KIPS Trans. Comp. and Comm. Sys.
Vol.10, No.10 pp.269~276
ISSN: 2287-5891 (Print), ISSN 2734-049X (Online)

https://doi.org/10.3745/KTCCS.2021.10.10.269

Equal Energy Consumption Routing Protocol Algorithm Based on Q-Learning for Extending the Lifespan of Ad-Hoc Sensor Network

Kim Ki Sang† · Kim Sung Wook††

ABSTRACT

Recently, smart sensors are used in various environments, and the implementation of ad-hoc sensor networks (ASNs) is a hot research topic. Unfortunately, traditional sensor network routing algorithms focus on specific control issues, and they can't be directly applied to the ASN operation. In this paper, we propose a new routing protocol by using the Q-learning technology, Main challenge of proposed approach is to extend the life of ASNs through efficient energy allocation while obtaining the balanced system performance. The proposed method enhances the Q-learning effect by considering various environmental factors. When a transmission fails, node penalty is accumulated to increase the successful communication probability. Especially, each node stores the Q value of the adjacent node in its own Q table. Every time a data transfer is executed, the Q values are updated and accumulated to learn to select the optimal routing route. Simulation results confirm that the proposed method can choose an energy-efficient routing path, and gets an excellent network performance compared with the existing ASN routing protocols.

Keywords : Reinforcement Learning, Q-Learning, Ad-Hoc Sensornetwork, Energy Consumption

애드혹 센서 네트워크 수명 연장을 위한 Q-러닝 기반 에너지 균등 소비 라우팅 프로토콜 기법

김 기 상† · 김 승 욱††

요 약

최근 스마트 센서는 다양한 환경에서 사용되고 있으며, 애드혹 센서 네트워크 (ASN) 구현에 대한 연구가 활발하게 진행되고 있다. 그러나 기존 센서 네트워크 라우팅 알고리즘은 특정 제어 문제에 초점을 맞추며 ASN 작업에 직접 적용할 수 없는 문제점이 있다. 본 논문에서는 Q-learning 기술을 이용한 새로운 라우팅 프로토콜을 제안하는데, 제안된 접근 방식의 주요 과제는 균형 잡힌 시스템 성능을 확보하면서 효율적인 에너지 할당을 통해 ASN의 수명을 연장하는 것이다. 제안된 방법의 특징은 다양한 환경적 요인을 고려하여 Q-learning 효과를 높이며, 특히 각 노드는 인접 노드의 Q 값을 자체 Q 테이블에 저장하여 데이터 전송이 실행될 때마다 Q 값이 업데이트되고 누적되어 최적의 라우팅 경로를 선택하는 것이다. 시뮬레이션 결과 제안된 방법이 에너지 효율적인 라우팅 경로를 선택할 수 있으며 기존 ASN 라우팅 프로토콜에 비해 우수한 네트워크 성능을 얻을 수 있음을 확인하였다.

키워드 : 강화학습, Q-러닝, 애드혹 센서 네트워크, 에너지 소비

1. 서 론

최근 센서 네트워크 기술의 발전으로 장비는 갈수록 소형화되고 통신은 증가하여 무선통신기술을 통한 여러 사물 인터넷 기술들이 발전하고 있다. 이는 소형 센서 네트워크를 통해서 여러 가지 사업을 할 수 있다. 일례로, 철새들에게 소형 센서를 장착하여 새들의 움직임을 포착하고 관찰할 수도 있고 센서를 부착한 부표를 바다에 뿌려서 해양에서 벌어지는 일들에 대해서 정보수집을 할 수도 있다. 이처럼 다양한 산업 분야에서 활발하게 이용되고 있다. 그러나 깊은 산속, 정글 밀림, 사람의 손길이 닿지 않은 지역 등 다양한 환경의 특성상 여러 가지 위험요인이 산재해 있어 현재의 기술을 이용한 탐색을 수행하기에는 큰 어려움이 따른다. 이를 해결하기 위한 많은 연구가 수행되고 있고, 최근 센서 네트워크 기술의 발달을 통해 무선 애드혹 센서 네트워크(ASNs)에 대한 연구가 활발하게 진행되고 있다[1].

애드혹 네트워크는 크게 소스 노드, 중계 노드, 싱크 노드로 구성되어있다. 센서 네트워크는 넓은 지역을 탐지해야 하는데 무선 네트워크 특성상 각각 노드들은 광범위하게 배치되어 멀티 홉 통신을 수행하게 된다. 이는 탐지된 데이터가 싱크노드까지 도달하기 위해 중계되어야 하는 노드들의 양이 증가함을 의미한다. 기본적으로 무선 센서 네트워크 환경에서 네트워크 환경을 이루는 노드들은 배터리를 통해서 전력을 공급받는 경우가 많다. 현재 배터리 기술로 센서 네트워크

※ 본 연구는 과학기술정보통신부 및 정보통신기획평가원의 대학ICT연구센터 지원사업의 연구결과로 수행되었음(IITP-2021-2018-0-01799).
† 비 회 원 : Wonik IPS 연구개발 연구원
†† 종신회원 : 서강대학교 컴퓨터공학과 교수
Manuscript Received : January 18, 2021
First Revision : April 1, 2021
Accepted : April 15, 2021
* Corresponding Author : Kim Sung Wook(swkim01@sogang.ac.kr)

의 수명이 1년이라면 그 뒤로는 더는 센서를 활용할 수가 없게 된다. 하지만 그 이후에도 계속해서 주변 환경에 대해서 지속적인 확인이 되어야 한다. 이처럼 센서 네트워크는 한 번 설치하면 그 다음에 유지보수 하는 것이 사실상 불가능하다. 따라서 배터리를 최소한으로 소모하기 위해서 네트워크 자원을 효율적으로 이용해야 한다[1,2].

무선 네트워크에서는 제한된 통신 범위, 높은 전송 횟수, 잦은 전송지연으로 인해 데이터 손실이 자주 발생한다. 손실된 데이터를 복구하기 위해 노드들은 재전송을 수행해야 하며 이 과정에서 에너지 소모량이 증가하게 된다. 한정된 자원을 이용하여 네트워크를 구성해야 하고, 노드의 위치 변경이 많이 이루어지는 만큼 노드의 에너지 소모량 증가는 전체 네트워크의 수명감소로 직결되기 때문에 에너지의 중요성이 부각이 된다. 따라서 소스 노드로부터 싱크 노드까지 최적의 중계경로를 탐색하여 에너지 효율을 높이는 과정이 필요하다[3,4].

본 논문에서는 다양한 환경에서 제약조건이 많고 예측하기 힘든 에너지의 소모가 발생할 수 있는 네트워크 환경에서 에너지 효율을 높이고 전체 네트워크의 수명을 늘리기 위해 Q-러닝을 적용한 라우팅 기법을 제안한다. 네트워크의 구성 노드들은 이웃 노드들의 정보가 담긴 자신의 Q 테이블을 가지고 어떤 노드로 전송을 할지 선택하게 된다. 각 노드는 자신의 선택에 대해 보상을 받게 되고, 전송을 반복적으로 수행하며 데이터가 축적되면 최적의 전송경로를 찾을 수 있게 된다. 본 논문에서는 이를 통해 데이터의 전송확률을 높이고 전체 네트워크의 에너지 소모율을 균등하게 배분하여 결과적으로 네트워크의 수명을 증가시켰다. 앞서 언급했듯이 환경의 특성상 전송실패가 발생할 가능성이 크기 때문에 전송 실패에 대한 정보를 누적하고 학습시켜서 전송 성공확률을 증가시켰다.

본 논문의 구성은 다음과 같다. 2장에서는 애드혹 센서네트워크와 관련된 기존의 연구들을 소개하고 3장에서는 제안 기법의 기반이 되는 Q-러닝에 관해 설명한 뒤 제안된 알고리즘에 대해 세부적으로 설명한다. 4장에서는 성능평가를 통해 제안된 기법의 우수성을 검증한다. 마지막으로 5장에서 결론에 대해 논의한다.

2. 관련 연구

본 장에서는 ASN 환경에서 진행된 에너지 효율을 향상시키기 위한 라우팅 알고리즘 연구에 대해 살펴본다. 본 장의 구성은 다음과 같다. 네트워크 전체의 수명을 증가시키는 방법을 적용한 기법과 네트워크를 지역적으로 분할하여 라우팅을 수행하는 연구기법을 소개한다.

2.1 ASN의 전체 수명연장을 위해 제안된 라우팅 프로토콜

ASN 환경에서 네트워크 전체의 수명을 연장하기 위해 많은 연구들이 공통적으로 추구하고 있는 것은 불필요한 데이터의 재전송이나 에너지의 불균형 소비 등을 막는 것이다. 이를 해결하기 위해 네트워크의 전송 경로를 보다 효율적으로 설정, 혹은 재설정 하여 노드의 에너지 소모량을 감소시키거나 데이터의 전송 확률을 높이는 등 네트워크의 수명을 연장하기 위한 다양한 라우팅 프로토콜이 제안되고 있다.

1) QLEC 기법

QLEC 기법[6]은 노드의 잔여 에너지를 고려해서 분산 네트워크를 형성한 다음 네트워크의 수명을 연장하는 것을 목표로 한다. 일반적인 네트워크의 라우팅 프로토콜은 노드 수, 또는 에너지의 소비량을 기반으로 하여 최단 경로를 찾는 알고리즘을 이용하여 수명연장을 달성한다. 그러나 이러한 방법을 통해 라우팅 프로토콜이 구성되면 일정한 경로를 통해 반복적인 전송이 발생하여 특정 노드들의 사용량이 급격하게 증가하게 되고 이는 해당 노드의 수명 단축을 초래한다. 다른 노드들에 비해 에너지 소모량이 많아진 노드는 빠르게 작동을 멈추게 되고, 라우팅 보이드 (Routing Void)를 생성하게 되어 네트워크 운용에 차질이 생기게 된다.

보다 잔류 에너지를 균등하게 사용하기 위해서 QLEC 기법에서는 강화학습 기법 중 하나인 Q-러닝을 이용하였다. 각 노드는 주변 노드들의 잔류 에너지의 정보를 각자 테이블에 저장한 후 그중에서 가장 잔류 에너지가 많은 노드를 헤드로 선택한다. 이를 기반으로 주변에 노드들을 가지고 클러스터링을 형성한다. 클러스터링에 포함되지 않은 노드들은 가장 가까우면서 에너지가 많은 헤드 노드를 선택하게 된다. 헤드 노드는 경로를 탐색하기 위해서 각 헤드 노드의 정보를 Q 테이블에 저장한다. Q-러닝 기법을 이용한 경로 탐색을 수행한다. 보상함수~(reward function)를 통해 현재 노드의 선택 결과에 대한 보상을 책정하여 Q 테이블에 반영한다. 이를 반복적으로 수행하며 에너지 소비를 고르게 분배하여 수명을 연장을 달성할 수 있다[6].

2.2 지역화 기반 라우팅 프로토콜

지역화 기반 라우팅 프로토콜은 네트워크를 지역적으로 분할하여 라우팅을 수행하는 방식이다. 기본적으로 최적의 경로만을 탐색하는 방식으로 라우팅을 수행하기 때문에 전송효율이 높다는 장점이 있다. 그러나 경로 탐색에 포함되는 노드의 수가 많아질수록 효율이 떨어진다는 단점이 있다. 이를 해결하기 위해 중계 노드 탐색 범위를 제한하거나, 다음 홉 노드에 패킷이 도달할 수 있도록 전송에너지를 조절하는 등 다양한 기법이 연구되고 있다.

1) GeRaF

Geographic Random Forwarding 기법(GeRaF)[7]은 다음 홉 전달 노드를 선택하기 위해 RTS/CTS 메시지 메커니즘을 사용한다. 최적의 릴레이 노드를 선택하기 위해서 무작위 선택에 기반한 포워딩 기술을 사용하였다. 먼저, 짧은 거리와 이용 가능한 노드의 평균 수를 함수로 구한 다음 다양한 멀티 홉 통신 솔루션을 구한다. 그리고 성능을 평가해서 최고의 릴레이 노드 하나를 선택하는 방식이다. 패킷이 싱크 노드에 전달되지 않는 한 모든 홉에서 같은 프로세스가 반복된다. 단점으로는 모든 노드가 참여해야 정확한 측정이 되고 과도한 에너지 소비와 RTS/CTS 패킷으로 인한 긴 지연은 GeRaF 프로토콜의 주요 제한 사항이다.

3. 제안된 기법

본 장에서는 ASN 환경에서 데이터를 다음 홉으로 전달하기 위해 이웃 노드들을 선별하는 과정을 소개한다. 이웃 노드의 선발은 강화학습 기법 중 유명한 기법인 Q-러닝을 기반으로 수행하게 된다. 본 장에서는 우선 Q-러닝에 대한 소개를 한 후 제안하는 라우팅 기법을 소개한다.

3.1 강화학습과 Q-러닝 알고리즘

본 절에서는 제안하는 기법의 이론적 기반이 되는 강화학습(Reinforcement Learning)과 Q-러닝을 소개한다. 우선 강화학습의 기본 모델에 관해 설명하고 Q-러닝이 유도되는 과정을 수식을 통해 설명한다.

1) 강화학습(Reinforcement Learning)

강화학습은 경험을 통해 학습하는 인간의 학습방식을 기계에 적용하여 연구한 학습기법이다. 특정 에이전트(agent)가 특정한 환경 탐색할 때 특정 상황(state)에서 특정한 행동(action)을 취하여 얻을 수 있는 보상(reward)에 기반하여 보상을 극대화할 수 있는 행동의 정책(policy)을 정의하는 것이 목적이다. Fig. 1은 에이전트가 강화학습을 통해 정책을 찾아가는 과정에 대한 개념도이다[5].

무언가를 학습하기 위해서는 문제에 대한 정의가 필요하다. 강화학습에서는 이러한 문제를 에이전트의 환경으로 정의하며 이는 마르코프의 의사결정 프로세스(MDP)로 표현한다[5]. 일반적인 MDP는$[S,A,P,R]$ 총 4개의 인자로 구성되어있고, S는 상태(state), A는 행동(action), P는 상태 전이 확률(state transition probabilities), R은 보상(reward)의 집합을 의미한다. 상태 전이 확률 P는 $P^{a}_{s\to s'}$로 구성되며 이는 현재 상태S서 특정 행동A를 통해 다음 상태S'으로 이동할 확률을 의미한다. $R(s,a)$는 상태s에서 행동a 취했을 때 받을 수 있는 보상을 의미하고$R^{a}_{s\to s'}$는 현재 상태s서 특정 행동a를 통해 다음 상태s'으로 이동했을 때 받을 수 있는 보상을 의미한다. 시간t에서 $P^{a}_{s\to s'}$와 $R(s,a)$를 수학적으로 표현하면 Equation (1)과 같다.

$$\begin{cases} P^{a}_{s\to s'} = \Pr\left\{S_{t+1}=S' \mid_{s=s_t, a=a_t}\right\},\ s.t.,\ \sum_{s'\in s} P^{a}_{s\to s'} - 1 \\ R(s,a) = \left\{R_t(s_t,a_t)\mid_{s=s_t, a=a_t}\right\} = \sum_{s_{t+1}\in S}\left(P^{a_t}_{s\to s_{t+1}} \times R^{a_t}_{s_t\to s_{t+1}}\right) \end{cases} \tag{1}$$

강화학습에서 정책(policy)π는 상태$s\in S$에서 행동$a\in A(s)$를 취할 확률$\pi(s,a)$를 의미한다. 정책π에서 상태S의 값은 $V^{\pi}(S)$로 정의되며 이는 정책π에 속한 행동을 취했을 때의 기대 보상을 의미한다. 이를 수식으로 표현하면 Equation (2)와 같다.

$$V^{\pi}(S) = E_{\pi}\left\{R_t(s_{t,a_t})\right\} = E_{\pi}\left\{\sum_{k=0}^{\infty}\left(\gamma^{k}\times R_{t+k}(s_{t+k},a_{t+k})\right)\right\} \tag{2}$$
$$,s.t.,\gamma\in[0,1)$$

여기서$E_{\pi}\{\ \}$는 시간t에서 정책π에 속한 행동의 결과로 얻게 되는 보상의 총합을 의미하고, γ는 미래의 보상에 대한 discount factor이다. Bellman's optimality를 통해 Equation (2)를 Equation (3)과 같이 재귀방정식으로 재정의할 수 있다.

$$\begin{aligned} V^{*}(s) &= \max_{\pi} V^{\pi}(s) = \max_{\pi} E_{\pi}\left\{\sum_{k=0}^{\infty}\left(\gamma^{k}\times R_{t+k}(s_{t+k},a_{t+k})\right)\right\} \\ &= \max_{\pi} \mathrm{E}_{\pi}\left\{\mathrm{R_t}(\mathrm{s_{t,a_t}}) + \sum_{\mathrm{k=0}}^{\infty}\left(\gamma^{\mathrm{k}}\times \mathrm{R_{t+k}}(\mathrm{s_{t+k}},\mathrm{a_{t+k}})\right)\right\} \\ &= \max_{a}\left[R_t(s_{t,a_t}) + \left(\gamma\times\left(\sum_{s_{t+1}\in S}\left(P^{a_t}_{s_t\to s_{t+1}}\times V^{*}(s+1)\right)\right)\right)\right] \end{aligned} \tag{3}$$

3.2 Q-러닝 알고리즘

강화학습의 기법 중 가장 유명한 Q-러닝 기법은 특정한 모델이 없이 다양한 상황에 적용하여 사용할 수 있는 특징(Model-free)이 있다. 기본 모델 없이 최적의 정책을 산출할 수 있어 다양한 상황에서 유동적으로 사용 가능하다는 장점이 있다. Q-러닝은 Q값이라 불리는 상태-행동의 쌍$Q(s,a)$,을 기반으로 학습한다. 상태s 정책π에 속한 행동a를 통해 얻을 수 있는 값은 $Q^{\pi}(s,a)$로 정의되며 이를 수식으로 표현하면 Equation (4)와 같다[5].

$$\begin{aligned} Q^{\pi}(s,a) &= E_{\pi}\left\{\sum_{k=0}^{\infty}\left(\gamma^{k}\times R_{t+k}(s_{t+k},a_{t+k})\right)\right\} \\ &= R_t(s_{t,a_t}) + \left(\gamma\times\left(\sum_{s_{t+1}\in S}\left(P^{a_t}_{s_t\to s_{t+1}}\times V^{*}(s+1)\right)\right)\right) \end{aligned} \tag{4}$$

Equation (3)과 (4)의 식을 통해 Equation (5)의 값을 구할 수 있다.

$$\begin{aligned} V^{*(s)} &= \max_{a} Q^{*}(s,a) = \max_{a} Q^{*}(s_t,a_t) \\ &= \max_{a_t}\left[R_t(s_t,a_t) + \left(\gamma\times\left(\sum_{s_{t+1}\in S}\left(P^{a_t}_{s_t\to s_{t+1}}\times \max_{a} Q^{*}(s_{t+1},a)\right)\right)\right)\right] \end{aligned} \tag{5}$$

$Q^{*}(s_t,a_t)$는 상태s_t에서 행동a_t를 수행한 후 최적의 정책을 따랐을 때의 기대 보상을 의미한다. 이는 반복 수행을 통해 근사화될 수 있다.

$$Q(s_t,a_t)\leftarrow(1-\alpha)\,Q(s_t,a_t) + \alpha\left[R_t(s_t,a_t) + \gamma\max_{a} Q(s_{t+1},a)\right] \tag{6}$$
$$s.t.,\alpha\in(0,1]$$

여기서α는 learning rate로 Q값을 업데이트할 때 현재의 Q값과 미래의Q값 사이의 가중치로서 학습 속도를 조정하는데 사용된다.

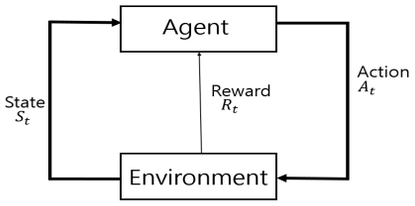


Fig. 1. Reinforcement Learning Model

3.3 애드혹 센서 네트워크 라우팅 프로토콜

애드혹 센서 네트워크 라우팅 프로토콜이 어떻게 작동하는지 자세하게 살펴본다. 제안된 프로토콜은 환경설정, 라우팅 알고리즘으로 구성되어 있다.

1) 애드혹 센서 네트워크 환경 알고리즘

환경설정 알고리즘의 핵심은 센서네트워크의 환경을 설정하는 것이다. 독립적인 제어를 위해서 각 센서노드들은 *Control_record*에 라우팅 정보를 개별적으로 관리한다. *Control_record*는 *Path_Cost*(*PC*)와 보상 기록 행렬($\mathbb{U}[\cdot]$)두 개의 파라미터로 구성되어있다. *PC*는 가까운 싱크노드까지의 통신 수행 능력의 정도를 나타내고, $\mathbb{U}[\cdot]$는 노드가 취한 행동의 보수가 기록된 행렬이다. *Control_record*의 정보는 노드들이 라우팅 경로를 결정하는데 이용된다.

*PC*는 가까운 싱크노드까지 라우팅을 수행할 때 발생하는 $Link\cos t(L_c)$를 통해 구한다. 두 센서 노드 사이의 링크 비용을 추정하기 위해 L_c를 정의한다. 각 센서노드에서 *PC*값은 현재 노드에서 싱크노드까지의 L_c의 총합으로 계산된다. 노드의 에너지 잔량과 에너지 소비율을 기반으로δ_i에서δ_j까지의 $L_c(i,j)$는 Equation (7)과 같이 구해진다.

$$L_c(i,j)=(1-\alpha)\left[(\alpha_i-\beta_i)\times\frac{d_{i,j}}{D_M}+\beta_i\times\left|\frac{v_{i,j}}{V_M}\right|\right] \quad (7)$$
$$+\alpha_i\left[(1-\gamma_i)\times(1-\frac{n_j}{N_j})+\gamma_i\times(1-\frac{\varepsilon_i}{E_M})\right]$$
$$s.t.,\begin{cases}\alpha_i=\dfrac{2\beta_i\gamma_i}{\beta_i+\gamma_i}\\ \beta_i=(1-\dfrac{d_{i,j}}{D_M})\\ \gamma_i=\dfrac{\varepsilon_i}{E_M}\end{cases}$$

여기서 $d_{(i,j)}$는 δ_j와δ_j사이의 거리이고, $v_{(i,j)}$는 δ_j와 δ_j 사이의 상대속도, n_j는 δ_j의 이웃 노드의 수, ε_i는 δ_i의 에너지 잔량을 의미한다. D_M과 V_M, E_M, N_j는 각각 센서 노드의 최대통신반경, 최대속도, 초기 에너지의 양, 의 이웃 노드의 최대 수를 의미한다.

본 논문에서는 실시간으로 변동하는 L_c를 추정하기 위해 각 파라미터들 사이에 $\alpha_i,\beta_i,\gamma_i$를 가중치로 부여한다. 현재 노드$\delta_i$와 다음 노드$\delta_j$사이의 거리가 가까울 경우, 노드 사이의 상대속도가 데이터 전송에 미치는 영향이 커지게 된다. 이러한 경우에는 노드 사이의 상대속도에 더 높은 가중치를 부여하기 위해 β_i에 높은 값을 부여하는 것이 적합하다. 반대로 노드 사이의 거리가 멀어질수록 상대속도가 미치는 영향이 줄어들기 때문에β_i의 밀도$\left(\frac{n_j}{N_j}\right)$의 영향을 크게 받지 않으며 전송이 가능하다. 이러한 경우 γ_i에 높은 값을 부여하여 에너지의 잔량에 더 높은 가중치를 두는 것이 적합하다. 반대로 δ_j의 밀도에 높은 가중치를 부여하는 것이 적합하다. β_i와γ_i의 조화 평균인 α_i를 거리와 속도, 에너지 잔량과 밀도 사이의 가중치로 부여하여 실시간으로 변화하는 환경을 반영한다.

네트워크 토폴로지를 구성하기 위해, 초기에는 각각의 개별노드에서 이웃노드들의 L_c값을 추정한다. 싱크 노드가 1홉 거리의 이웃 노드로 위치하고 있는 노드들은 해당 노드에서 싱크노드까지의 L_c값을 *PC*값으로 하고 PC_i을 Equation (8)과 같이 설정한다.

Algorithm 1.

1. if node δ_i δ_j 사이의 거리가 멀어지면,
   Then β_i의 값은 낮아진다.
   Else β_i의 값이 높아진다.
2. if node δ_j의 에너지 잔량이 낮다면,
   Then γ_i의 값이 낮아진다.
   Else γ_i의 값이 높아진다.
3. $L_c(i,j)=(1-\alpha)\left[(\alpha_i-\beta_i)\times\frac{d_{i,j}}{D_M}+\beta_i\times\left|\frac{v_{i,j}}{V_M}\right|\right]$
   $+\alpha_i\left[(1-\gamma_i)\times(1-\frac{n_j}{N_j})+\gamma_i\times(1-\frac{\varepsilon_i}{E_M})\right]$
   $s.t.,\begin{cases}\alpha_i=\frac{2\beta_i\gamma_i}{\beta_i+\gamma_i}\\ \beta_i=(1-\frac{d_{i,j}}{D_M})\\ \gamma_i=\frac{\varepsilon_i}{E_M}\end{cases}$
   식을 가지고 L_c를 계산한다.
4. if 선택한 노드가 싱크노드와 한-홉 거리에 있다면,
   Then 해당 노드의 *PC*값을 싱크노드와의 L_c값으로 설정한다.
   자신의 *PC*값을 이웃 노드들에 전파한다.
5. if 이웃 노드에서 *PC*값을 전달받았을 경우,
   Then 현재 노드의 L_c값에 이웃 노드의 *PC*값을 더해 자신의 *PC*값으로 저장한다.
6. Equation (8)을 이용하여 노드 δ_i의 *PC*값을 계산한다.
7. For(j = 1 ; j <= N_i ; j++)
   Equation (9)를 이용해서 이웃 노드의 *PC*값을 기반으로 노드 δ_i의 보상 기록행렬 $\mathbb{U}[\cdot]$을 업데이트 한다.

$$PC_i=\min_j\{PC_j+L_c(i,j)\}, s.t.\ , j\in N_i \quad (8)$$

여기서N_i는 δ_i의 이웃노드들의 집합이다. Equation (8)에서 계산된 이웃노드들의 *PC*값을 토대로 $\mathbb{U}[\cdot]$값이 Equation (9)와 같이 초기화된다.

$$\mathbb{U}_i[j]_{j\in N_i}=PC_j+L_c(i,j) \quad (9)$$

[Algorithm 1]은 환경설정 알고리즘을 pseudo code 로 나타낸 것이다. 1~2단계에서는 노드 δ_i와 δ_j사이의 거리를 통해 가중치 β_i를 구하고, 노드 δ_i의 에너지 잔량을 통해 가중치 γ_i를 구한다. 가중치 β_i와 γ_i의 조화평균으로 가중치 α_i를 구한 후, 3단계에서 Equation (7)의 식을 통해 노드 δ_i에서 δ_j까지의 $Link\cos t(L_c)$를 구한다. 4단계에서는 현재 노드가 싱크 노드와 1홉거리 이웃인지를 판별한다. 1홉 거리 이웃이라면 현재 노드에서 싱크노드까지의 L_c값을 *PC*값으로 설정한 후 그 값을 이웃 노드들에 전달한다. 5단계에서는 이웃 노드들에 *PC*값을 전달받을 경우, 현재노드로부터 해당 이웃노드까지의 L_c값과 전달받은 *PC*값을 더하여 현재 노드의 *PC*값을 설정한다. 6단계에서는 Equation (8)의 식을 통하여 노드 δ_i의 PC값을 결정한다. 7단계에서는 Equation (9)의 식을 통해 이웃 노드들의 PC값을 토대로 $\mathbb{U}[\cdot]$값을 초기화 한다.

2) 애드혹 센서 네트워크 라우팅 알고리즘

제안된 환경설정 알고리즘이 완료되면 애드혹 센서 네트워크의 가상 토폴로지는 싱크 노드를 뿌리로 하는 스패닝 트리

로 형성된다. 그러나 라우팅 작업 중 실시간으로 변화하는 환경에 의해 PC값과 $\mathbb{U}[\cdot]$값은 동적으로 변화하게 되며 이를 제어할 추가적인 라우팅 알고리즘이 필요하다.

애드혹 센서 네트워크의 라우팅 알고리즘을 설계하기 위해 Multi-player Markov Decision Process(MMDP)를 고려한다. 각각의 센서노드들은 플레이어의 역할을 수행하고, 플레이어 $i(\delta_i)$는 최적의 라우팅 $\psi_i^* \in \mathbb{A}_i$를 찾는다. 여기서 $\mathbb{A}_i$는 노드 δ_i의 행동(action)들의 집합으로 $|\mathbb{A}_i| = |N_i|$의 관계가 형성되고, ψ_i^*는 1홉 거리에서 이웃 노드들 중 하나의 노드를 선택하는 라우팅 정책을 의미한다. 시간 t에서 노드 δ_i의 상태와 행동은 $s_i(t)$와 $a_i(t)$로 정의할 수 있으며 각각 $s_i(t) \in \mathbb{S}_i, a_i(t) \in \mathbb{A}_i$ 관계를 갖게 된다. $U_i(s_i(t), a_i(t))$는 시간 t에서 노드 δ_i의 상태가 $s_i(t)$일 때 행동 $a_i(t)$를 취했을 때 발생하는 δ_i의 보상을 의미한다. $U_i(s,a)$ 함수는 노드 δ_i의 동작 및 성능을 결정하기 때문에 Q-러닝의 학습에 중요한 역할을 수행한다.

Q-러닝을 적용한 ASN 라우팅 알고리즘의 최종 목표는 패킷을 최대 보상으로 싱크 노드에 전달하는 것이다. $a_i(t) = a_i^m$는 시간 t에서 m번째 이웃 노드를 선택하는 행동을 의미하고, 이때의 $\mathbb{U}_i[\cdot]$는 Equation (10)과 같다.

$$
\begin{aligned}
&U_i(s_i(t), a_i(t) = a_i^m) = R_i(s_i(t), a_i^m) \\
&= \frac{1}{PC_m^t + L_c(i,m)} \times a_i^m \left(\prod_{t-\Delta t}^{t} S_i^m(t) \right) \\
&\text{s.t.}, S_i^m(t) \begin{cases} 1, \text{액션} a_i^m \text{을통해전송이성공하였을경우} \\ 1, \text{액션} a_i^m \text{을통해중계노드로선택되지않았을경우} \\ 0.9, \text{액션} a_i^m \text{을통해전송이실패하였을경우} \end{cases}
\end{aligned} \tag{10}
$$

여기서 PC_m^t는 시간 t에서 m번째 이웃 노드의 PC값을 의미하고, $S_i^m(t)$는 a_i^m의 결과(전송이 성공했는지, 실패 했는지)를 의미한다. m번째 이웃 노드의 PC값에 δ_i와 δ_m의 L_c값을 더하면 δ_i부터 싱크노드까지의 총 전송 비용을 구할 수 있게 된다. 따라서 $PC_m^t + L_{c(i,m)}$의 값은 낮을수록 높은 전송효율을 갖게 된다. a_i^m의 전송 성공여부를 누적으로 학습하기 위해 $t-\Delta t$ 시간부터 t시간까지의 이웃노드들의 전송 성공여부를 $S_i^m(t)$에 저장한다. 이때 a_i^m로 선택되어 전송이 성공하였을 경우엔 1, a_i^m로 선택되지 않았을 경우엔 1의 보상을 부여하고 전송이 실패하였을 경우엔 0.9의 페널티를 부여한다. 누적된 성공확률을 보상에 반영하여 노드들을 정확하게 학습시킬 수 있다.

Q-러닝 알고리즘은 기본적으로 $greedy$알고리즘을 기반으로 동작한다. 가장 높은 Q값을 갖는 노드를 선택하는 행동을 취해 가장 높은 보상을 받는 것을 목표로 한다. 이는 싱크노드까지 패킷을 전달할 때 최적의 경로를 선택하는데 도움이 된다[8].

효과적으로 Q-러닝 알고리즘을 발전시키기 위해서는 $\varepsilon-greedy$ 정책을 적용해야 한다. $\varepsilon-greedy$정책이란 답을 얻기 위해서는 충분하게 경험해보는 것이 필요한데 단순히 $greedy$알고리즘을 통해 선택할 경우에는 얻은 답이 최적의 해답이라고 장담할 수 없기 때문에 ε값이라는 일정한 모험의 요소를 추가한 것이다. ε값을 정의하는 것은 확률 결정 문제이다. 상태 s에서 $\varepsilon-greedy$정책 π를 통해 행동 a를 선택할 확률은 $\pi(a|s)$로 정의된다.

Algorithm 2.

1. Equation (10)을 이용하여, 시간 t에서 m번째 노드를 중계노드로 선택했을 때의 보상을 계산한다.
2. if 액션 a_i^m을 통해 전달된 패킷이 전송에 실패하였을 경우, Then Equation (10)의 $S_i^m(t)$를 이용하여 $\mathbb{U}_i[\cdot]$에 패널티를 부여한다.
3. 계산된 유틸리티 $\mathbb{U}_i[\cdot]$를 Equation (6)에 대입하여 이웃노드들의 Q값을 계산한다.
4. if 효과적으로 Q-러닝 알고리즘을 발전시키기 위해서, $\varepsilon-greedy$ 정책을 적용한다. 정책은 Equation (11)번을 통해 정의된다.
5. 계산된 유틸리티$\mathbb{U}_i[\cdot]$와 $\varepsilon-greedy$정책을 기반으로 다음 중계노드를 선택하여 데이터를 전송한다.

멀티 센서 라우팅 모델을 통해 각 센서 노드는 현재 ASN의 시스템 상황을 학습하여 $\pi(a|s)$를 통해 릴레이노드가 선택될 확률을 결정한다. $\pi(a|s)$는 Equation (11)과 같이 정의된다.

$$
\pi(a|s) = \begin{cases} \dfrac{\varepsilon}{m} + 1 - \varepsilon & \text{if } a^* = \underset{a \in A}{\arg max} Q(s,a) \\ \dfrac{\varepsilon}{m} & otherwise \end{cases} \tag{11}
$$

여기서 m은 액션으로 선택하는 노드를 방문한 횟수를 의미하고 ε값은 최적의 노드를 제외한 나머지 노드를 랜덤 하게 선택할 수 있도록 주어진 상수 값이다.

[알고리즘 2]는 라우팅 알고리즘을 pseudo code로 나타낸 것이다. 1단계에서는 Equation (1)의 식을 이용하여 노드 δ_i의 m번째 이웃노드가 시간 t에서 릴레이노드로 선택되었을 때의 δ_i의 보상$\mathbb{U}[\cdot]$을 계산한다. 2단계에서는 시간 t에서 노드 δ_i의 행동 a_i^m의 결과가 성공했는지 실패했는지를 기록한 $S_i^m(t)$를 통해 실패한 전송에 대한 페널티를 부여한다. 3단계에서는 Equation (10)을 통해 획득한 유틸리티 $\mathbb{U}[\cdot]$를 Equation (6)에 대입하여 이웃노드들의 Q 값을 계산한다. 4단계에서는 Q-러닝 알고리즘을 효과적으로 발전시키기 위해 Equation (11)을 이용하여 $\varepsilon-greedy$정책을 설정한다. 5단계에서는 계산된 유틸리티 $\mathbb{U}_i[\cdot]$와 $\varepsilon-greedy$정책을 기반으로 다음 중계노드를 선택하여 데이터를 전송한다.

3.4 Q-러닝 알고리즘

본 논문에서는 Q-러닝 알고리즘을 적용한 새로운 라우팅 알고리즘을 제안한다. Q-러닝 기반의 라우팅 모델을 사용하여 시시각각으로 변화하는 상황을 기록, 수집한다. 따라서 센서 노드는 현재 상황을 개별적으로 학습하고 최상의 라우팅 경로를 찾아낸다. 현재의 환경 조건은 동적으로 변화하므로 라우팅 작업 중에 각 노드는 주기적으로 라우팅 정보를 업데이트하고 현재 전략을 재평가한 후 그에 따라 패킷 전달을 위해 인접 노드 중 하나를 선택한다. 이러한 절차를 통해 네트워크 전체의 전송 효율을 높일 수 있고 결과적으로 네트워크의 수명을 연장할 수 있다. 제안된 라우팅 알고리즘의 주요한 단계는 다음과 같다.

Step1: 싱크 노드가 $setup$메시지를 브로드캐스팅 한다. 메시지를 수신한 센서 노드는 Equation (7)의 식에 따라 L_c를 개별적으로 추정한 후 $setup$메시지를 이웃 노드들에 전파한다.

Step2: 노드가 다수의 $setup$메시지를 수신하면, 이 노드는 싱크 노드에 도달하기 위해 이들 중 하나를 선택하고 Equation (8)과 (9)의 식에 따라 PC값과 $\mathbb{U}[\cdot]$를 초기화한다.

Step3: 라우팅 결정 순간 마다, 각각의 센서노드들은 계산된 유틸리티$\mathbb{U}_i[\cdot]$와 $\varepsilon-greedy$정책에 따라 통계적으로 다음 노드를 선택한다.

Step4: 선택된 라우팅 전략을 기반으로 각 센서 노드들의 $\mathbb{U}[\cdot]$벡터값이 실시간으로 수정된다.

Step5: 일정한 시간마다, 각각의 센서노드들의 PC,$\mathbb{U}[\cdot]$ 및 $P(\cdot)$값은 Equation (8), (9) 및 (11)의 식에 의해 주기적으로 재추정된다.

Step6: 시간 에서 각 노드들의 라우팅 전략은 Q-learning mechanism을 사용하여 Equation (6)의 식에 따라 선택된다.

Step7: 자신의 데이터를 싱크 노드로 전달하기 위해 각 노드들은 Step1~6을 반복적으로 수행한다.

Step8: 반복적인 라우팅 절차를 통해 노드들은 개별적으로 학습하고 이를 통해 최적의 라우팅 경로를 선택하여 네트워크 전체적으로 균형 잡힌 시스템 성능을 달성한다.

4. 성능 평가

이번 장에서는 본 논문에서 제안하는 Q-러닝 기반의 애드혹 센서 네트워크 라우팅 프로토콜의 성능을 시뮬레이션을 통해 기존에 존재하는 타 기법들과의 차이를 비교한다. 이를 통해 본 논문에서 제안하는 기법의 성능을 평가한다. 시뮬레이션은 C++ 을 이용하여 개발한 시뮬레이터를 이용하였다. 성능평가를 위한 네트워크 환경은 다음과 같다.

Table 1. System Parameters는 성능 평가를 위해 가정한 네트워크 환경 및 매개변수이다. 각 실험의 성능 평가 기준은 한 번 전송에 참여하는 평균 노드의 수, 라우팅 성공률(Routing Success Rate: RSR), 전체 네트워크의 평균 수명(Average Lifetime of entire Network: ALN) 등에 대해 비교하였다.

QASN(Q-LearningAd-hocSensorNetwork)의 성능 분석을 위해 기존의 지리적 라우팅의 기법인 GeRaF[7]기법과

Table 1. System Parameters

System parameters	Value
number of node	200
node of Transmission radius	50m
node of energy	2000000
maximum neighbor nodes	30
network size	200m×200m
number of experiment	100
Probability of failure	$1-\frac{d_{i,j}}{D_M}$
ε value	0.5
ε decay	$\frac{1}{m}$

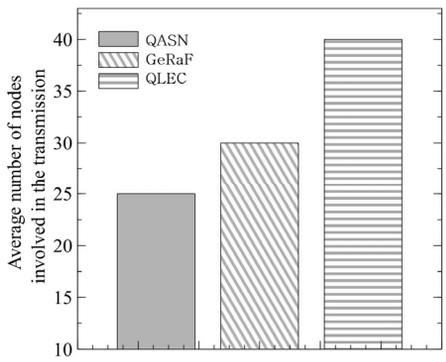


Fig. 2. Average Number of Nodes Involved in the Transmission

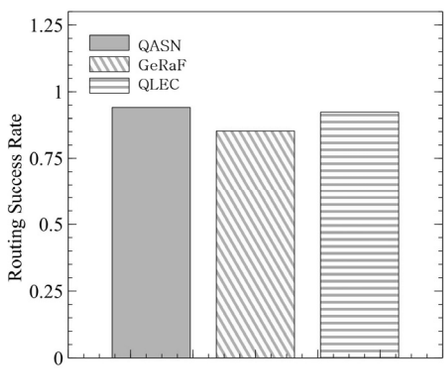


Fig. 3. Routing Success Rate

Q-러닝 기반의 기법인 QLEC[6]기법을 통해 라우팅 성공률과 평균 수명에 대해 비교하였다. 각 실험은 100회 반복 시뮬레이션을 수행한 후 그 평균값을 사용하였다.

Fig. 2는 소스 노드부터 싱크 노드까지 데이터를 전송할 때 한 라운드 전송에 참여하는 노드의 수이다. 위의 그래프는 제안된 기법과 GeRaF기법과 QLEC기법의 평균 노드의 수를 보여준다. QLEC과 GeRaF기법은 평균 35개와 30개 정도로 비슷한 결과를 보였다. 그 이유는 QLEC 기법은 클러스터링 알고리즘을 이용하기 때문에 하나의 헤드 노드가 포함하고 있는 노드의 수가 많으므로 전송에 참여하는 노드의 수가 다른 기법에 비해 많아서 라우팅에 GeRaF 기법은 브로드 캐스트 기반으로 데이터를 전송을 수행하기 때문에 평균 노드의 수가 많다. 반면 제안된 기법은 에너지의 잔량과 밀도 등 복합적인 요소를 고려하여 최대 30개의 노드만을 참여노드로 구성하였다.

Fig. 3은 제안된 기법과 GeRaF 기법과 QLEC기법의 전체 전송 횟수 중 소스 노드부터 싱크 노드까지 라우팅을 성공한 전송의 비율을 나타낸다. 실험 결과 제안된 기법은 GeRaF 기법과 비교하면 약 8.89%가량, QLEC기법에 비해서는 약 1.75% 더 높은 성공률을 보였다. 전송실패에 대한 별다른 대처가 없는 GeRaF 기법은 잦은 전송지연과 데이터 유실이 발생하는 넓은 환경의 특성상 라우팅 성공률이 상대적으로 떨어진다. 또한 라우팅 내에 중계 노드의 수가 적어질수록 라우팅 성공률이 낮아지게 된다. QLEC 기법의 경우 클러스터링을 형성하는 특성상 라우팅 성공확률이 상대적으로 높다. 하지만 클러스터링에서 헤드 노드의 전송량이 많으므로 에너지

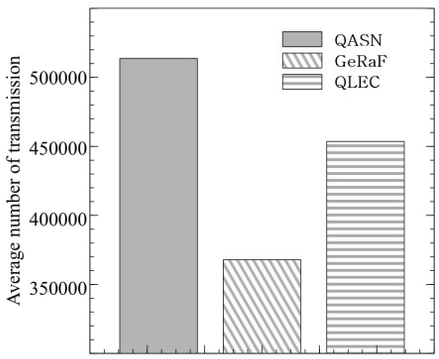


Fig. 4. Average Number of Transmission

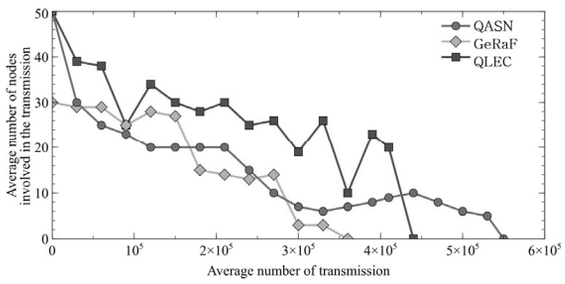


Fig. 5. Average Number of Nodes Participating Nodes

소비가 편중된다. 데이터 전송이 실패하였을 경우 해당 노드의 Q 값을 낮추는 페널티를 부여한 후 재전송을 수행한다. 중계 실패 노드의 Q 값이 다른 노드의 Q 값보다 낮아질 때까지 반복적으로 수행하기 때문에 에너지의 낭비가 발생한다는 단점 또한 있다. 제안된 기법은 전송실패가 발생한 노드에 대해 Equation (10) 을 이용하여 페널티를 부여한다. 현재 시점에서부터 다섯 번의 전송에 대한 실패를 누적하여 적용하므로 전송실패노드를 중계 노드로 선택할 확률이 기하급수적으로 낮아진다. 위 실험을 통해 제안된 기법은 GeRaF 기법에 비해 높은 라우팅 성공률을 갖는 것을 확인할 수 있다.

Fig. 4는 데이터 전송 시작부터 네트워크 종료할 때까지 제안된 기법과 GeRaF 기법과 QLEC기법의 평균 전송 횟수를 나타낸다. 실험 결과 제안된 기법은 GeRaF 기법과 비교하면 26.35%,QLEC기법에 비해 10.93%만큼 증가한 횟수로 전송을 수행하였다. GeRaF 기법의 경우에는 에너지 효율을 중시하는 라우팅을 하므로 정해진 중계 노드만을 전송에 참여하므로 라우팅 내의 중계 노드들의 수명이 다하면 다른 노드의 수명과 상관없이 네트워크가 종료된다. 네트워크의 밀도에 따라 변하는 특성을 보여주고 있다. 밀도가 높으면 라우팅 노드의 수가 증가하여 단거리 통신의 효율적 통신이 가능하지만 밀도가 낮으면 노드 간의 수가 적어지기 때문에 통신에 들어가는 에너지량이 높아진다. 따라서 다른 기법들에 비해 전송 횟수가 현저하게 낮게 나타난다. QLEC 기법은 클러스터링 형성을 통해서 헤드 노드 간의 통신을 한다. 이렇게 되면 헤드 노드의 자원이 참여하는 자원보다 낮아지게 되면 헤드 노드를 교체하게 되고 헤드 노드는 다시 그룹에 속한 노드가 된다. QLEC기법은 GeRaF의 기법보다 높은 전송 횟수를 보여준다. 하지만 헤드 노드의 변경과 클러스터링을 형성해야 하는 과정에서 소모되는 에너지의 양이 많다. 형성된 클러스터 헤더 간의 네트워크 간에 재전송으로 인한 에너지 소모량이 많아 특정 노드의 수명이 빠르게 줄어든다는 단점이 있다.

Fig. 5는 시뮬레이션을 반복 수행할 때 전송에 참여하는 노드의 수의 평균의 변화량을 나타낸다. 시간의 흐름은 데이터의 전송 횟수로 대체하여 실험을 진행하였다. GeRaF 기법은 지리적 라우팅과 EPA라는 패킷의 이득, 신뢰성 및 에너지 소비의 균형을 맞추는 단위 에너지 소비당 예상되는 패킷 향상 값을 통해 효율적인 경로를 탐색하는 라우팅 프로토콜이다. 전송 초기 단계부터 최적의 경로를 찾아가기 때문에 제안된 기법과 QLEC기법보다 적은 중계 노드를 갖게 된다. 시간의 흐름에 따라 제안된 기법과 QLEC기법은 학습을 통해 최적의 경로를 찾아가게 되어 포함하는 중계 노드의 수가 줄어들게 되고 GeRaF 기법의 경우 EPA의 상한선 내에 효율적인 노드의 수가 거리에 따라서 적어지게 된다. 노드들의 에너지가 적어지게 되면 통신에 들어가는 에너지가 많아지게 되고 효율이 낮아진 노드들은 통신에 참여할 수 없게 된다. GeRaF의 경우에는 밀도가 높은 경우의 통신에 적합하기에 평균 수명이 짧은 단점이 있다. QLEC기법은 노드 간의 클러스터링을 형성하고 클러스터링에서 가장 에너지가 높은 노드를 헤드 노드로 선정하게 된다. 그리고 클러스터링에서는 에너지의 잔량을 가지고 통신을 하게 된다. 헤드 노드들은 중계 노드로써 헤드 노드 간의 통신을 통해서 싱크노드까지 통신을 하게 된다. 에너지 잔량을 기반으로 헤드 노드를 선택할 경우 한 홉 간 전송의 에너지 효율은 높아질 수 있지만 싱크노드까지 완성된 최적의 라우팅 경로에 포함된 헤드 노드의 수가 많아질 수 있다는 문제점이 발생한다. 제안된 기법은 다음 중계 노드를 선택할 때 에너지의 잔량 뿐만 아니라 노드 간 거리, 중계 노드의 밀도 등 다양한 환경변수를 고려하여 더욱 효과적인 학습 체계를 구축하였다. 이를 통해 제안된 기법은 데이터를 1회 전송할 때 최소한의 노드를 이용하여 높은 에너지 효율을 달성하였다. 시뮬레이션의 결과 제안된 기법은 QLEC 기법보다 약 10.9% 높은 전송 횟수를 달성하여 네트워크의 수명 연장을 효과적으로 달성하였음을 확인할 수 있었다.

5. 결 론

ASN 라우팅 체계는 특정 제어 문제에 집중되어 있다. 그러나 이러한 점 때문에 균형 잡힌 시스템 성능을 얻는 것이 매우 어렵고 부적합하다. 환경적인 제약요소가 많고 자연적인 장애물로 인해 다양한 위치에 센서 노드들이 배치되지 못하여 불필요한 에너지 소비와 전송 지연 등이 발생하게 된다. 물리적인 관여가 어려운 환경의 특성상 균형 잡힌 시스템 성능은 시스템 전체의 수명에 관여하여 매우 중요하게 작용한다. 따라서 전체 네트워크의 수명을 증가시키기 위해서는 각 노드에서 소모되는 에너지의 양을 균등하게 배분하는 방법이 필요하다.

본 논문에서는 효율적인 에너지 소비를 통해 애드혹 센서 네트워크의 수명을 연장하기 위한 Q-러닝을 이용한 라우팅

프로토콜을 제안하였다. 또한 전송 실패 노드에 대한 페널티를 누적으로 부과하여 실패 노드를 회피할 확률을 높였고 에너지, 거리 등 다양한 환경요소를 고려하여 학습 효과를 높이는 기법을 제안하였다. 시뮬레이션을 통해 제안된 기법은 GeRaF 기법과 비교하면 약 8.89%가량, QLEC기법에 비해서는 약 1.75% 더 높은 성공률을 보였다. 전송 횟수에 대해서 비교하면 제안된 기법은 GeRaF 기법과 비교하면 26.35%, QLEC 기법에 비해 10.93%만큼 증가한 횟수로 전송을 수행하여 네트워크 수명 증가를 달성하였음을 확인할 수 있었다. 제안된 기법은 네트워크 내의 생존 노드의 수가 감소하여도 노드들의 잔여 에너지량을 검토하여 최적의 경로를 탐색함으로써 높은 라우팅 효율을 달성했음을 알 수 있었다. 또한 정책을 통해 시간이 지남에 따라 높은 학습효율을 달성하여 노드들의 에너지 소비율을 균등하게 배분해 전체 네트워크의 수명 연장을 달성한 것을 확인하고 기존의 기법에 비해 더욱 효율적인 라우팅 기법임을 증명하였다.

References

[1] Q. Sang, H. Wu, L. Xing, H. Ma, and P. Xie, "An energy-efficient opportunistic routing protocol based on trajectory prediction for FANETs," in *IEEE Access*, Vol.8, pp.192009-192020, 2020,.

[2] R. Priyadarshi, L. Singh, A. Singh, and A. Thakur, "SEEN: Stable energy efficient network for wireless sensor network," *2018 5th International Conference on Signal Processing and Integrated Networks (SPIN)*, Noida, India, pp.338-342, 2018.

[3] B. S. Rani and K. Shyamala, "Energy efficient load balancing approach for multipath routing protocol in Ad Hoc networks," *2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP)*, Gangtok, India, pp.1-5, 2019.

[4] S. Din, A. Paul, A. Ahmad, and J. Kim, "Energy efficient topology management scheme based on clustering technique for software defined wireless sensor network," *Peer-to-Peer Networking and Applications Volume*, Vol.12, No.2, pp.348-356, 2019.

[5] R. S. Sutton and A. G. Barto, "Reinforcement learning: An introduction," MIT Press, 2018.

[6] K. Li, H. Huang, X. Gao, F. Wu, and G. Chen, "QLEC: A machine-learning-based energy-efficient clustering algorithm to prolong network lifespan for IoT in high-dimensional space," In *Proceedings of the 48th International Conference on Parallel Processing (ICPP 2019)*, Association for Computing Machinery, pp.1-10, 2019.

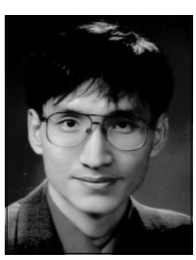
[7] M. Zorzi and R. R. Rao, "Geographic random forwarding (GeRaF) for ad hoc and sensor networks: Multihop performance," in *IEEE Transactions on Mobile Computing*, Vol.2, No.4, pp.337-348, 2003.

[8] J. Bahi, W. Elghazel, C. Guyeux, M. Hakem, K. Medjaher and N. Zerhouni, "Reliable diagnostics using wireless sensor networks," *Computers in Industry*, Vol.104, pp.103-115, 2019.

[9] Z. Mammeri, "Reinforcement learning based routing in networks: Review and classification of approaches," *IEEE Access*, Vol.7, pp.55916-55950, 2019.

김 기 상

https://orcid.org/0000-0002-2206-6014
e-mail : thirdson87@sogang.ac.kr
2019년 상명대학교 컴퓨터공학과(학사)
2021년 서강대학교 컴퓨터공학과(석사)
2021년 ~ 현 재 Wonik IPS 연구개발 연구원
관심분야 : 강화학습을 이용한 네트워크



김 승 욱

https://orcid.org/0000-0003-1967-151X
e-mail : swkim01@sogang.ac.kr
1993년 서강대학교 전자(학사)
1995년 서강대학교 전자(석사)
2003년 Syracuse University, Computer Science(박사)
2005년 중앙대학교 컴퓨터공학부 조교수
2006년 ~ 현 재 서강대학교 컴퓨터공학과 교수
관심분야 : Front-end Design & Verification Methodology