MANET에서의 상황 인지 구조를 이용한
에너지 보존 라우팅 연구

전운, 이범재, 지삼현, 김순국, 두정민, 이강환
Korea University of Technology and Education
winduser@kut.ac.kr

요 약
Ad hoc 셀시 네트워크에서 노드는 제한된 에너지를 가지며, 에너지의 사용량은 토폴로지 구조나 셀의 노드의 동작에 의존한다. 본 논문에서는 low power distributed MAC을 기반을 두고, MANET에서 clustering 구조의 네트워크를 위한 효과적인 에너지 절약을 위한 새로운 구조를 제안한다. 본 제안된 구조의 특징은 Ad hoc 네트워크에서 에너지에 관련된 메터리 파워다, 노드의 거리, 전송 전력 등의 정보를 활용한 상황 인식을 사용하였다. 에너지 상황 인식을 적용한 제안된 맵의 구조는 clustering 형상과 에너지 전송의 상황 및 데이타의도달 인식을 향상시킨다. 또한 context aware computing을 이용함으로써, 상황과 제한된 rule에 따라서 노드의 동작과 네트워크 라우팅의 상황에 적응하도록 노드를 조정 할 수 있는 구조이다.


I. 서 론

Ad hoc 네트워크는 동적 토폴로지를 가지는 모바일 자체 구조 망이다. 그리고 모든 노드는 라우터와 같은 역할을 할 수 있다. 이 Ad hoc 네트워크의 셀시는 독립적인 메터리 전원을 가지며, 그 에너지는 제한된다. 그러므로 노드의 수명을 늘리기 위해서는 가능한 에너지를 효율적으로 사용하는 값은 매우 중요한 논람이다.

네트워크의 수명과 출력을 늘리기 위해서 노드의 에너지 소비와 라우팅 링크 비용을 줄일 필요가 있다. 그러므로 어떤 네트워크 구조와 어떤 라우팅 path를 선택하느냐가 Ad hoc 네트워크에서 매우 중요하다.


Ad hoc 네트워크의 토폴로지는 노드의 장애 에너지량과 에너지 소비 비용이 유효적으로 변화하기 때문에 유효적인 다른 상황에 따른 구체적인 반응을 연속적으로 보여야 한다. 그래서 본 논문에서는 clustering head 노드에서 상황 인식 컴퓨터를 사용하여 라우팅 path을 결정한다.

본 논문에서는 새로운 Energy conserving Context aware Clustering algorithm(EEC)을 제안한다. 이 알고리즘은 전송 전력, 노드의 이동성 노드의 장애 에너지량과 같은 여러 에너지 요소를 종합적으로 고려하여 최소한의 이웃 노드 N에 대한 낮은 전송 전력을 가지며, 전력 에너지량이 높고, 낮은 이동성을 가진 노드를 cluster head로 선택할 수 있다.


본 논문은 다음과 같이 구성된다. 2장에서 MANET을 위한 상황 인지 구조를 기초로 rule을 보이며, 3장에서는 Energy conserving context aware clustering algorithm (EEC)와 the optimal energy routing protocol algorithm (OERP)에 대해 설명한다. 그리고 4장에서 결론으로 구성된다.

II. MANET에서의 에너지 절약 라우팅을
위한 상황 인지 구조

본 논문에서 제안하는 상황 인지 구조를 기초로
한 rule은 네트워크 clustering 형성과 네트워크 라우팅 실험을 위해 설계되었다.

본 논문에서 제안하는 상황 인지 구조는 ECA (Event Condition Action) rules [5]을 이용하며, 주 구조는 fig1에서 보는 바와 같다. 이 상황 인지 구조는 communication unit, processing unit, blackboard로 이루어져 있다. 본 논문에서는 이 구조에 오직 energy elements과 masking part를 이용한다. Communication unit은 외부 장치로부터 상황 정보 얻거나 컴퓨터 된 결과를 보낼 때 사용된다. Processing unit은 rule에 따라 상황 정보를 처리한다. 이 프로세서 예제는 Blackboard의 working 메모리와 상황 지식을 가지고 센서로부터 제일든 정보를 근간으로 판단 기준이 되는 임계값 결정 등을 처리 하는 기능 구조로 되어 있다. 또한 새로운 상황 정보를 얻게 되면 즉시 업데이트 하는 유동적인 정보 구조를 가진다.

![Fig. 1. 상황 인지 구조](image)

본 논문은 Clustering 구조 형성 및 유지보수와 에너지 절약을 위한 라우팅 path 선택, 상황 rule 설계를 위한 요소 정보를 고려해야 한다. 위의 고려사항은 다음 장에서 다루도록 한다.

III. ECC와 OERP 알고리즘

1. ECC알고리즘의 제안

이 장에서 ECC(context aware weighting clustering algorithm)에 대해 소개하겠다.

<table>
<thead>
<tr>
<th>상태</th>
<th>$E_i$</th>
<th>$P_i$</th>
<th>$W_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average $E_i$ is very low</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Average $P_i$ is high</td>
<td>1</td>
<td>0.5</td>
<td></td>
</tr>
<tr>
<td>Average $V_i$ is high</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

$E_i$, $P_i$, $V_i$는 일반적인 비트워크의 에너지 요소이다. 각각 노드의 잔여 에너지량, 전송 전력, 노드의 속력을 나타낸다. Table 2는 각기 다른 상태에서의 $E_i$, $P_i$, $V_i$가 다른 값을 갖는 것을 보여준다.

1.3 ECC algorithm rules

ECC algorithm rules는 다음과 같은 순서를 따른다.

**Step1.** 각 노드는 주변 노드 자신의 정보를 보내고, 수신모드로 대기한다.

**Step2.** $E_i$, $P_i$, $V_i$를 연산하여 ECC value $W_i$를 얻는다.

**Step3.** $W_i$를 연산하여, $W_i$가 $W_0$값을 초과할 경우, 노드는 부모노드가 되어, 이와 노드에게 부모 노드 선언 메시지를 보내고, cluster를 형성한다. 그 후 자신의 노드 정보를 수집하기 시작한다. 그러나, 가장 큰 $W_i$ 값을 가지지 않는다면 노드는 자신의 노드로 부모 노드로부터 선언 메시지를 받을 준비를 하며, 선언 메시지를 받으면, cluster ID를 설정하고
cluster 험버가 되어 주기적으로 부모 노드에게 자신 노드의 존재 여부를 보내게 된다.

Fig. 2. ECC cluster 구조

본 논문에서 제시한 상황 인식 구조는 각기 다른 요소에 고려하여 상황을 분류한다. 그 예로 Table 3은 6가지 요소에 고려한 상황 분류를 보여준다. 식 (1)에 따라, 얻어진 ECC W를 고려하여, 각 노드의 상태가 cluster head인지 cluster member인지 결정 한다. Table 3은 기초로 한 이 네트워크 cluster구조는 fig.2에서 보여지듯이, cluster 1에 노드 1이 부모 노드가 되고, 노드 2,3,4,5가 자식 노드가 된다. 노드 6은 이외 노드가 된다.

Table 3 상황 분류 형태

<table>
<thead>
<tr>
<th>ID</th>
<th>Et</th>
<th>Pt</th>
<th>Wi</th>
<th>Wt</th>
<th>상태</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>10</td>
<td>10</td>
<td>0.05</td>
<td>parent 노드</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>20</td>
<td>20</td>
<td>0.01</td>
<td>Child 노드</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>10</td>
<td>20</td>
<td>0.02</td>
<td>Child 노드</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>20</td>
<td>10</td>
<td>0.02</td>
<td>Child 노드</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>20</td>
<td>20</td>
<td>0.075</td>
<td>Child 노드</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>10</td>
<td>15</td>
<td>0.4</td>
<td>Neighbor 노드</td>
</tr>
</tbody>
</table>

1.4 ECC clustering 유지 보수

1.4.1 부모 노드의 유지 보수

\[ W_{th} = \text{avg}W \times \beta (0 < \beta < 1) \]

\( W_{th} \)가 식(2)처럼 정의되고, 만일 부모 노드의 ECC weighting Wi가 W, <Wth라면, 부모 노드는 부모 노드와 변경 메시지를 자신 노드에게 보내고 부모 노드를 변경 한다. 현 cluster에 부모 노드가 없으면, 그 자식 노드들은 가장 가까운 이웃 cluster에 참가한다.

만일 부모 노드가 자식 노드로부터 존재 여부 메시지를 받지 못하면, 부모 노드는 자신 노드 리스트를 업데이트 시키고, 그 자식 노드의 정보를 제거한다.

1.4.2 자식 노드의 유지 보수

자식 노드가 부모 노드로부터 부모 노드 변경 메시지를 받았다면 부모 노드를 변경 한다.

만일 자식 노드가 부모 노드까지의 신호 세기가 미리 정의된 Sa보다 적으면, 존재 여부 메시지를 보낼 수 없게 된다. 이 경우에는 근처의 새로운 cluster에 참여하게 된다. Fig.3은 ECC clustering의 형성과 유지 보수에 관련한 흐름도이다.

1. ECC cluster 네트워크를 위한 OERP algorithm (Optimal Energy Routing Protocol) 제안

Cluster기반 Ad hoc 네트워크에서 자식 노드로부터 부모 노드로 부터 부모 노드로의 전송이나, 부모 노드간의 전송은 각 cluster의 hop이 길어짐수록 데이터의 복잡도가 증가한다. 이것은 데이터의 전송량을 늘리고, 소비 전력 또한 증가시키는 문제점이 있다. 본 논문은 이런 문제점을 해결하기 위하여 OERP 알고리즘을 사용한다.

2.1 Routing path 방안

MANET에서의 에너지 보존을 향상시키기 위해서 소스 노드에서 목적지 노드까지의 다양한 라우팅 path를 계획 할 수 있다[3]. 에너지 요소들은 RREQ (Route require)에 의해진다. 본 논문은 에너지 보존을 고려한 라우팅 path 선정에 중점을 둔다. 현재의 대부분의 Ad hoc 네트워크의 에너지 보존 기반 라우팅 프로토콜은 MER[1]이나 OMM[2]와 같이 오직한 축면에서의 문제점을 고려한다. 본 논문에서 전파 에너지량과 링크 path의 에너지소비를 종합적으로 고려한 새로운 라우팅 알고리즘인 OERP를 제안한다. OERP에서 모든 소스로부터 목적지까지의 path를 P라 하고, 선택된 path를 path p라고 하며,
그 path p의 노드들은 Np라는 한다. (p∈P)
이 때 노드 i 수명 Ti는

\[ T_i = \frac{E_i}{q \sum_{j \in N_i} e_j} \]  (3)

노드 i의 에너지 소비 비율 hi는

\[ R_i = \frac{e_{hi}}{E_i} \]  (4)

path p 에너지 소비 비율 Rp는

\[ R_p = \frac{\min_{i \in p} R_i}{\sum_{i \in p} E_i} \]  (5)

\[ \min_{p \in P \mid R} \]를 path p의 R의 최소값이라 하고, \( \min_{P} \)을 path p의 T의 최소값이라 하면, Path p의 OERP value Up는

\[ U_p = \min_{i \in P \mid R} \frac{T_p}{R_p} + \min_{j \in P \mid R} R_j \]  (6)

### 2.2. OERP 알고리즘의 성능

MinE라고 path p의 E(i∈Np)의 최소값이고, path p의 간선 에너지량을 Ep(p∈P)라 했을 때 Ep최소값은 \( \min_{p \in P \mid R} \)라고 하고, Ep의 최대값을 maxEp라 가정한다.

Fig.4의 예제에서 노드 1을 소스 노드이고, 목적 노드는 노드 6이 할 때, 소스 노드 1에서 목적 노드 6까지의 path는 path1:1,3,4,6, path2:1,2,4,6, path3:1,2,5,6 총 3개까지 이다. Path 1의 \( \min E_i = 3 \), path2의 \( \min E_i = 2 \), path3의 \( \min E_i = 4 \)라고 하면, \( \min E_p = \min(3,2,4) = 2 \), \( \max E_p = \max(3,2,4) = 4 \) 이다.

![Fig. 4. A simple multipath Routing](image)

*Table.4 OERP element value*

<table>
<thead>
<tr>
<th>Path</th>
<th>( \min T_i )</th>
<th>( \min R_i ) (%)</th>
<th>1/Rp</th>
<th>( \min E_p )</th>
<th>( U_p )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path1</td>
<td>30</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>40</td>
</tr>
<tr>
<td>Path2</td>
<td>20</td>
<td>10</td>
<td>5</td>
<td>2</td>
<td>35</td>
</tr>
<tr>
<td>Path3</td>
<td>20</td>
<td>5</td>
<td>7</td>
<td>4</td>
<td>32</td>
</tr>
</tbody>
</table>

Step1. 노드가 RREQ (Route require)받을 때, myID=Dest라 할 때,
A. No, RREQ가 myID를 더했는 후 \( R_i \, T_i \) 연산하여, RREQ R, \( T_i \) E를 다닌다.
B. Yes, 노드는 가까운 가까운 후 multipath P에 대한 OERP값 \( U_p \) 연산하고 식(7)에 따라 path P'를 선택한다.

\[ E_p = \frac{\min E_p + \alpha (\max E_p - \min E_p)}{0 \leq \alpha \leq 1} \]  (7)

\( \alpha \)은 \( \max(1,0) \)에 의해, path p(p∈P')는 최우선 path로 선택된다.

\[ U_p = \max \{ U \} \]  (8)

Step2. 소스 노드로부터 보내진 메시지를 응답을 보낸다.

Table.4의 단순한 multipath 구조에서 각 노드로부터 보내진 응답 메시지를 기초로 얻어진 path별 \( \min T_i \), \( \min E_i \), \( 1/Rp \) 시스템으로 얻어진 \( U_p \)이 \( \min E_i \)를 포함하고 있다. 여기서 \( \alpha = 1 \)가 가장했을 때, \( \min E_p = 2 \)이므로 식(7)에 의해 \( E_p = 2 \)가 된다. 그러므로 P'=[path1, path3]가 되며, 식(8)에 의해 path1이 최우선 path로 선택된다.

### IV. 결론

본 논문에서 Ad hoc 네트워크를 위한 energy conserving context aware clustering algorithm (ECC)와 Optimal Energy Routing Protocol (OERP)을 제시했다. 제시된 ECC를 사용한 OERP 알고리즘은 Ad hoc 네트워크에서 에너지 소비를 줄이기 위해, 에너지 요소를 종합적으로 고려한 알고리즘이다.

제시된 ECC를 적용한 OERP는 네트워크 합포로지와 노드와 야간에 에너지량을 알 필요가 있다. 이것은 모든 노드가 자신의 야간 에너지량을 다른 노드에게 알려야 하므로, 거대한 네트워크에서는 적합하지 않다. 하지만 Ad hoc 네트워크와 같은 소규모 네트워크에서는 기존의 알고리즘보다 효율적인 에너지 관리를 통해 네트워크의 수명을 연장할 수 있다.

참고 문헌


[8] WANG Haitao, ZHANG Shaoren, SONG Lihua, “Architectures and clustering algorithms for mobile Ad hoc network”.
