. 소스노드들에 의해 생성된 데이터들은 안정된 전원과 풍부한 저장용량을 가진 데이터 수집 장치(싱크 노드) 등에 전송하여 저장할 필요가 있다. 그러나 발생하는 데이터량이 방대하므로 몇몇 개의 싱크노드로 데이터를 집중시키는 것은 효율적이지 못하므로 네트워크 내에 분산 저장하여 일부 노드들이 유실되는 경우에도 유실되지 않은 노드들로부터 데이터를 복원할 수 있도록 해야 한다[1][11]. 수 많은 노드들이 산재한 IoT/WSN에서 발생한 데이터를 분산 저장하여 처리하는 가장 효율적인 방법으로 파운틴 부호를 이용한 분산 저장 방법이 가장 효율적인 것으로 인정되고 있다[2].

본 논문에서는 IoT/WSN을 랜덤 기하 그래프(random geometric graph, RGG)를 이용하여 모델링하고 이를 이용하여 파운틴 부호 기반 분산 저장 시스템의 부호 성능을 시뮬레이션을 통해 분석한다.

2. RGG 랜덤 그래프를 이용한 WSN/IoT 모델링

2.1 Erdős-Rényi 모델과 RGG 모델

n 개의 노드들을 갖는 무선 센서 네트워크는 데이터를 발생시키는 k 개의 소스 노드들을 포함한 형태로 볼 수 있으며 이는 수학적으로 랜덤 그래프(random graph)를 이용하여 모델링할 수 있다. 1950년대에 처음 제시된 Erdős-Rényi 랜덤 그래프에서는 평면상에 n 개의 노드(vertex라고 함)들이 균일하고도 랜덤하게 분포되어 있고 노드들 사이가 정해진 확률 p 에 의해 edge 들이 연결되

어 랜덤 그래프를 형성한다[2][3]. 이와 같은 Erdős-Rényi 랜덤 그래프를 $G(n, p)$ 로 표시한다. Erdős-Rényi 랜덤 그래프의 가장 큰 장점은 랜덤 그래프의 특성이 수학적으로 거의 완전하게 설명될 수 있다는 점이며 그동안 이에 대한 연구가 활발히 이루어져 왔다. 그러나 Erdős-Rényi 랜덤 그래프는 실제 네트워크의 특성과는 부합되지 않는 면이 많아 현대의 복잡한 네트워크를 모델링하는 데에는 한계가 있다. 예를 들어, 노드와 노드 사이를 연결하는 edge 들의 경우 Erdős-Rényi 랜덤 그래프에서는 통계적으로 i.i.d. 특성을 갖는 것으로 가정하고 있으나 현실세계의 네트워크는 크게 다르다. 특히 이질적인 네트워크들이 서로 연결되어 하나의 거대한 네트워크를 이루고 있는 인터넷의 특성을 Erdős-Rényi 랜덤 그래프를 통해 모델링하는 것은 어렵다. 이러한 단점을 보완할 수 있는 랜덤 그래프 모델이 랜덤 기하 그래프(이하 RGG로 표시)이며 무선 센서 네트워크의 특성을 수학적으로 모델링하는데 널리 사용되고 있다[3].

RGG에서는 노드들이 일정 영역 내에서 임의의 확률 분포에 따라 랜덤하게 산재하며, 주어진 거리 r 내에 있는 두 노드(이웃 노드라고 함)는 edge로 연결된다. 이때 거리 r 은 노드

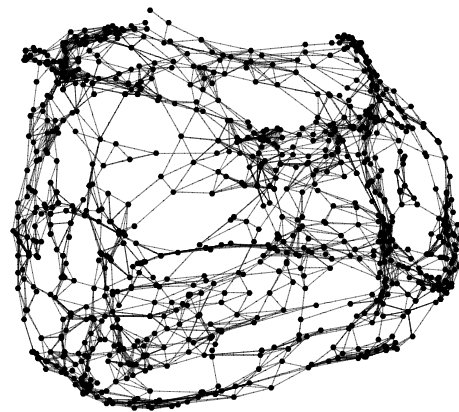


그림 1. 랜덤기하그래프를 이용한 WSN의 모델링 예 ($d=2, n=1000, r=0.05$)
 Fig. 1. A WSN Model using RGG ($d=2, n=1000, r=0.05$)

들의 통신 범위를 반영한다. f 를 R^d 에서 정의된 확률 밀도 함수라고 하고, X_1, X_2, \dots 를 동일한 확률 밀도 f 를 갖고 랜덤하게 산재된 노드들을 나타내는 i.i.d. 랜덤 변수라고 하면 노드 X_n 과 통신 반경 r 내에서 연결된 edge들이 형성하는 랜덤 그래프를 랜덤 기하 그래프 $G(X_n, r)$ 이라고 한다. 그림 1은 $[0, 1]$ 사이에서 균일한 확률 분포 f 를 갖는 랜덤 기하 그래프를 나타낸 것인데, 이때 $d=2$ 이고 노드 수는 $n=1000$ 이며 통신 반경은 $r=0.05$ 이다[3].

k 개의 소스 노드로부터 발생된 k 개의 소스 데이터를 네트워크 내에 저장하기 위해 소스 패킷은 랜덤 워크(random walk)를 이용하여 전달된다. 각 노드에서 랜덤 워크 패킷을 수신하면 부호화 과정을 수행하여 부호화된 부호 패킷을 저장하고 랜덤 워크 패킷은 다시 이웃 노드 중 하나로 전송된다[4][6][7].

2.2 분산 저장을 위한 LT 부호

LT 부호(Luby transform code)는 무선 센서 네트워크에서 데이터 분산 저장에 사용될 수 있는 효율적인 부호이다[5]. k 개의 소스 블록 $\{x_1, x_2, \dots, x_k\}$ 에 대해 LT 부호는 다음과 같이 생성된다. 정수 집합 $\{1, 2, \dots, k\}$ 에 대해 적용되는 확률 분포 Ω 를 이용해 d 개의 정수를 랜덤하게 선택한다. LT 부호에 사용되는 확률 분포 Ω 는 Ideal Soliton 분포이며 식 (1)과 같다[8][9].

$$\Omega_I(d) = \begin{cases} 1/k, & d=1, \\ 1/[d(d-1)], & d=2, 3, \dots, k \end{cases} \quad (1)$$

Ideal Soliton 분포를 이용한 LT 부호는 복호화 과정에서 복호 과정이 중단되는 경우가 빈번하게 발생하여 복호화 전체가 실패로 끝나는 경우가 있으므로 이를 방지하기 위해 실제로는 식 (2),(3)과 같은 Robust Soliton 분포 $\Omega_R(d)$ 를 사용하게 된다. $R = c_0 \ln(k/\delta) \sqrt{k}$ 라고 하면

$$r(d) = \begin{cases} R/dk, & d=1, \dots, k/R-1, \\ R \ln(R/\delta)/k, & d=k/R, \\ 0, & d=k/R+1, \dots, k \end{cases} \quad (2)$$

이고, 이때 c_0 는 임의의 상수이며 $0 < \delta < 1$ 의 값을 가진다. 식 (2)를 이용하여 Robust Soliton 분포 $\Omega_r(d)$ 는 식 (3)과 같이 표현할 수 있다.

$$\Omega_R(d) = \frac{\tau(d) + \Omega_I(d)}{\sum_{i=1}^k (\tau(i) + \Omega_I(i))}, \quad d=1, 2, \dots, k \quad (3)$$

임의의 노드 u 는 센서 노드로부터 출발한 랜덤 워크의 길이가 $C_1 n \log n$ 보다 작은 패킷을 수신할 경우 LT 부호화 과정을 수행하고 랜덤하게 선택된 이웃 노드로 패킷을 전송한다. 한편, 랜덤 워크의 길이가 $C_1 n \log n$ 보다 큰 패킷은 버린다. k 개의 소스 패킷들이 모두 $C_1 n \log n$ 값 보다 큰 값을 가지면 분산 전송과 부호화 과정은 종료된다.

복호화 과정은 데이터 수집기가 h 개의 노드들로부터 저장 데이터를 수집하여 복호화 과정을 수행한다. 이때 $\eta = h/k$ 를 복호율(decoding ratio)이라고 하며 이는 LT 부호에서 복호 오버헤드에 해당한다. h 개의 부호 데이터로부터 k 개의 소스 데이터를 LT 부호의 복호 기법인 BP(belief propagation) 복호 기법을 이용하여 복호하며, k 개의 소스 데이터가 모두 복호되지 않은 채로 복호화 과정이 중단되면 다른 노드들로부터 저장된 데이터를 추가로 전송 받아 복호화 과정을 계속하고 전송 받은 노드들이 없으면 복호화 과정은 실패로 끝나게 된다[10].

각 노드에서 랜덤 워크 패킷을 수신하면 부호화 과정을 수행하여 부호화된 부호 패킷을 저장하고 랜덤 워크 패킷은 다시 이웃 노드 중 하나로 전송된다. LT 부호의 부호화가 데이터를 발생시키는 소스 노드에서 하는 것이 아니라 데이터를 수신하는 분산된 각 노드에서 개별적으로 수행한다는 점이 기존의 LT 부호화와 가장 다른 점이다.

3. Decentralized storage 부호

k 개의 데이터 패킷들을 $\mathbf{x} = [x_0, x_1, \dots, x_{k-1}]$ 으로 표시하자. k 개의 데이터 패킷들을 이용해 식 (4)와 같이 부호 패킷 $\mathbf{y} = (y_1, y_2, \dots)$ 을 생성할 수 있다. 부호 패킷 y_i 는 k 개의 소스 패킷들로부터 다음과 같이 생성된다[8]-[10].

$$y_i = x_{r_0} \oplus x_{r_1} \oplus \dots \oplus x_{r_{d-1}} \quad (4)$$

부호화기는 $[0, k]$ 범위에 있는 정수 d 를 확률 분포 $\Omega_r(d)$ 를 이용하여 선택한 후 이에 해당하는 d 개의 데이터 패킷들을 식 (4)와 같이 XOR 연산을 적용하여 부호화 한다. 모든 부호 패킷은 부호 패킷을 생성하기 위해 사용한 d 개의 소스 패킷들이 XOR 연산에 적용됨을 나타내는 벡터 $\mathbf{b}_i = [b_{i,0}, b_{i,1}, \dots, b_{i,k-1}]$ 를 추가하여 송신한다. 이때 소스 패킷 m_j 가 부호 패킷 y_i 를 생성하는데 사용되었으면 $b_{i,j} = 1$ 값을 갖고 사용되지 않았으면 $b_{i,j} = 0$ 의 값을 가진다. 여기에서 x_{r_j} 는 XOR 연산에 사용된 r_j 번째 데이터 패킷이다.

3.1 분산 저장을 위한 부호화 알고리즘

Decentralized LT 부호화는 WSN에 존재하는 k 개의 소스 노드가 분산되어 있으므로 기존의 LT 부호와 같이 한 곳에서 부호화를 할 수 없다. 따라서 분산 되어있는 n 개의 노드들이 각자 k 개의 소스 패킷에 대한 부호 패킷을 생성하여 각 각의 노드에서 각 각 생성한 부호 패킷을 따로 저장하는 것이다. 즉 각 노드들은 RSD 분포 $\Omega_d(d)$ 를 통해 발생시킨 d 를 이용해 각 노드를 방문하는 소스 패킷 중에서 d 개의 소스 패킷을 선정하여 부호화한 후 각자 분산 저장하게 된다. 부호화 과정을 요약하면 다음과 같다.

(1) 초기화 단계 : 각 소스 노드 $s(s=1, \dots, k)$ 는 데이터 x_s 에 ID 및 life counter를 포함하고 있는 헤더를 추가한다. 이제 소스 노드는 이웃 노드들 중 한 노드를 랜덤하게 선정하 소스 데이터를 전

송하고, 수신 노드 u 는 부호화할 부호 패킷을 $y_u = 0$ 으로 초기화한다.

(2) 부호화 단계 : 한 노드 u 에서 확률 분포 $\Omega_d(d)$ 에 따라 값 d 를 생성한다. 랜덤 워크 형태로 수신된 데이터 패킷의 수명 카운터 값이 $C_1 n \log n$ 보다 작으면 노드 u 는 (d/k) 확률 값에 따라 수신된 데이터 패킷을 선정하여 부호화 한다. 수신 패킷의 카운터 값을 증가시키고 부호화 과정이 끝나면 다음 이웃 노드로 전송할 수 있도록 대기시킨다. 만일 수명카운터 값이 $C_1 n \log n$ 이상이면 모든 노드들을 모두 방문한 것으로 보아 없애고 네트워크 내의 노드들이 모두 k 개의 소스 패킷을 수신하면 부호화 단계를 마친다.

3.2 BP 복호 알고리즘

소스 데이터를 복원하고자 하는 데이터 수집 장치는 $k(1+\epsilon)$ 개의 노드들로부터 각 노드가 가지고 있는 부호 패킷을 전송 받아 복호하게 된다.

이때 부호 패킷에는 부호화에 사용된 정보 벡터 $\mathbf{B} = [b_1, b_2, \dots, b_n]^T$ 가 같이 포함되어 있으며 이를 이용해 선형 연립방정식 $\mathbf{B}\mathbf{x} = \mathbf{y}$ 을 구성할 수 있다. 선형 연립방정식 $\mathbf{B}\mathbf{x} = \mathbf{y}$ 로부터 독립이 아닌 방정식들만 제거하면 $(k \times k)$ 행렬 \mathbf{G} 와 $(k \times 1)$ 벡터 \mathbf{Y} 로 구성되는 연립방정식 $\mathbf{G}\mathbf{m} = \mathbf{Y}$ 를 얻을 수 있다. 이제 연립방정식 $\mathbf{G}\mathbf{m} = \mathbf{Y}$ 의 해를 구하는 과정이 복호화 과정이다.

BP 복호 알고리즘에서는 차수가 1인 행 $\mathbf{G}[i]$ 를 탐색하여 행 $\mathbf{G}[i]$ 에서 '1'의 열 위치가 j 라고 하면 $m_j = y_i$ 로 복호한다. 이때 $\mathbf{G}[i]$ 는 \mathbf{G} 행렬의 i 번째 행을 의미한다. 동시에 $\mathbf{G}[i]$ 의 j 번째 열에 있는 모든 '1'들을 '0'으로 바꿈과 동시에 '0'으로 바뀌는 행에 해당하는 부호 패킷 y_l 을 y_i 와 XOR 연산을 하여 선형 방정식을 갱신한다. \mathbf{G} 행렬의 원소가 모두 0이 될 때까지 반복하는데, \mathbf{G} 행렬의 원소가 모두 0이 되면 복호가 성공하여 종료되며 모두 0이 되기 전에 차수가 1인 행이 존재하지 않으면 복호 실패로 중단된다. \mathbf{G} 행렬의 원소가 모

두 0이 되기 전에 복호 실패로 복호화 과정이 중단되면 이웃 노드들에게 부호 패킷을 추가로 요청하게 되며 차수-1의 부호 벡터를 기다려 복호를 계속 진행한다. 따라서 BP 복호 알고리즘은 k 값이 작을 때 오버헤드가 크게 증가하는데 이는 복호가 성공하기 전에 리플이 자주 비기 때문이다 [8].

4. 시뮬레이션 결과

본 논문에서는 랜덤 기하 그래프로 모델링된 $n = 100, 200$ 인 WSN에서 분산 저장부호의 복호 성능을 살펴보기 위해 다음과 같이 시뮬레이션을 수행하였다.

우선 WSN은 랜덤 워크 형태로 데이터가 전송되는 랜덤 기하 그래프로 모델링하였다. 노드 연결성을 확보하기 위해 통신 반경이 $r = 0.2$ 이상의 값을 갖는 랜덤 그래프를 생성하였다. 생성된 랜덤 그래프 상에서 n 개의 노드들에 대해 10%와 20%에 해당하는 k 개의 소스 노드들을 균일하고 랜덤하게 선택하였다. RSD 확률 분포 $\Omega_r(d)$ 를 구성하는 파라미터는 $c = 0.1$ 과 $\delta = 0.01$ 로 하였다.

데이터 수집 장치가 복호를 위해 부호 패킷을 요청한 이웃 노드들이 h 개라고 하면 복호율(decoding ratio) η 는 $\eta = h/k$ 로 계산할 수 있다. n 개의 노드들 중에서 h 개의 노드들을 랜덤하게 선택하여 데이터 요청 노드 집합을 구성한 후 BP 복호를 적용하여 시뮬레이션을 수행하였다.

그림 2는 네트워크의 총 노드 수가 $n = 100$ 일 때와 200일 때 복호율 η 에 따른 복호 성공률 P_s 를 나타낸 것이다. 그림 2에서 BP 복호 방식의 복호 성공 확률은 노드 수 n 보다는 소스 노드 수 k 값에 따라 크게 영향을 받음을 알 수 있다. 특히 n 값에 관계없이 $\eta \leq 2.0$ 일 때 복호 성공 확률은 70%를 상회함을 알 수 있다.

그림 3은 네트워크의 총 노드 수가 $n = 100$ 일 때와 200일 때 복호율 η 에 따른 복호 연산량을 나타낸 것이다. 그림 3에서 BP 복호 방식의 복호 연산량은 소스 노드 수 k 값이 증가함에 따라 기

하급수적으로 증가함을 알 수 있는데, 이는 소스 노드의 수가 증가할수록 LT 부호의 길이가 길어지고 이에 따라 복호 연산량이 크게 증가하는데 원인이 있다.

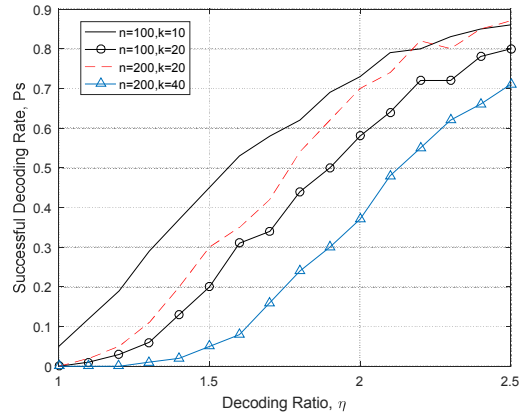


그림 2. 복호율 η 에 따른 복호 성공률 ($C1 = 1$)
Fig. 2. Successful decoding rate vs. η ($C1 = 1$)

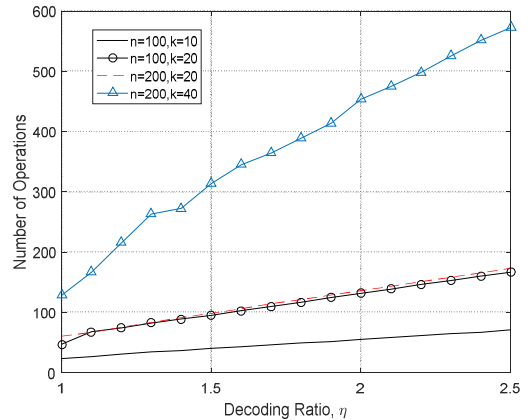


그림 3. 복호율 η 에 따른 연산량 ($C1 = 1$)
Fig. 3. Number of operations ($C1 = 1$)

5. 결론

본 논문에서는 노드 수가 $n = 100, 200$ 인 무선 센서 네트워크를 랜덤 기하 그래프로 모델링하여 데이터 분산 저장을 위한 분산 부호의 복호 성능을 시뮬레이션을 통해 분석하였다.

네트워크의 총 노드 수가 $n = 100$ 일 때와 200일

때 복호율 η 에 따른 복호 성공률은 노드 수 n 보다는 소스 노드 수 k 값에 따라 크게 영향을 받을 수 있었다. 특히 n 값에 관계없이 $\eta \leq 2.0$ 일 때 기존의 LTCDS 부호의 경우 약 60~70%의 성공률을 보이거나[4] BP 복호기의 복호 성공률은 70%를 상회함을 알 수 있다.

복호율 η 에 따른 복호 연산량을 살펴본 바, BP 복호 방식의 복호 연산량은 소스 노드 수 k 값이 증가함에 따라 기하급수적으로 증가함을 알 수 있었다. 이는 소스 노드의 수가 증가할수록 LT 부호의 길이가 길어지고 이에 따라 복호 연산량이 크게 증가하는데 원인이 있는 것으로 생각된다.

크기가 큰 무선 센서 네트워크의 노드 수는 일반적으로 수천 개 이상이므로 향후 n 의 값을 크게 증가시켜 시뮬레이션을 수행하는 것이 필요하다. 그림 3에서 확인한 바와 같이 n 값이 증가하여 이에 따른 k 값이 증가하면 연산량은 기하급수적으로 커지므로 복호 연산량을 줄일 수 있는 연구도 반드시 필요하다.

REFERENCES

- [1] Dejan Vukobratovič, Čedomir Stefanović, Vladimir Cmojevič, Francesco Romano Fantacci, "Rateless Packet Approach for Data Gathering in Wireless Sensor Networks," *IEEE Journal on Selected Areas in Communications* Volume: 28, Issue: 7, September 2010
- [2] Hoyoung Cheong, "On-the-Fly Belief Propagation Decoding of Decentralized Luby Transform Codes for Distributed Storage in a Small-Scale Wireless Sensor Network," *Journal of KIIECT*, Vol.9 No. 5, Dec. 2016, pp. 503-510.
- [3] M. Penrose, *Random Geometric Graphs*, New York: Oxford University Press, 2003.
- [4] Zhenning Kong, Salah A. Aly, Emina Soljanin, "Decentralized Coding Algorithms for Distributed Storage in Wireless Sensor Networks," *IEEE Journal on Selected Areas in Communications*, Volume: 28, Issue: 2, February 2010.
- [5] A. Kamra, V. Misra, J. Feldman, and D. Rubenstein, "Growth codes: Maximizing sensor network data persistence," in *Proc. of ACM Sigcom'06*, Pisa, Italy, September, 2006.
- [6] Samira Chouikhi, Inès El Korbja, Yacine Ghamri-Doudanec, Leila Azouz Saidane, "A Survey on Fault Tolerance in Small and Large Scale Wireless Sensor Networks," *Computer Communications*, Volume 69, 15 September 2015, pp. 22 - 37.
- [7] C. Avin and G. Ercal, "On the cover time of random geometric graphs," in *Proc. 32nd International Colloquium of Automata, Languages and Programming (ICALP'05)*, Lisboa, Portugal, pp. 677 - 689, July, 2005.
- [8] M. Luby, "LT Codes," *Proc. of the 43rd Annual IEEE Symp. on Foundations of Comp. Sc.*, pp 271-280, Vancouver, Canada, November 2002
- [9] Valerio Bioglio, Marco Grangetto, Rossano Gaeta, Matteo Sereno, "On the fly Gaussian Elimination for LT Codes," *IEEE Communications Letters*, Volume: 13, Issue: 12, December 2009, pp 935-955.
- [10] Hoyoung Cheong, "An Improved Belief Propagation Decoding for LT Codes," *Journal of KIIECT*, Vol.7 No. 4, Dec. 2014, pp. 223~228.
- [11] Yoo, Jinho, Choi, Sung-Gi, "Energy Efficient Routing Protocol for Mobile Wireless Sensor Networks," *Journal of KIIECT*, Vol.8 No. 5, Dec. 2015, pp. 405-411.

저자약력

정 호 영(Ho-Young Cheong)

[중신회원]



- 1987년 8월 : 연세대학교 대학원 전자공학과 (공학석사)
- 1995년 2월 : 연세대학교 대학원 전자공학과 (공학박사)
- 1995년 4월 ~ 현재 : 남서울대학교 정보통신공학과 교수

<관심분야>

채널부호, 분산 저장 부호, 무선 센서 네트워크