Journal of Institute of Control, Robotics and Systems (2014) 20(10):1079-1083
http://dx.doi.org/10.5302/J.ICROS.2014.14.8015 ISSN:1976-5622 eISSN:2233-4335

곤충 행동 양식 기반의 애드 혹 네트워크를 위한 존 반경 결정 알고리즘

Insect-Inspired Algorithm for Zone Radius Determination of Ad-hoc Networks

이 혜 민, 김 동 성*
(Hea-Min Lee[1] and Dong-Seong Kim[1,*])
[1]Dept. of Electronic Engineering, Kumoh National Institute of Technology

Abstract: In this paper, a new zone radius determination algorithm is proposed for a nature-inspired routing protocol that emulates the foraging behavior of bees based on their ability to find an optimal route from nectar sites. Instead of changing the radius of nodes one-hop by one-hop, the proposed algorithm alters the radius of nodes as gaps of another radius and adapt quickly to network conditions. The simulation results show that the proposed algorithm has higher efficiency compared with existing studies in an aspect of computational complexity and end-to-end delay.

Keywords: Insect-inspired algorithm, Zone routing protocol, Ad Hoc network, IZRP, Zone radius determination

I. 서론

동적으로 변하는 무선망에서 안정된 데이터 전송을 보장하기 위해 Mobile Ad hoc Network (MANET)에 대한 다양한 연구들이 진행되어 왔다[1-5].

애드 혹 통신의 네트워크 환경은 존 반경안의 노드들의 수가 빈번하게 바뀜에 따라 불안정성을 보인다. 이러한 문제점들을 고려해 존 라우팅 프로토콜의 효율성을 개선시키기 위한 방안으로 독립적인 존 라우팅 프로토콜 IZRP (Independent Zone Routing Protocol)이 연구되었다[6].

본 논문에서는 MANET에서 수렴 시간을 단축시켜 네트워크 연결성을 높이고 계산적인 효율을 높이기 위해 꿀을 찾는 벌의 행동방식을 모방하는 존 반경 결정 알고리즘을 제안한다. 기존의 IZRP 알고리즘은 최소 탐색 방법에 따라 한 홉씩 존 반경을 증가시키기 때문에 새로운 최적의 존 반경을 빠르게 결정 할 수 없어 지연이 길다는 문제점이 있다. 이를 수정한 제시한 알고리즘은 꿀을 찾는 벌에 근거하며 대기벌이 이전의 꿀이 있는 지역을 현재의 지역과 비교하여 꿀의 확보율이 높은 장소로 빠르게 이동하는 행동양식을 IZRP에 반영하였다.

본 논문의 구성은 다음과 같다. 서론에 이어 II 장에서는 관련연구를 소개하고 문제점을 제시한다. III 장에서는 벌의 행동 양식을 분석하고 기존의 IZRP의 단점을 보안한 새로운 알고리즘을 제안한다. IV 장에서는 제안된 방법의 효용성을 증명하기 위해서 모의실험을 통한 성능 평가를 하고, 마지막 V 장에서는 결론 및 향후 연구에 대해 논의한다.

* Corresponding Author
Manuscript received July 7, 2014 / revised September 10, 2014 / accepted September 15, 2014
이혜민, 김동성: 금오공과대학교 전자공학부
(lhm0407@kumoh.ac.kr/dskim@kumoh.ac.kr)
※ 본 연구는 미래창조과학부 및 정보통신산업진흥원의 "창조ICT 융합인재양성사업" 및 지역혁신인력양성사업 (NRF-2012H1B8A2026109)으로 수행된 연구결과임
※ 이 논문은 2014 제 29회 ICROS 학술대회에 초안이 발표되었음[11].

II. 문제점 분석

IZRP는 최소 탐색 방법(min traffic search scheme)과 적응 트래픽 측정 방법(adaptive traffic estimation scheme)을 조합한 방식이다. 또한 이 방식은 분산된 노드들의 존 반경을 각각 변경한다[6-8].

IZRP의 최소 탐색 방법은 한 홉씩 존 반경을 증가시키거나 감소시켜 최소한의 라우팅 트래픽을 유지하며 탐색한다. 측정 간격 동안, 노드를 통과하는 라우팅 트래픽이 측정되고, 만약 현재 측정되는 트래픽이 이전보다 적으면 존 반경은 증가시킨다. 반대로 측정된 라우팅 트래픽이 이전에 측정된 것보다 크면 존 반경은 감소된다. 이 과정은 트래픽을 최소로 유지하기 위한 최적의 존 반경을 찾을 때 까지 반복 된다.

하지만 기존의 IZRP은 최소 탐색 방법에 따라 최소 트래픽을 유지하기 위해 한 홉씩 존 반경을 증가시키기 때문에 새로운 최적의 존 반경으로 빠르게 변할 수 없어 지연이 길다는 문제점이 발생한다.

무선 센서 네트워크에서 송신지 노드와 수신지 노드 사이에 라우팅의 성능을 높이기 위해 벌들의 꿀을 찾는 방식에서 착안된 ABC (Artificial Bee Colony) 알고리즘이 연구되었다[9]. 본 연구에서는 이를 수정해 기존의 IZRP에 적용하여 종단 지연을 낮춰 네트워크의 연결성을 높이고자한다.

III. 벌의 행동을 적용한 존 반경 결정 알고리즘

본 절에서는 꿀벌이 먹이를 찾는 행동을 수학적으로 분석하여 모델링한 ABC 알고리즘을 기존의 존 반경 결정 알고리즘에 적용하고자 한다.

Copyright© ICROS 2014

표 1. ABC알고리즘을 설명하기 위해 설정된 기호.
Table 1. Bee-Inspired Algorithm Notations.

파라미터	정의
E_b	일벌의 수
O_b	대기벌의 수
S_b	탐색벌의 수
i	꿀이 있는 장소의 위치 번호
F_i	i지역에서 확보율
P_i	i지역을 대기벌이 꿀 확보율이 높은 지역을 선택할 가능성
L_i	꿀 확보율이 높은 위치나 방향
L_{ij}^{curr}	현재 꿀이 있는 지역의 위치
L_{ij}^{anew}	새로운 꿀이 있는 지역의 위치
L_{ij}^{old}	과거의 꿀이 있는 지역의 위치
O_b^{curr}	현재 대기벌의 수
O_b^{prev}	이전의 대기벌의 수
MCM	최대 순환 숫자

1. 벌의 행동을 IZRP에 적용하는 과정

ABC 알고리즘은 Karaboga에 의해 소개된 휴리스틱(Heuristic) 알고리즘이다[9]. 이 알고리즘을 소개하기 위해 설정된 기호를 표 1에 나타냈다.

ABC 알고리즘에서 일벌, 대기벌, 탐색벌이 있다. 전형적으로 ABC 알고리즘에서 벌의 행동은 그림 1과 같이 세 단계로 나뉜다. 1단계에서 일벌은 꿀이 있는 지역의 확보율을 계산하기 위해 채집 장소로 이동하여 꿀을 발견하면 8자춤(waggle dance)을 춘다[10]. 이를 발견한 대기벌은 2단계에서 보이는 것과 같이 일벌을 따라 8자춤을 춘다. 이때 8자춤의 진동수가 높으면 높을수록 꿀 확보율이 높은 지역을 뜻한다. 마지막 3단계에서, 대다수의 대기벌이 진동수가 더 높은 춤을 추는 일벌들을 따라 춘다. 이후 일벌은 탐색벌이 되어 더 꿀 확보율이 좋은 장소를 탐색하기 위해 이동한다.

그림 1에서 꿀의 양에 따라 이동하는 벌의 행동 양식을 나타내기 위해 확보율이 좋은 채집 장소와 좋지 않은 장소를 구분하였다. 수식 (1)은 i구역을 대기벌이 선택 할 가능성을 p_i로 나타내었다.

$$p_i = \frac{F_i}{\sum_{i=1}^{SN} F_i}. \tag{1}$$

식 (1)에 의해, 꿀 채집이 유리한 장소를 대기벌이 찾아낸다. 대기벌이 i구역을 선택한 후 다른 대기벌은 i구역의 근방인 $i+1$구역의 확보율을 고려하고 i구역과 비교한다. 만약 새로운 장소의 확보율이 이전의 장소보다 좋으면 이전은 장소는 버려지고 새로운 장소가 선정된다. 이 3단계의 과정을 계속 반복한다.

이전 지역과 비교하여 이웃 지역이 확보율이 더 높음을 증명하기 이전에 식 (2)를 사용하여 기존 지역의 확보율을 계산한다.

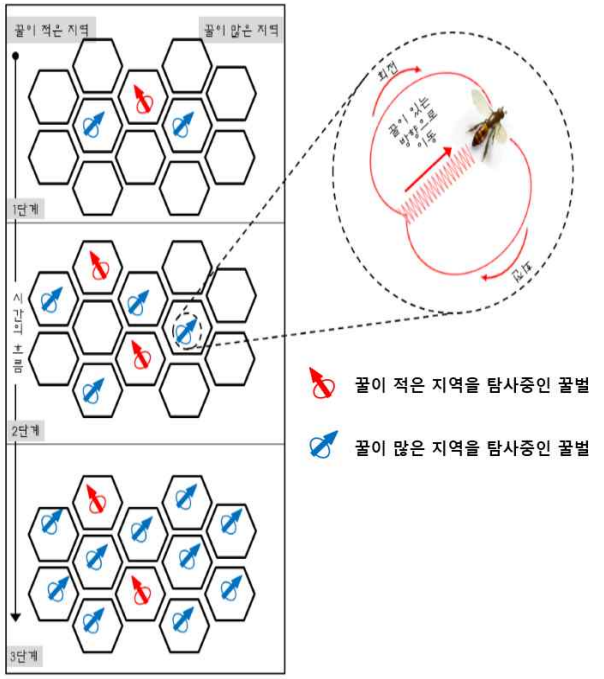

그림 1. 단계별로 꿀을 찾는 벌의 일 분담.
Fig. 1. The recruitment of bees for foraging.

$$L_{ij}^{curr} = L_{ij}^{old} + \sigma(L_{ij}^{old} - L_{kj}). \tag{2}$$

σ은 [-1, 1] 사이의 값을 가지며 k는 인접한 꿀 채집 장소의 위치 번호를 나타낸다. 또한 j는 벌의 진행방향과 거리 값을 갖는 벡터 값이다. 식 (2)의 값을 도출한 후 식 (3)을 계산해 탐색벌이 기존의 지역을 벗어나 확보율이 높은 새로운 지역을 탐지한다.

$$L_{ij}^{anew} = MinL_{ij} + Rand(0,1)(MaxL_{ij} - MinL_{ij}). \tag{3}$$

2. 제안된 알고리즘으로 보완된 IZRP

알고리즘 1은 IZRP의 존 반경 결정 알고리즘에서 최소 탐색 방법과 조정 트래픽 측정 방법을 보여준다. 하지만 최소 탐색 방법은 새로운 최적의 존 반경을 찾기 위해 이웃한 노드를 한 홉씩 증가 시켜야하기에 빠른 시간 내에 찾지 못하는 단점이 있다.

알고리즘 2는 한 홉씩 늘리지 않고 존 반경의 차($\triangle\rho$)에 따라 인접 노드의 ρ의 값을 바꾸는 메시지를 보낸다. 이는 이전의 꿀이 있는 지역과 현재의 지역과 비교해서 꿀이 많은 장소를 결정하는 대기벌의 행동 양식을 반영하였다. 존 반경 결정 알고리즘에서 노드는 자신의 반경과 인접한 노드의 반경을 비교하여 인접한 노드로 $\triangle\rho$값만큼 반경 수정을 요청하는 URM (Up-Radius Massage)와 DRM (Down-Radius Massage)를 보낸다.

$T_1(n)$은 알고리즘 1을 이용하여 입력의 수가 n일 때 존 반경 계산 시간이다. 처음엔 $T_1(n)$을 1로 초기치 설정한다. 존 반경 결정 알고리즘은 현재 IZRP의 트래픽인 Z(ρ)의 입력을 2000에서 20000사이에 균일하게 분포된 함수로 가정했다. 노드들은 IZRP 트래픽에 근거한 그들의 존 반경

을 고작 한 홉 증가 또는 감소시켜 변경한다. 즉 중심 노드의 존 반경과 이웃노드의 반경의 차이($\triangle\rho$)가 2보다 클지라도 노드는 최적의 반경을 찾을 때까지 한 홉씩 증가 또는 감소시킨다. 그러므로 이 단계에서 계산시간은$(1+n+(n-1))$으로 계산된다. 이후 최상의 적응반경을 결정하는 적응 트래픽 측정 방법을 사용하는 단계로 진입한다. 현재의 존 반경은 이전의 존 반경과 비교하여 네트워크의 리액티브 라우팅 또는 프로액티브 라우팅의 트래픽에 의해 결정된다. $T_1(n)$의 계산 시간은 다음과 같이 측정한다.

$$T_1(n) = 5n + 7. \quad (4)$$

알고리즘 1: IZRP의 존 반경 결정 알고리즘 (기존 알고리즘)

1:**Procedure** IZRP Algorithm
2:**input**: n = Z(ρ);
//노드의 존 반경 크기에 따른 트래픽
3:**output**: Optimal Zone Radius(ρoptimal);
4: **for** Process n **do**
5: Check the difference of nodes's zone radii $\Delta\rho$.
6: **if** Z(ρ) < $Z_{prev}(\rho)$ **then**
7: **for** Process ρ_{curr} with $\Delta\rho$ **do**
8: ρ_{curr}=ρ++;
9: **end for**
10: **else**
11: **for** Process ρ_{curr} with Δdo
12: ρ_{curr}=ρ--;
13: **end for**
14: **end if**
15: **if** ρ_{curr}< $\rho_{optimal}$ **then**
16: I_{E_j} increases, I_{A_i} decreases.
17: **else**
18: I_{A_i} increases, I_{E_j} decreases.
19: **end if**
20: Obtain $\Gamma(\rho) = \frac{I_{E_j}}{I_{A_i}}$.
21: **if** $\Gamma(\rho) > \Gamma_{threshold}(\rho)$ **then**
22: $\rho_{optimal}$= ρ_{curr}++;
23: **else**
24: $\rho_{optimal}$= ρ_{curr}--;
25: **end if**
26: **end for**
27: **return**: Min Search Scheme;
28: **end procedure**

$T_2(n)$는 기존의 IZRP의 존 반경 결정법에서 수정된 알고리즘 2를 이용해 설정된 존 반경 계산 시간이다. 이 알고리즘의 존 반경 결정 방법은 최소 탐색 방법 단계에서 기존의 존 반경 결정 방법과는 차이가 있다. 한 홉씩 증가시켜 존 반경을 미세 조정하는 것 대신에 이웃노드를 존 반경의 차이($\triangle\rho$)만큼 변화시킨다. 그러므로 제안한 알고리즘의 존 반경 계산 시간은 다음과 같다.

$$T_2(n) = 2n+6. \quad (5)$$

식 (4)와 (5)에서 우리는 $T_2(n)$이 $T_1(n)$보다 작다는 것을 알 수 있다. 게다가 $T_1(n)$과 $T_2(n)$의 방정식은 입력 n의 상수 값에 비례하여 상승하게 된다. 입력 n의 상수를 보면, 제안한 알고리즘은 노드에서 많은 양의 패킷을 전송할 때 트래픽을 감소시킨다. 즉 제안한 알고리즘이 존 반경 결정시간을 단축해 준다는 것을 식 (5)를 통해 알 수 있다.

알고리즘 2: IZRP에서 총체적인 제어 트래픽을 줄이기 위해 제안된 알고리즘

1:**Pocedure** PROPOSED ALGORITHM
2:**input**:*n* =*Z*(ρ) //total control traffic corresponding to the number of hops of nodes' zone radii in the network.
3:**output**: Optimal Zone Radius ($\rho_{optimal}$);
4: **for** Process *n* **do**
5: **if** *Z*(ρ) ! = $Z(\rho)_{prev}$ **then**
6: Check the difference of nodes's zone radii $\Delta\rho$.
7: **if** $\Delta\rho$= $|\rho_i-\rho_j|$<2 **then**
8: node *i* requests the adjacent nodes to directly change $\Delta\rho$by sending a message URM($\Delta\rho$);
9: **else**
10: node *i* requests the adjacent nodes to directly change $\Delta\rho$by sending a message DRM($\Delta\rho$);
11: **end if**
12: **end if**
13: **if** ρ_{curr} < $\rho_{optimal}$ **then**
14: I_{E_j} increases, I_{A_i} decreases.
15: **else**
16: I_{A_i} increases, I_{E_j} decreases.
17: **end if**
18: Obtain $\Gamma(\rho) = \frac{I_{E_j}}{I_{A_i}}$.
19: **if** $\Gamma(\rho) > \Gamma_{threshold}(\rho)$ **then**
20: $\rho_{optimal}$= ρ_{curr}++;
21: **else**
22: $\rho_{optimal}$= ρ_{curr}--;
23: **end if**
24: **end for**
25: **return**: Min Search Scheme;
26: **end procedure**

IV. 모의실험 및 분석

본 절에서 IZRP 알고리즘과 제안된 알고리즘의 입력의 수에 따른 계산 복잡성을 모의실험을 통해 비교하고 그 차이를 증명한다. 그 다음 두 알고리즘의 종단 지연을 비교하

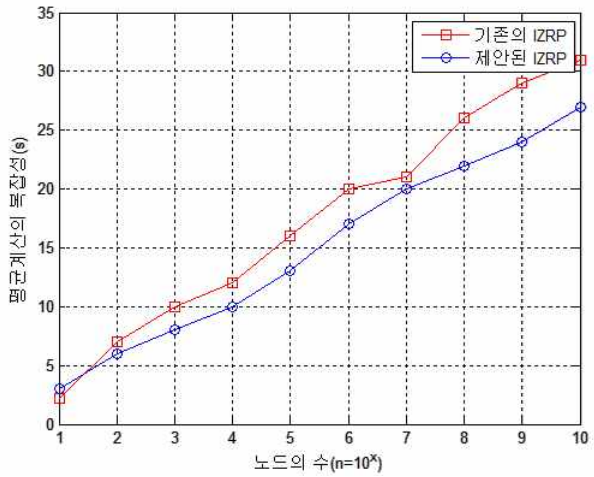

그림 2. IZRP과 제안된 알고리즘의 계산 복잡성 비교.

Fig. 2. Computational complexity of the IZRP algorithm verse that of our proposed algorithm.

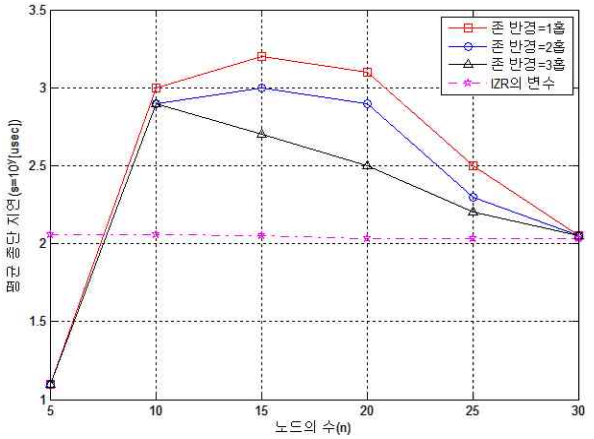

그림 3. 기존의 방법에서 존 반경이 1, 2, 3 일 때와 IZR을 사용할 때의 평균 종단지연 비교.

Fig. 3. Comparison results between the average end-to-end delay in ZRP with different zone radii 1 to 3 and IZRP.

여 제안된 알고리즘이 기존의 IZRP 알고리즘 보다 지연이 짧아 효율성이 높음을 보였다.

그림 2는 입력의 수(n)에 따른 기존 알고리즘과 제안된 알고리즘의 계산의 복잡성의 차이를 나타낸다. 특정 CPU clock(Intel Core2Due 2.66GHz(CPU), 2GB(RAM), WindowXP)을 사용하여 도출된 실행 시간에 따라 계산 복잡성을 결정하였다. 이 복잡성은 입력의 수 n과 노드의 존 반경 크기에 따른 제어 트래픽에 따라 변경된다. 알고리즘 2에서 인접한 노드에서 홉의 수가 URM/DRM을 통하여 직접적으로 변경될 때 최적의 존 반경을 찾기 위한 시간이 현저히 감소된다. 입력의 수 n이 2000일 때 IZRP보다 10%정도 복잡성가 낮고 n=10,000일 때 20%, n=20,000일 때 23%의 개선을 보여 지연을 줄였다.

노드의 수에 따른 평균 종단 지연 시간의 값의 변화를 그림 3을 통해 알 수 있다. 존 반경이 1에서부터 3으로 변할 때의 그래프를 보면 노드의 수가 증가함에 따라 존 반경이 큰 경우, 지연 시간이 감소됨을 알 수 있다. 그리하여 IZR을 사용하여 존 반경을 설정한 경우 우수한 성능을 보였다.

V. 결론

본 논문에서는 벌이 꿀을 찾아다니는 행동을 근거하여 MANET에서 노드의 이동성을 고려하여 효율적인 존 라우팅 결정 알고리즘을 제안하였다. 모의실험 결과에서, 제안된 방법이 기존의 것과 비교하여 MANET에서 계산 복잡성 및 종단 지연 시간의 측면에서 보다 개선됨을 보였다.

IZRP기반의 MANET에서 노드들의 이동성은 존 반경 조절시 지연을 야기한다. 또한 존 반경은 MANET에서 이기종간의 중요한 역할을 하며 변화하는 네트워크 상태 변화에 빠르게 적응하게 한다. 그러므로 제안된 알고리즘은 존 반결 결정시 지연을 줄여 라우팅 트래픽을 감소시킨다.

향후 연구 방향은 대규모의 무선 애드 혹 네트워크에서 네트워크 연결성을 더욱 증가시킬 수 있는 방법에 대해 연구할 것이다.

REFERENCES

[1] T.-L. Pham and D.-S. Kim, “Routing protocol over lossy links for ISA100.11a industrial wireless sensor networks,” *Wireless Networks*, pp. 1-12, 2014.

[2] T. T. Le and D.-S. Kim, “An efficient throughput improvement through bandwidth awareness in cognitive radio networks,” *Journal of Communications and Networks (JCN)*, vol. 16, no. 2, pp. 146-154, 2014.

[3] T. T. Le, P. T. A. Quang, S. Y. Shin, and D.-S. Kim, “BAR: Bandwidth-Aware opportunistic localized-routing for cognitive radio networks,” *GC12 WS MIMOCR*, pp. 937-942, Dec. 2012.

[4] T. M. Phuong and D.-S. Kim, “Efficient power control scheme for cognitive industrial sensor networks,” *International Journal of Control and Automation (IJCA)*, vol. 7, no. 3, pp. 177-188, 2014.

[5] D. D. Tan and D.-S. Kim, “Dynamic traffic-aware routing algorithm for multi-sink wireless sensor networks,” *Wireless Networks*, vol. 19, no. 8, 2013.

[6] P. Samar, M. Pearlman, and Z. Haas, “Independent zone routing: an adaptive hybrid routing framework for ad hoc wireless networks,” *IEEE/ACM Transactions on Networking*, vol. 12, no. 4, pp. 595-608, Aug. 2004.

[7] M. Pearlman and Z. Haas, “Determining the optimal configuration for the zone routing protocol,” *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1395-1414, Aug. 1999.

[8] D. Ravilla, V. Sumalatha, and P. C. S. Reddy, “Performance comparisons of ZRP and IZRP routing protocols for Ad hoc Wireless Networks,” *International Conference on Energy, Automation and Signal*, pp. 1-8, Dec. 2011.

[9] D. Karaboga, “An idea based on honey bee swarm for numerical optimization,” *Technical Report*, Oct. 2005.
[10] T. D. Seeley, “*Honeybee Democracy*,” Princeton University Press, pp. 280-288, 2010.
[11] H. M. Lee, D.-S. Kim, “nsect-Inspired Algorithm for Zone Radius Determination in Independent Zone Routing Protocol,” *Proc. of 2014 29th ICROS Annual Conference (ICROS 2014) (in Korean)*, Daegu, Korea, pp. 25-26 Jun. 2014.

이 혜 민

2014년 금오공과대 전자공학과 졸업. 2014년 동 대학원 석사. 관심분야는 정보 및 네트워킹, 무선 네트워킹, 실시간 시스템.

김 동 성

1992년 한양대학교 전자공학과 학사졸업. 2003년 서울대학교, 전기 및 컴퓨터공학부 박사졸업. 2004년 Cornell 대학교 ECE 박사 후 연구원. 2004년 3월~현재 금오공대 전자공학부 부교수. 2014년 6월~현재 미래창조과학부 지원 ICT 융합 특성화 연구센터 센터장. 관심분야는 국방/산업용 제어 통신망, 실시간 임베디드 시스템.