

DTN에서 노드 간 연결 가능성과 마스킹 연산을 이용한 중계노드 선정 기법

정래진¹ · 전일규² · 우병훈³ · 구남경⁴ · 이강환^{5*}

Relay Node Selection Method using Node-to-node Connectivity and Masking Operation in Delay Tolerant Networks

Rae-jin Jeong¹ · Il-Kyu Jeon² · Byeong-hun Woo³ · Nam-kyoung Koo⁴ · Kang-whan Lee^{5*}

^{1,2}Interdisciplinary Program in Creative Engineering and Department of Computer Science Engineering, Korea University of Technology and Education, Cheonan 31253, Korea

^{3-5*}Department of Computer Science Engineering, Korea University of Technology and Education, Cheonan 31253, Korea

요 약

본 논문에서는 이동 속성 정보를 활용하여 이동 노드간의 연결 가능성을 분석하고 마스킹 기법을 이용하여 이웃한 이동노드 중 목적 노드와 연결 가능성이 가장 높은 이동노드를 중계노드로 선정하는 EPCM(Enhanced Prediction-based Context-awareness Matrix)을 제안한다. 기존 Delay Tolerant Network (DTN)의 전송방식은 노드의 단순 이동성에 의존하여 목적노드로 메시지를 전송하게 된다. 이러한 경우 목적노드와의 연결성이 낮은 이동노드를 중계노드로 선정하게 되면 이동노드의 메시지 저장 및 처리 능력 제한으로 인하여 전송 지연 또는 패킷 손실의 원인이 된다. 본 논문의 제안된 알고리즘에서는 이동노드의 속도와 방향 속성 정보를 고려하여 목적노드와의 연결성을 계산하고 마스킹 연산을 활용하여 가장 높은 연결 가능성을 가지고 있는 중계노드를 선정하여 목적 노드까지 메시지를 전달하게 된다. 모의실험에서 Epidemic 및 PROPHET 알고리즘과 제안하는 알고리즘의 패킷 전송률을 비교한 결과 제안하는 알고리즘이 노드의 이동 속성을 고려한 연결성으로 보다 높은 패킷 전송률을 보여주었다.

ABSTRACT

This paper propose an improving relay node selection method for node-to-node connectivity. This concern with the mobility and analysis of deployed for masking operation using highest connectivity node. The major of Delay Tolerant Network (DTN) routing protocols make use of simple forwarding approach to transmit the message depend on the node's mobility. In this cases, the selection of the irrelevant mobile node induced the delay and packet delivery loss caused by limiting buffer size and computational power of node. Also the proposed algorithm provides the node connectivity considering the mobility and direction select the highest connectivity node from neighbor node using masking operation. From the simulation results, the proposed algorithm compared the packet delivery ratio with PROPHET and Epidemic. The proposed Enhanced Prediction-based Context-awareness Matrix(EPCM) algorithm shows an advantage packet delivery ratio even with selecting relay node according to mobility and direction.

키워드 : 지연내성망, 상황인식, 연결성, 마스킹

Key word : Delay Tolerant Network, Context-awareness, Connectivity, Masking

Received 24 February 2016, Revised 29 February 2016, Accepted 21 March 2016

* Corresponding Author Kang-Whan Lee(E-mail:kwlee@koreatech.ac.kr, Tel:+82-42-560-1356)

Department of Computer Science & Engineering, Korea University of Technology and Education, Cheonan 31253, Korea

Open Access <http://dx.doi.org/10.6109/jkiice.2016.20.5.1020>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서론

무선 네트워크에서 노드의 이동성에 기인한 연결 단절 현상을 극복하기 위해 노드간의 연결 가능성을 분석하는 기회적 통신(Opportunistic Communication)이 사물 간 사회적 관계성 및 이동성 분석 등의 연구가 융합되어 많은 응용연구가 이루어지고 있다[1,2]. 기회적 통신은 노드간의 연결을 상시 확립하는 것이 아니라 서로 다른 노드가 물리적으로 근접하는 기회에 노드 간 연결을 확립한다. DTN(Delay Tolerant Network)은 기회적 통신방식으로 노드 간 통신을 수행하는 네트워크이며, 지연 내성망이라고도 한다. DTN은 통신을 위한 기반시설이 부족한 바다, 사막같이 드넓은 지역과 일정한 시간 및 경로로 대중교통 수단이 내부를 순환하는 도시, 통신 기반 시설이 부족한 재해재난 환경에서의 통신수단으로 연구되고 있다[3-5].

DTN은 기회적 통신 방식에 따라 종단 간 경로확립과 정 없이 서로 다른 노드가 서로 근접하면 연결을 확립하며, 근접한 이웃노드의 버퍼에 메시지를 저장한다. 이후 메시지는 이동노드의 이동성에 따라 운반 및 중계되어 네트워크를 이동하며, 최종적으로 목적노드에게 전송된다. 종단 간 통신은 메시지 전송을 이동노드의 이동성에 의존하고 있으므로 통신이 확립된 모든 이동노드 중에서 목적노드에게 메시지를 성공적으로 전달할 수 있는 이웃노드를 선정하는 것이 중요하다. 단, 목적노드에게 전달되지 못한 메시지는 이동 노드의 제한된 메시지 저장 및 처리능력으로 전송 지연 시간이 증가하며, 패킷 손실 현상이 발생할 가능성이 있다. 따라서 이동 노드간의 연결 가능성이 높은 중계노드를 선정하여 효과적으로 메시지를 전달하는 방안이 필수적이다.

제안하는 알고리즘인 EPCM(Enhanced Prediction-based Context-awareness Matrix)은 노드의 이동 속성 정보를 활용하여 노드간의 연결 가능성을 산출하며, 마스킹 연산을 활용하여 이웃한 노드들 중 연결성이 가장 높은 노드를 중계노드로 선정한다. 이동 속성 정보를 활용하여 연결 가능성을 판단함으로써 해당 이동노드를 지향하고 빠르게 이동하는 중계노드를 선택하여 메시지를 전달할 수 있다. 또한 본 연구에서는 마스킹 연산을 활용하여 물리적으로 근접한 이웃노드 뿐만 아니라 이웃노드를 거쳐 연결되어 있는 다른 노드들을 활용하여 중계노드를 선정함으로써 메시지를 효과적으로

전달 할 수 있는 중계노드를 선정하여 메시지를 목적노드에게 전달한다. 모의실험에서 대표적인 중계노드 선정 기법인 Epidemic 및 PROPHET 알고리즘과 제안하는 알고리즘을 패킷전송률 측면을 비교하였을 때, 노드의 이동 속성 정보를 활용하여 연결 가능성을 산출하고, 중계노드를 선정한 제안하는 알고리즘이 더 향상된 패킷전송률을 나타내었다.

II. 관련연구

노드간의 연결을 상시 유지하지 않고 노드 간에 물리적으로 근접하였을 경우에 한하여 연결을 확립하는 DTN의 특성에 따라 이동노드의 속성정보를 기반으로 하여 효율적인 중계노드를 선택하는 연구가 진행되어 왔다. Epidemic은 근접하여 통신을 확립한 모든 이동노드들에 전송할 메시지를 복사하여 전송하고 이동노드의 이동성을 활용하여 네트워크에 넓게 전파함으로써 목적노드에게 메시지를 전달하는 중계노드 선정 기법이다[6]. Epidemic에서는 서로 다른 이동노드가 근접하면 이동 노드에 저장된 메시지의 정보를 포함하고 있는 Summary vector를 교환하게 된다. 이를 교환하여 각 이동노드에 저장된 특정 메시지의 존재 유무에 대하여 파악할 수 있으며 Summary vector 안의 메시지 정보를 통해 연결되어 있는 이동노드가 특정 메시지를 가지고 있지 않다면, 해당 메시지를 복사하여 버퍼에 저장하게 된다. 저장된 메시지는 네트워크 상에 넓게 퍼져나가며 목적노드에게 전달된다.

노드들의 연결 횟수, 시간, 노드간의 이행성에 따른 연결가능성으로 중계노드를 선택하는 알고리즘인 PROPHET(Probabilistic ROuting Protocol using History of Encounter and Transitivity)은 노드들이 연결된 경우 두 노드에 대한 연결가능성이 증가, 연결되지 않은 경우 시간에 따라 연결가능성이 감소하게 된다[7]. 또한 제 3의 노드를 거쳐 연결되는 경우를 고려하여 연결가능성을 계산한다. 메시지는 연결이 확립되어 있는 노드 중 연결 가능성이 가장 높은 노드를 선택하여 메시지를 전달하게 된다.

노드의 사회적인 관계성을 중계노드 선정에 고려한 BUBBLE Rap 알고리즘은 노드들의 사회적인 관계성을 커뮤니티(community)로 그룹화하고, 센트럴리티

(centrality)로 속성을 분석하여 메시지를 전달한다[8]. 커뮤니티는 인간관계, 지리, 직업, 흥미 등의 공통된 속성에 따라 나누어진 그룹을 의미하며, 내부에서 여러 서브 커뮤니티로 나누어져 계층구조를 이룬다. 센트럴리티는 커뮤니티 안에 소속된 노드 중에서 다른 노드와 상호작용하는 정도를 나타낸다. Bubble Rap은 커뮤니티에서 높은 센트럴리티를 가진 이동노드를 중계노드로 선정하여 메시지를 전달하는 기법이다.

III. 본 론

3.1. 연결 가능성 분석

본 논문에서 제안하는 노드 간 연결 가능성은 노드의 속도와 방향을 활용하여 노드간의 연결 가능성을 산출하는 방법을 제시하고 있다. 서로 다른 노드가 근접하여 통신을 확립하면, 노드의 ID, 속도와 방향, 속성정보의 유효기한, 연결 상태 등을 나타내는 네트워크 상황 벡터를 교환한다.

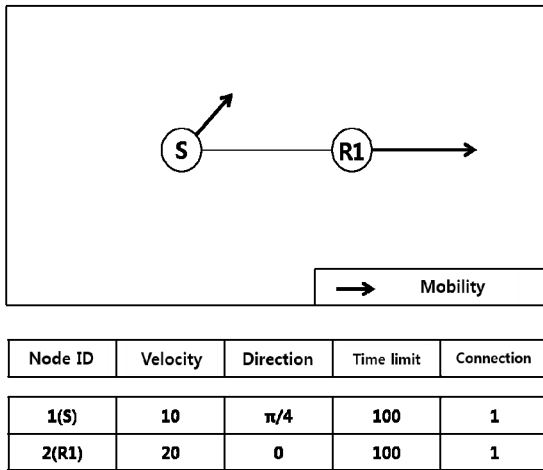


Fig. 1 Node-to-Node Connection and Network Status Vector

여기서, 유효기한은 속성정보가 네트워크 상황에 저장되어 있는 기한을 나타내며, 노드간의 연결이 단절된 후 1씩 감소하여 0이 되면 해당 네트워크 상황 벡터가 제거된다. 연결 상태는 노드간의 연결 상태를 0

과 1로 표현한다.

그림 1에서 소스노드(S)가 목적노드(D)에게 메시지를 전달하고자 할 때, 소스노드는 자신의 네트워크 상황 벡터를 참조하여 목적노드의 이동 속성 정보가 저장되어 있는지 확인한다. 존재하지 않는다면, 수식 1에 따라 주변의 이웃 노드와 자신간의 방향의 유사성과 이동 속도 등을 고려한 연결 가능성을 산출한다.

$$CN_{s,r} = CN_r(v) \times (1 - CN_{s,r}(Dir)) = \frac{v_r}{\sum_{r' \in NH} v_{r'}} \times (1 - \frac{|Dir_r - Dir_s|}{\sum_{r' \in NH} |Dir_{r'} - Dir_s|}). \quad (1)$$

여기서, $CN_{s,r}$ 는 소스 노드 s 와 이웃 노드 r 의 연결 가능성을 의미하며, v_s 와 Dir_s 는 노드 s 의 예측된 속도와 방향을 나타낸다. NH 는 이웃 노드의 집합이며 $CN_r(v)$ 는 특정 이웃 노드 r 가 다른 이웃 노드들과의 속도 비를 0과 1사이의 값으로 나타내며 r 의 속도가 빠를수록 큰 값으로 표현된다. 마찬가지로 $CN_{s,r}(Dir)$ 은 s 와 r 의 방향차이의 비를 0과 1사이의 값으로 표현하며, 이 값이 작을수록 두 노드가 유사한 방향으로 이동하고 있음을 나타낸다. 단, $CN_{i,j}(Dir)$ 에서 $|Dir_i - Dir_j|$ 의 값이 0일 때, 0로 표현된다.

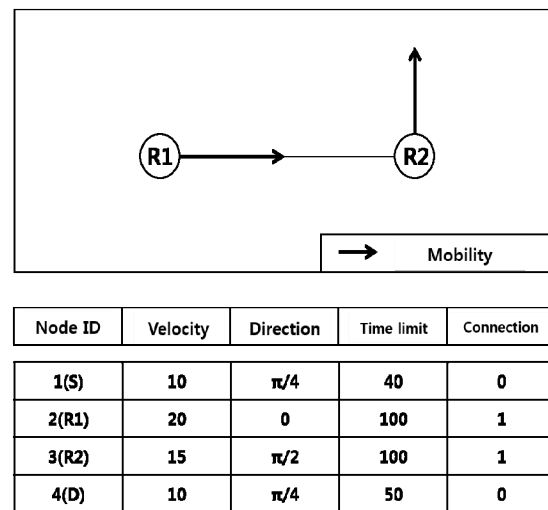


Fig. 2 Node-to-Node Connection and Updated Network Status Vector

그림 2와 같이 중계노드(R1)는 네트워크를 이동하면서 다른 중계노드(R2)와 통신을 확립하면 서로가 보유하고 있는 다른 노드들의 이동 속성 정보를 교환하며, 각자의 네트워크 상황 벡터를 갱신한다. 갱신한 네트워크 상황 벡터에서 메시지를 전달하고자 하는 목적노드(D)의 이동 속성 정보가 존재한다면, 목적노드(D)와 이동 속성 정보를 교환한 중계노드(R2)의 속도와 방향을 고려하여 그림 3과 같이 목적노드(D)를 지향하는 방향을 산출하게 된다. 단, 갱신한 네트워크 상황 벡터에서 존재하지 않는다면, 수식 1을 활용하여 방향의 유사성과 이동속도 등을 고려한 연결 가능성에 따라 중계노드를 선택한다.

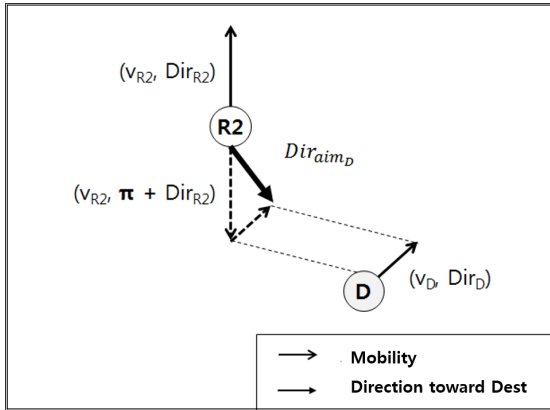


Fig. 3 Calculation of direction toward a destination node

수식 2은 목적노드의 이동 속성 정보가 네트워크 상황 벡터에 존재할 때 목적노드를 지향하는 방향을 나타내고 있다.

$$Dir_{aim_D} = \arctan\left(\frac{v_{R2} \times \sin(\pi + Dir_{R2}) + v_D \times \sin Dir_D}{v_{R2} \times \cos(\pi + Dir_{R2}) + v_D \times \cos Dir_D}\right) \quad (2)$$

여기서, Dir_{aim_D} 는 목적노드를 지향하는 방향이며, v_i 와 Dir_i 는 이웃 노드 i 의 속도와 방향을 의미한다.

산출된 목적 노드를 지향하는 방향과 주변 이웃 노드의 속도와 방향을 고려하여, 해당 노드와 목적 노드가 근접하여 통신을 확립할 수 있는 상황을 그림 4와 같이 3가지로 구분하였다.

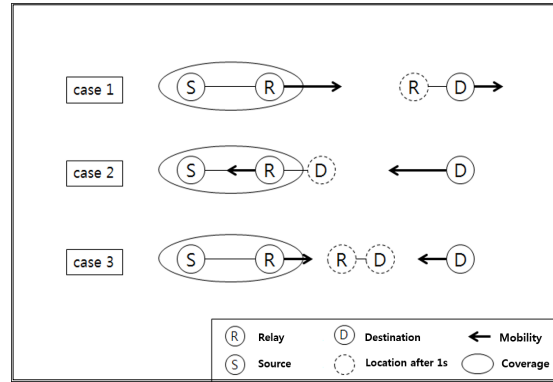


Fig. 4 Situation in which destination and the relay is close

첫 번째 경우에서 이웃 노드가 목적 노드를 지향하는 방향과 유사한 방향으로 빠르게 이동하는 상황이다. 즉, 이웃 노드의 방향과 목적 노드를 지향하는 방향의 차와 이웃노드와 목적노드의 차가 $\pi/2$ 보다 작고, 이웃 노드의 속도가 목적 노드보다 빠른 경우이다. 첫 번째 상황에 대해 수식3을 활용하여 이웃 노드 r과 목적 노드 d에 대한 연결 가능성을 표현 할 수 있다.

$$CN_{r,d} = CN_r(v) \times (1 - CN_{r,d}(Dir)) \times (1 - CN_{r,aim_D}(Dir)) = \frac{v_r}{\sum_{r' \in NH} v_{r'}} \times \left(1 - \frac{|Dir_r - Dir_d|}{\sum_{r' \in NH} |Dir_{r'} - Dir_d|}\right) \times \left(1 - \frac{|Dir_r - Dir_{aim_D}|}{\sum_{r' \in NH} |Dir_{r'} - Dir_{aim_D}|}\right). \quad (3)$$

여기서, $CN_{r,d}$ 는 이웃 노드 r 와 목적 노드 d 의 연결 가능성을 나타내며, v_r 와 Dir_d 는 이웃 노드 r 의 속도와 방향을 의미한다. Dir_{aim_D} 는 속성정보를 전달한 목적 노드 d 를 지향하는 방향, NH 는 이웃 노드의 집합이다. 단, $CN_{r,d}(Dir)$ 와 $CN_{r,aim_D}(Dir)$ 는 $|Dir_r - Dir_d|$ 와 $|Dir_r - Dir_{aim_D}|$ 가 0일 때, 0로 나타내어진다. 다음의 표 1과 같이 이웃 노드 R1, R2와 목적 노드 D의 속도와 방향이 주어질 때, R1과 R2에 대한 연결 가능성은 아래와 같이 구할 수 있다.

Table. 1 Mobility Property of Nodes(case 1)

Node \ Property	Velocity	Direction
R1	30	0°
R2	20	0°
D	10	0°

예시

$$CN_{R1,D} = CN_{R1}(v) \times (1 - CN_{R1,D}(Dir)) \times (1 - CN_{R1,aim_D}(Dir))$$

$$= \frac{30}{30+20} \times (1-0) \times (1-0) = 0.6,$$

$$CN_{R2,D} = CN_{R2}(v) \times (1 - CN_{R2,D}(Dir)) \times (1 - CN_{R2,aim_D}(Dir))$$

$$= \frac{20}{30+20} \times (1-0) \times (1-0) = 0.4.$$

두 번째 경우는 목적 노드와 이웃 노드가 이동하는 방향은 같으나 목적 노드가 이웃 노드의 근처로 빠르게 이동하는 상황을 나타내고 있다. 즉 이웃 노드의 방향과 목적 노드를 지향하는 방향의 차가 $\pi/2$ 보다 크며 이웃 노드와 목적노드의 방향의 차도 $\pi/2$ 보다 작고, 목적 노드의 속도가 이웃 노드보다 빠른 경우이다. 두 번째 상황에 대해 수식4를 활용하여 이웃 노드 r 와 목적 노드 d 에 대한 연결 가능성을 표현 할 수 있다.

$$CN_{r,d} = (1 - CN_r(v)) \times (1 - CN_{r,d}(Dir)) \times (1 - CN_{r,aim_p}(Dir)) \quad (4)$$

다음의 표 2와 같이 이웃 노드 R1, R2와 목적 노드 D의 속도와 방향이 주어질 때, R1과 R2에 대한 연결 가능성은 아래와 같이 구할 수 있다.

Table. 2 Mobility Property of Nodes(case 2)

Node \ Property	Velocity	Direction
R1	10	180°
R2	20	180°
D	30	180°

예시

$$CN_{R1,D} = (1 - CN_{R1}(v)) \times (1 - CN_{R1,D}(Dir)) \times (1 - CN_{R1,aim_D}(Dir))$$

$$= (1 - \frac{10}{10+20}) \times (1-0) \times (1-0) = 0.6,$$

$$CN_{R2,D} = (1 - CN_{R2}(v)) \times (1 - CN_{R2,D}(Dir)) \times (1 - CN_{R2,aim_D}(Dir))$$

$$= (1 - \frac{20}{10+20}) \times (1-0) \times (1-0) = 0.3.$$

세 번째 경우는 목적 노드와 이웃 노드가 서로를 향해 이동하고 있는 상황을 나타내고 있다. 즉 이웃 노드의 방향과 목적 노드를 지향하는 방향의 차가 $\pi/2$ 보다 작으며, 목적 노드와 이웃 노드의 방향도 $\pi/2$ 보다 작은 경우이다. 세 번째 상황에 대해 수식 5를 활용하여 이웃 노드 r 와 목적 노드 d 에 대한 연결 가능성을 표현 할 수 있다.

$$CN_{r,d} = CN_r(v) \times CN_{r,d}(Dir) \times (1 - CN_{r,aim_p}(Dir)) \quad (5)$$

위의 목적 노드와 이웃 노드가 서로 근접하여 통신을 확립할 수 있는 상황에 따라 연결 가능성을 산출하며, 다음 장의 마스킹 연산을 활용하여 가장 높은 연결 가능성을 보이는 이웃 노드를 중계노드로 선정하여 메시지를 전달한다.

다음의 표 3과 같이 이웃 노드 R1, R2와 목적 노드 D의 속도와 방향이 주어질 때, R1과 R2에 대한 연결 가능성은 아래와 같이 구할 수 있다.

Table. 3 Mobility Property of Nodes(case 3)

Node \ Property	Velocity	Direction
R1	10	0°
R2	20	0°
D	30	180°

예시

$$CN_{R1,D} = CN_{R1}(v) \times CN_{R1,D}(Dir) \times (1 - CN_{R1,aim_D}(Dir))$$

$$= \frac{10}{10+20} \times \frac{|0^\circ - 180^\circ|}{|0^\circ - 180^\circ| + |0^\circ - 180^\circ|} \times (1-0)$$

$$= 0.3 \times 0.5 \times 1 = 0.15,$$

$$CN_{R2,D} = CN_{R2}(v) \times CN_{R2,D}(Dir) \times (1 - CN_{R2,aim_D}(Dir))$$

$$= 0.6 \times 0.5 \times 1 = 0.3.$$

3.2. 마스크 연산을 활용한 중계노드 선정

연결 가능성 분석 기법을 통해 산출된 이웃 노드와 목적 노드의 연결 가능성을 통하여 해당 이웃노드가 목적 노드와 연결을 확립할 수 있는 가능성을 판단 할 수 있다. 마스크 연산을 활용한 중계노드 선정기법은 각 이웃 노드에 대하여 연결 상태를 근접행렬로 표현하고, 연결 가능성이 가장 높은 이웃 노드를 중계노드로 선택한다. 단, 모든 이웃 노드가 목적 노드와의 연결 가능성이 낮은 경우 이웃 노드들의 근접행렬을 병합하고 마스크 연산을 활용하여 더 넓은 범위 연결 상태 정보를 활용함으로써 중계노드를 선택한다.

제안하는 중계노드 선정기법은 각 노드마다 자신과 이웃하는 노드와의 연결 상태를 0과 1로 확립되어 있는 노드들 중에 연결 가능성이 가장 높은 이웃 노드에게 메시지를 전달한다. 하지만 그림 5와 같이 소스 노드 2의 이웃 노드 1과 3이 목적노드의 반대방향으로 이동함에 따라 목적 노드와 연결가능성이 낮은 상황이라면 메시지를 전달할 수 없다.

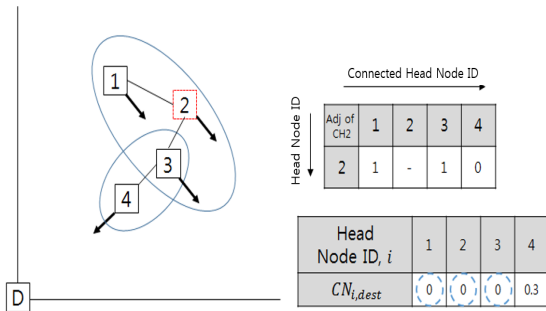


Fig. 5 Situation of destination and neighbor node

따라서 소스 노드 2는 그림 6과 같이 주변의 이웃 노드 1과 3의 인접행렬들을 OR 연산을 하여 병합행렬을 생성한다.

이웃 노드들의 인접행렬을 병합함으로써 소스 노드 2는 주변의 노드와 2-Hop 범위의 노드 4에 대해 연결 확립 여부를 확인할 수 있게 된다. 여기서, 노드 4의 연결을 나타내고 있는 인접행렬을 병합한다면 소스 노드 2로부터 3-Hop 범위의 노드의 연결 확립 여부를 찾을 수 있다. 따라서 이웃 노드의 인접행렬을 병합할수록 더 넓은 범위의 노드를 파악할 수 있게 된다.

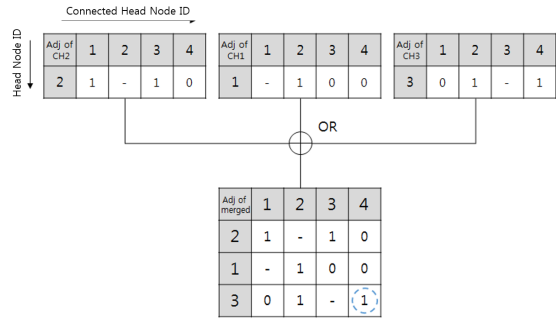


Fig. 6 Merged adjacency matrix of neighbor node

단, DTN의 통신방식에 따라 노드의 통신범위가 좁아 연결이 확립된 이웃노드의 수가 적을 수 있으므로, 본 논문에서는 소스 노드로부터 2-Hop 범위의 이웃 노드들에 대해서만 한정하여 연결 확립 여부를 검색한다. 2-Hop 범위의 노드들 중 목적 노드와 연결 가능성이 높은 노드가 존재한다면, 이웃 노드를 거쳐 메시지를 전달하는 중계노드를 선택하여 경로를 설정한다. 제안하는 기법은 마스크 연산을 병합된 인접행렬에 적용하여 경로상의 이웃 노드의 연결 가능성이 0인 경로를 제외 하면서 최종적으로 연결 가능성이 높은 중계노드를 선택하여 가장 높은 경로를 설정하게 된다. 소스 노드 2로부터 메시지를 전달하기 위해 표 4의 의사코드와 같이 소스 노드 2에 대한 마스크 비트열을 생성하고 병합된 인접행렬의 각 행에 마스크 연산을 수행한다.

Table. 4 Pseudo-code of Mask bit Array Generation

Procedure : function of making mask bit array

Input

- set of connected head node
- source head node id

output

- mask bit array for masking operation

Begin

src ← source_head_node_id

for a = 1 to n(connected_head_node) **do**

if a == src **do**

mask_bit_array[a] ← 0

else

mask_bit_array[a] ← 1

end

end

output mask_bit_array

마스크 비트열을 병합된 인접 행렬에 적용하면 그림 7과 같이 인접행렬에서 소스 노드 2에 대한 다른 이웃 노드의 연결을 제거하게 되므로 경로 산출 과정에서 메시지가 소스 노드 2로 되돌아오지 않도록 수정된다.

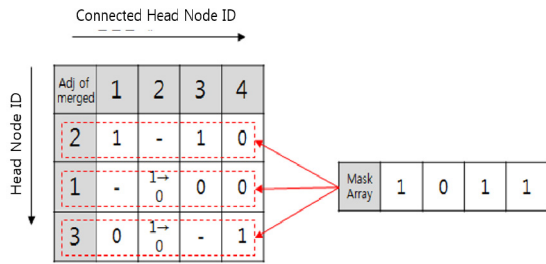


Fig. 7 Result of Masking Operation at adjacency matrix

마스크 연산을 통해 메시지의 순환을 방지한 후, 병합된 인접행렬의 소스 노드 2의 행에서 연결되어 있는 이웃 노드 1과 3의 연결 상태를 확인한다. 병합된 인접행렬에 따라 이웃 노드 1과 연결된 노드가 존재한다면 해당 노드의 연결 가능성을 비교하지만, 다른 이웃 노드들과 연결을 확립되지 않으므로 메시지를 전달할 수 없다.

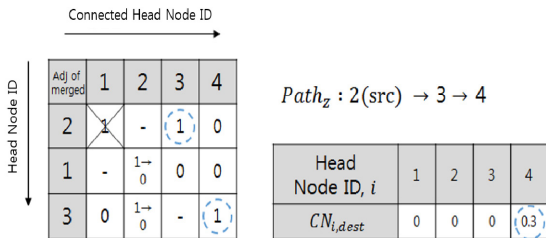


Fig. 8 Merged Adjacency Matrix and Selected Relay

따라서 그림 8과 같이 연결되어 있는 또 다른 이웃 노드 3의 연결 상태를 확인한다. 병합된 인접행렬의 값에 따라 이웃노드 3은 4와 연결을 확립하고 있으며, 노드 4와 목적 노드의 연결 가능성을 산출하여 높은 연결 가능성이 도출된다면 이웃 노드 2, 3, 4를 선택하여 경로를 산출한다. 이후 그림 9과 같이 목적 노드와 연결 가능성이 높은 중계노드가 연결을 확립하게 된다면 병합된 인접행렬을 재구성하고 마스크 연산을 적용하여 목적 노드에 근접할 수 있는 중계노드를 선

정하게 된다.

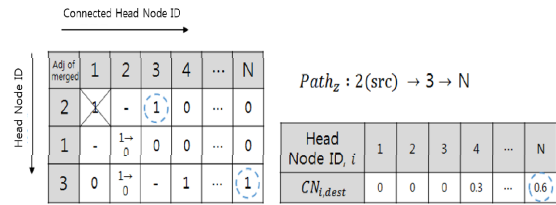


Fig. 9 Relay Node Selection According to Updated Connection

아래의 표 5는 마스크 연산을 활용한 중계노드 선정 기법의 의사코드를 나타내었다.

Table. 5 Pseudo-code of Relay Node Selection

Procedure : relay node selection technique

Input

- source head node, destination head node,
- set of neighbor head node NH,
- set of connected head node

output

- high connectivity relay node

Begin

src ← source head node,

dest ← destination head node

for a = 0 to n(NH) **do**

relay_node = argmax_i CN_{i,dest}

end

if relay_node != null **do**

return relay_node

else

merge_adj_matrix[0] ← adj_matrix of src

for b = 1 to n(NH) **do**

merge_adj_matrix[b]

← adj_matrix of NH[b]

end

mask_bit_array ← function of

making mask bit array for src

for c = 1 to n(NH)+1 **do**

merge_adj_matrix[c]

← merge_adj_matrix[c] & mask_bit_array

end

for d = 1 to n(NH) **do**

if merge_adj_matrix[0][d] == 1 **do**

for e = 1 to n(connected_head_node) **do**

```

if merge_adj_matrix[d][e] == 1 do
    relay_node ← argmax CNe,dest
                    e
end
end
end
end
output relay_node
    
```

IV. 실험 및 분석

본 논문에서 제안하는 알고리즘은 노드의 이동 속성 정보를 활용하여 노드간 연결성을 분석하고 마스킹 연산을 활용하여 중계노드를 선정한다. 알고리즘의 성능을 비교하기 위하여 비교알고리즘으로 메시지를 복제하여 네트워크에 전파하는 Epidemic과 노드 간의 연결 횟수에 따라 연결 가능성을 산출하는 PROPHET알고리즘을 선정하였다. 모의실험 환경은 아래의 표 6은 모의 실험 환경을 나타내고 있다.

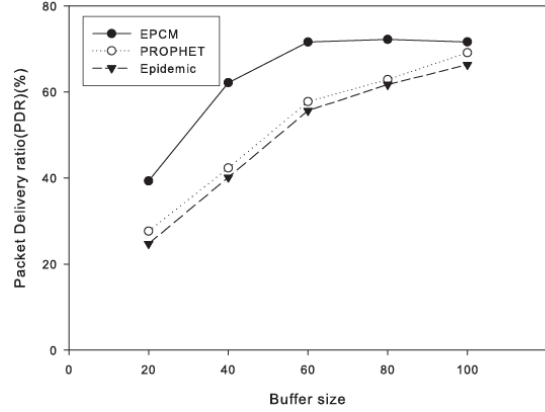
Table. 6 Simulation Environments

Parameter	Value
Network size	1000x1000, 1500x1500 (m ²)
Number of node	50, 100 (EA)
Velocity of Node	0~10, 15, 20, 25, 30 (m/s)
Coverage	20 (m)
Transmission Period	1 (sec)
Simulation Period	1800 (sec)

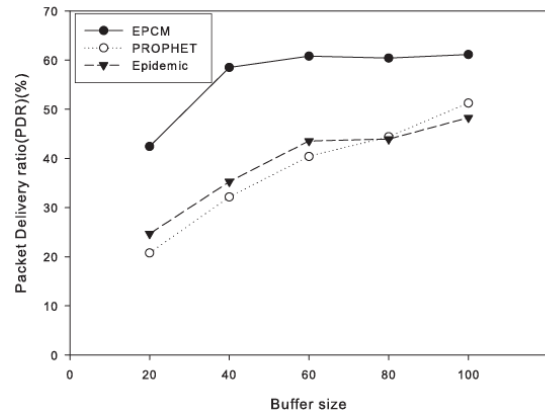
모의실험에서 진행시간이 완료된 후, 아래의 수식 6에 따라 패킷 전송률을 계산하였다.

$$PDR(t) = \frac{flow_{dest}(t)}{flow_i(t)} \quad (6)$$

여기서, $flow_{dest}(t)$ 는 목적 노드의 흐름양, $flow_i(t)$ 는 이동노드 i 의 흐름양을 의미한다.



(a)



(b)

Fig. 10 Packet Delivery Ratio according to Buffer Size (a) 1000x1000 m², 50 nodes (b) 1500x1500 m², 100 nodes

그림 10과 같이 모든 비교 알고리즘에서 버퍼크기가 증가할수록 패킷전송률이 높아지는 결과를 나타내었다. 이는 버퍼크기가 DTN에서 중요한 요소임을 확인할 수 있으며, 또한 네트워크 영역의 크기가 상대적으로 작은 경우에 노드가 근접할 수 있는 가능성이 높아 더 좋은 패킷 전송률을 보여주었다.

그림 11과 같이 모든 비교 알고리즘에서 노드의 최대 속도가 증가할수록 패킷전송률이 높아지는 결과를 나타내었다. 이는 노드의 속도가 DTN에서 중요한 요소임을 확인할 수 있으며, 또한 네트워크 영역의 크기가 상대적으로 작은 경우에 노드가 근접할 수 있는 가능성이 높아 더 좋은 패킷 전송률을 보여주었다.

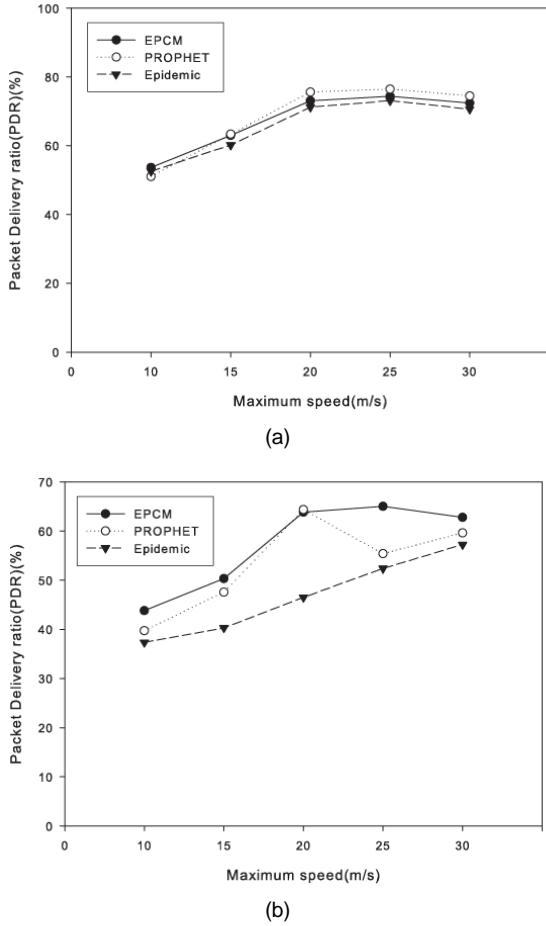


Fig. 11 Packet Delivery Ratio according to Maximum Velocity (a) 1000x1000 m², 50 nodes (b) 1500x1500 m², 100 nodes

네트워크 영역의 크기에서 1000×1000(m²)의 환경에서 수행한 실험결과가 비교한 알고리즘 모두 비슷한 성능을 보였으며, 1500×1500(m²)의 환경보다 약 10% 정도 더 높은 패킷 전송률을 보여주었다. 단, 이동노드들이 지속적으로 이동하는 환경에서는 조금 낮은 성능을 보였으나 속도가 점점 빠를수록 패킷전송률의 격차가 줄어들고 있다. 이는 메시지를 복제하여 네트워크 전체에 전파하는 Epidemic 알고리즘의 특징에 따라 메시지가 빠르게 확산됨으로써 나타나는 결과로 분석할 수 있다.

V. 결론

본 논문에서는 이동 속성 정보를 고려하여 이동노드 간 연결 가능성을 평가하며, 연결 가능성을 행렬로 표현한 후 마스킹 기법을 활용한 중계노드 선정하는 기법을 제안하였다. 기존의 DTN에서는 목적노드와 연결성이 낮은 중계노드가 선정되어 메시지를 전송하였을 경우 제한된 메시지 저장 및 처리능력으로 인한 전송 지연 및 패킷 손실 현상이 발생하였다.

이를 극복하기 위해 제안된 알고리즘은 노드의 이동 속성 정보를 고려하여 가장 높은 연결 가능성을 나타내는 중계노드를 선정하였으며, 마스킹 기법을 통해 더 넓은 범위에서 중계노드를 선정하게 된다. 모의 실험에서 제안하는 알고리즘과 Epidemic, PROPHET 알고리즘을 패킷 전송률 측면에서 성능을 비교하였으며, 이동 속성 정보를 고려하였을 때 더 효과적으로 메시지를 전달할 수 있음을 알 수 있었다. 이후 속도와 방향 외에 다른 여러 속성정보를 활용하여 이동 노드의 연결 가능성을 분석함으로써 효율적으로 메시지를 전달할 수 있는 연구가 필요하다.

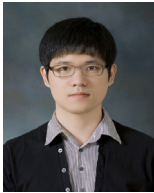
ACKNOWLEDGMENTS

This research was supported by the MSIP (Ministry of Science, ICT and Future Planning) Planning 2014H1C1A1066391), Korea, under the Specialized Co-operation between industry and academic support program (NIPA-2014-H0808-14-1007) supervised by the NIPA(National IT Industry Promotion Agency) and Education and Research Promotion Program of KUT.

REFERENCES

- [1] M. R. Schurgot, C. Comaniciu and K. Jaffres-Runser, "Beyond Traditional DTN Routing : Social Networks for Opportunistic Communication," *IEEE Communications Magazine*, vol. 50, no. 7, pp. 155-162, July 2012.

- [2] L. Pelusi, A. Passarella and M. Conti, "Opportunistic Networking : Data Forwarding in Disconnected Mobile AdHoc Networks," *IEEE Communications Magazine*, vol. 44, no. 11, pp.134-141, Nov. 2006.
- [3] S. Ehsan, K. Bradford, M. Brugger, B. Hamdaoui, Y. Kovchegov, D. Johnson and M. Louhaichi, "Design and Analysis of Delay-Tolerant Sensor Networks for Monitoring and Tracking Free-Roaming Animals," *IEEE Transactions on Wireless Communications*, vol. 11, no. 3, pp.1220-1227, Mar. 2012.
- [4] C. Quadri, D. Maggiorini, S. Gaito and G. P. Rossi, "On the Scalability of Delay-Tolerant Routing Protocols in Urban Environment," *Wireless Days (WD), 2011 IFIP*, pp.1-6, Oct. 2011.
- [5] N. Uchida, N. Kawamura, G. Sato, and Y. Shibata, "Delay Tolerant Networking with Data Triage Method based on Emergent User Policies for Disaster Information Network System", *Mobile Information Systems*, vol. 10, no. 4, pp.347-359, Oct. 2014.
- [6] A. Vahdat and D. Becker, "Epidemic Routing for Partially-Connected Ad Hoc Networks," Duck Univ. tech. rep. CS-2000-06, 2000.
- [7] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic routing in intermittently connected networks," in *Service Assurance with Partial and Intermittent Resources*, Springer, Berlin Heidelberg, pp 239-254, 2004.
- [8] P. Hui, J. Crowcroft and E. Yoneki, "BUBBLE Rap: Social-Based Forwarding in Delay Tolerant Networks," *IEEE Transactions on Mobile Computing*, vol. 10, no. 11, pp. 1576-1589, Dec. 2010.



정래진(Rae-Jin Jeong)

2014년 한국기술교육대학교 컴퓨터공학부 학사
 2014년 ~ 한국기술교육대학교 창의융합공학협동과정 ICT융합전공 석사과정
 ※ 관심분야 : Delay Tolerant Network, Social Network, WSN, Ad-hoc network



전일규(Il-Kyu Jeon)

2013년 한국기술교육대학교 컴퓨터공학부 학사
 2014년 ~ 한국기술교육대학교 창의융합공학협동과정 ICT융합전공 석사과정
 ※ 관심분야 : Ad-hoc network, DTN, Ubiquitous computing



우병훈(Byeong-hun Woo)

2009년 ~ 한국기술교육대학교 컴퓨터공학부 재학
 ※ 관심분야 : Ubiquitous computing, WSN, Database



구남경(Nam-kyoung Koo)

2009년 ~ 한국기술교육대학교 컴퓨터공학부 재학
 ※ 관심분야 : Ubiquitous computing, Java Programming, Android, Database



이강환(Kang-Whan Lee)

1983년 한양대학교 전자공학과 학사
1989년 중앙대학교 전자공학 석사
1989년 한국전자통신연구원 선임연구원
2002년 중앙대학교 전자공학 박사
2004년 특허청 서기관
2005년 ~ 한국기술교육대학교 정보기술공학부 교수

※ 관심분야 : WSN, Ad-hoc network, 차세대이동통신기술, Wireless SoC