

# Ad-hoc 네트워크 환경에서 DSDV 라우팅 알고리즘을 이용한 위치 정보 시스템 구현

임효영, 이정구, 곽종욱\*  
영남대학교 컴퓨터공학과  
e-mail:hyoyoung\_@naver.com

## Implementation of Location Information System using DSDV Routing Algorithm in Ad-hoc Network Environment

Hyo Young Lim, Jeong Gu Lee, Jong Wook Kwak  
Department of Computer Engineering, Yeungnam University

### 요 약

본 논문에서는 다수의 이동 가능한 관리 대상을 소수의 관리자가 효율적으로 관리하기 위하여 Ad-hoc 네트워크 환경에서 DSDV(Destination Sequenced Distance Vector) 라우팅 알고리즘을 활용하는 위치 정보 시스템을 구현한다. 제안된 시스템에서는 다수의 위치 정보 노드들이 Ad-hoc 네트워크 환경에서 DSDV 라우팅 알고리즘으로 네트워크를 구축하고, 구축된 네트워크에서 하나 이상의 위치 정보 노드가 사라지면 진동과 같은 경고 시스템을 활성화한다. 또한 제안된 시스템은 안드로이드 환경에서 블루투스 통신을 통해 시각적으로 노드들의 위치를 확인시켜 주어 관리 대상에 대한 관리를 도와준다. 본 시스템으로 모의 성능 평가를 수행한 결과, 사용자 위치 정보 노드 간 1:1 통신을 직선거리에서 수행하였을 때, 약 250m까지 통신이 가능하였으며, 1:N 통신의 경우 약 100m 이내의 거리에서 정상적으로 통신이 이루어지는 것을 확인하였다. 본 논문에서 구현된 시스템은 유지된 아동 관리 시스템, 놀이공원 미아 방지 시스템, 관광객 관리 시스템 등 여러 시스템에 매우 유연하게 적용이 가능하며, 그 활용도가 높을 것으로 예상된다.

### 1. 서론

최근 주목 받고 있는 무선 네트워크의 기술 중 하나인 Ad-hoc 네트워크는 기지국이나 AP(Access Point) 없이 무선으로 통신이 가능한 노드들끼리 서로 정보를 주고받는 네트워크 구조로서, 최근 관련 연구들이 활발하게 진행되고 있다. 이에 본 논문에서는 Ad-hoc 네트워크 환경을 활용하는 사용자 위치 정보 시스템을 구현한다. 유지된 관리 시스템이나 미아방지 시스템과 같이, 소수의 관리자가 다수의 이동 가능한 관리 대상을 관리하는 방법으로는 관리자가 관리 대상을 시야에 두고 일일이 확인하면서 이탈하지 않도록 하는 것이 주된 해결책이다. 하지만 이는 지극히 제한적이며, 여러 돌발 상황이 발생 할 수 있다.

이를 위해 본 논문에서는 다수의 관리 대상을 소수의 관리자가 효율적으로 관리 할 수 있도록 Ad-hoc 네트워크를 이용하여 다수의 사용자 위치 정보 노드들이 네트워크를 구성하고 위치 정보를 안드로이드 단말을 통하여 확인하도록 한다. 또한 하나 이상의 사용자 위치 정보 노드가 기 구축된 네트워크에서 사라지면 진동이나 알람과 같은 경고 시스템을 활성화시켜 사라진 사용자 정보 노드의

위치 파악에 도움을 주는 기능을 제공한다. 이하 본 논문의 구성은 다음과 같다. 2장에서는 본 논문의 관련연구 및 배경지식에 대해 알아보고, 3장에서는 전체 시스템 구성과 구체적인 구현 방식, 그리고 실제 동작 과정에 대해 설명한다. 4장에서는 개발 및 성능평가 환경을 소개하고, 본 논문에서 제안한 시스템의 성능에 대해 분석하고 평가한다. 마지막으로 5장에서는 본 논문에서 구현한 시스템에 대한 결론에 대해서 논의한다.

### 2. 관련연구 및 배경지식

#### 2.1 무선 네트워크

무선 네트워크(Wireless Network)란, 무선 전파를 이용하는 통신 네트워크를 통칭하는 말이다. PC나 PDA와 같은 단말기를 통해 접속되는 경우가 많으며, 무선 인터넷 규약에 의해 사용하는 WAP와 무선 랜(LAN)을 사용하는 방법이 대표적이다[1].

무선 네트워크의 종류는 기존 통신 인프라의 지원 여부에 따라서 두 가지로 분류한다. 이동 단말들이 유선 환경에 기반을 둔 기지국이나 AP를 중심으로 구성되는 Infrastructure 네트워크와 노드들로 구성된 인프라가 없는 Ad-hoc 네트워크로 분류 할 수 있다.

\* 교신저자(corresponding author)



(그림 1) 시스템 구조

2.2 Infrastructure 네트워크

Infrastructure 네트워크는 망의 중단 부분이 무선으로 유선 네트워크와 연결되어 Last Hop Network라 한다. 기존에 설치된 인프라를 통해서만 데이터의 송수신이 이루어지는 구조로 현재 많이 접하고 있는 이동 전화망이나 무선 LAN 등을 예로 들 수 있다. 인프라가 있는 네트워크는 기존 인프라를 통해서만 통신이 가능하므로 지진과 같은 재해나 테러 혹은 전쟁과 같은 상황에서 기지국이나 AP의 고장, 유선 단절과 같은 상황 발생 시 원활한 통신을 할 수 없다는 단점이 있다.

2.3 Ad-hoc 네트워크

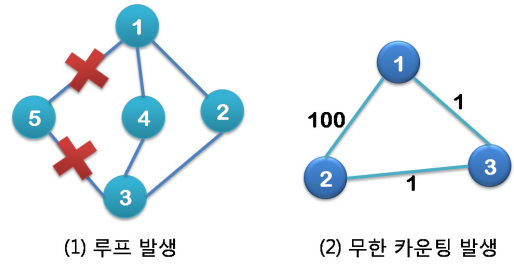
Ad-hoc 네트워크는 중앙 집중화된 관리가 지원되지 않는 환경에서 이동성 지원 기반 시설의 도움 없이 임시로 네트워크를 구성하는 이동 호스트와 무선 인터페이스의 집합이다[2]. 이러한 네트워크 구조에서 각 이동 호스트는 단말 호스트이면서 또한 라우터로 동작하여 패킷을 다른 이동 호스트로 전달한다. 이 때, 이동성 지원 기술은 위치 정보가 서로 다른 하드웨어 특성 영역 혹은 서로 다른 이동 통신망 사이에서 단말의 이동을 추적하고, 필요하다면 네트워크 구성 요소 사이에 상호 전달하는데 소요되는 방법론을 통칭한다. Ad-hoc 네트워크는 기존 인프라가 필요하지 않기 때문에 임시 구성용 네트워크나 재난 지역과 전쟁터와 같은 기반 시설이 없는 환경에서 적용가능토록 주로 군사용 망에 중점을 두어 연구 개발 되었으나, 최근 원거리 네트워크뿐만 아니라 근거리 로컬 네트워크에서도 상업적으로 주목 받기 시작했다[3].

3. 위치 정보 시스템의 구현

3.1 시스템 구조

(그림 1)는 본 논문에서 제안하는 시스템의 전체적인 구조도이다. (그림 1)에서 사용자 위치 정보 노드는 본 시스템에서 구현한 하드웨어 디바이스를 지칭하는 것이다.

각 사용자 위치 정보 노드에 FreeRTOS를 포팅하고 DSDV 라우팅 알고리즘을 구현하여 자체적으로 네트워크를 형성한다. 노드 간에 구축된 네트워크는 안드로이드 환경에서 블루투스 통신을 통해 정보를 상호 교환한다.



(그림 2) DV 알고리즘의 문제점

3.2 DSDV 알고리즘을 이용한 위치 정보 시스템

3.2.1 DV 라우팅 알고리즘

DV(Distance Vector) 알고리즘의 특징은 다음과 같다. 우선, 모든 노드들은 라우팅 테이블을 유지하면서, 주기적으로 라우팅 테이블을 모든 이웃들에게 전송한다. 이 때, 홉 수가 Cost가 되어 홉 수가 최소인 최적의 경로를 찾는다. 라우팅 테이블에서 Destination은 도착 가능한 모든 목적지이며, Nexthop은 목적지에 최적으로 도착하기 위해 거쳐 가야하는 노드이고 Hopcount는 목적지까지의 홉 수이다. 하지만 DV 알고리즘은 (그림 2)에서 보이는 것처럼 루프 또는 무한 카운팅을 발생 시킬 수 있다는 문제점을 가지고 있다.

3.2.2 DSDV 라우팅 알고리즘

DSDV(Destination Sequenced Distance Vector) 라우팅 알고리즘은 앞절에서 서술된 DV 알고리즘의 개선된 버전으로 다음과 같은 특징을 가진다[4].

우선, 모든 노드들은 라우팅 테이블을 유지하면서, 주기적으로 라우팅 테이블을 모든 이웃들에게 전송한다. 이 때, 순서 번호를 이용하여 루프가 발생하는 것을 방지하며, 각 노드들은 더 큰 순서 번호의 정보로 갱신한다. 만약 특정 노드가 네트워크상에서 연결이 끊어지면 Metric (Number of Hops to Destination)을 무한대로 설정하고, 순서 번호를 증가 시킨 후 주위 노드들에게 정보를 송신한다. 주위 노드들에게 보내는 라우팅 정보는 Destination, Metric, Destination Sequence Number이다. 순서 번호는 각각의 라우팅 정보 전송 시 자신의 순서 번호를 증가 시키고 만약 노드가 최종 목적지에 도달할 수 없다면 순서 번호를 1 증가시키고, Metric을 무한대로 한다. 라우터 정보는 높은 목적지의 순서 번호가 도달하면 이를 갱신하며, 같은 순서 번호가 온다면 더 적은 Metric을 선택 한다. 마지막으로 새로운 노드가 추가되거나 연결이 끊어질 경우 Metric의 변화는 즉시 이웃들에게 전파한다. <표 1>은 DSDV 알고리즘에서 사용하는 라우팅 테이블의 예제이다.

<표 1> 라우팅 테이블 예제

Dest	Next	Metric	Sequence Num	Install Time
A	A	0	A-550	1000
...	...	...	...	...
C	B	2	D-588	1200
D	B	3	D-312	1200

<표 2> 패킷 정의

Felid	Start Code	Sour ID	Next ID	Dest ID	Length	Type	Data	Check Sum
Size	5byte	1byte	1byte	1byte	1byte	1byte	Variable	1byte

3.2.3 패킷 정의

사용자 위치 정보 노드가 송수신하는 데이터 패킷의 구조는 <표 2>과 같다.

먼저 Start Code는 데이터 전송의 시작을 알리는 패킷이며 5byte로 구성되어 있다. Sour ID와 Next ID, Dest ID 필드는 각각 송신지의 ID, 라우팅 테이블에서 목적지에 도달하기 위해 거쳐 가는 다음 노드 ID, 목적지의 ID이며, 각각 1byte이다. Length는 1byte이며 Data길이 + 1(Type)의 값을 가진다. Type은 GPS 정보, GPS 정보 ACK, 라우팅 패킷, 라우팅 경로 요청, 라우팅 경로 응답으로 총 다섯 개로 나뉜다. Data 패킷은 실제 사용할 Data 값을 갖고 있으며 Type에 따라 크기가 유동적이다. Check Sum은 Length부터 Data까지의 합에 반전을 취한 값을 사용하며 크기는 1byte이다.

3.3 안드로이드 어플리케이션

3.3.1 블루투스 통신

안드로이드 환경에서 블루투스 통신을 수행 할 때 먼저 설정하여야 할 부분은 *AndroidManifest.xml* 파일에서 퍼미션을 지정하는 부분이다. 이 때, *BLUETOOTH\_ADMIN* 퍼미션을 설정 한 후에 *BLUETOOTH* 퍼미션을 지정하여야 정상적으로 동작한다[5].

다음으로 블루투스 UUID를 설정해야한다. 해당 어플리케이션에 맞는 UUID를 설정 해주어야 한다. 본 논문에서 제안한 시스템의 경우는 SSP(Serial Port Profile)를 이용한다. 따라서 이에 맞는 UUID인 "00001101-0000-1000-8000-00805F9B34FB"를 사용한다.

3.3.2 사용자 인터페이스

안드로이드 사용자 인터페이스는 (그림 3)에 나타나 있다. 메인 화면에서 위치를 정하고 블루투스 연결을 한 다음 지도를 보는 화면이 나타난다. 안드로이드 단말기와 위치정보 노드와의 블루투스 통신을 통해 실시간으로 노드의 위치를 확인 할 수 있으며, 노드가 구축된 네트워크상에서 이탈 될 경우 안드로이드 단말기에서는 이를 알려주는 이벤트가 발생한다. 만약 자신의 노드가 네트워크를 이탈 했을 경우 다이얼로그로 알릴 수 있게 구현하였다[6].

4. 성능평가

4.1 개발 환경

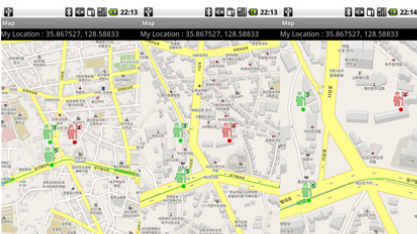
본 논문에서 구현한 시스템의 개발 환경은 <표 3>과 같다. Operating System은 Windows XP SP3과 Windows 7을 사용하였다. 개발 툴은 Inventor 2008, PADS2007, Orcad10.5, IAR Embedded Workbench 5.41, Android SDK 2.1v, Eclipse Galileo를 사용하였다. 개발 언어는 사용자 위치 정보 노드에서 C를 사용하였고, 안드로이드 어플리케이션은 JDK 1.6을 사용하여 구현하였다.

<표 3> 개발 환경

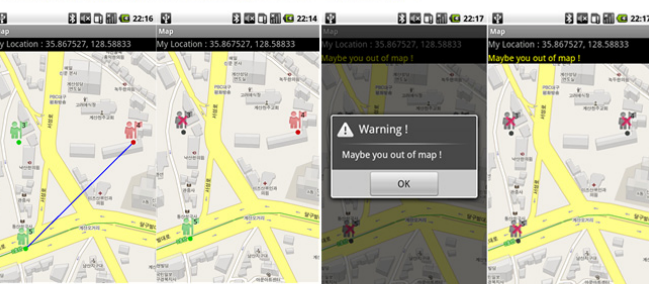
OS	Windows XP SP3 Windows 7
Tools	Microsoft Visual Studio 2008 Inventor 2008 PADS2007 Orcad10.5 IAR Embedded Workbench 5.41 Android SDK 2.1v Eclipse Galileo
Language	C Java JDK 1.6



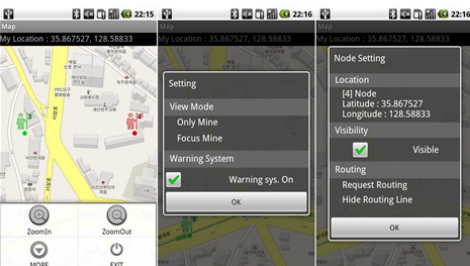
(1) 메인 화면



(2) 노드 위치 및 지도



(3) 자신 혹은 다른 노드가 맵을 이탈 하였을 때



(4) 설정화면

(그림 3) 안드로이드 단말의 상황별 사용자 인터페이스

&lt;표 4&gt; 모의 성능 평가 결과

1:1	노드 개수	최대 거리	비고
	2개	250m	직선
1:N	노드 개수	노드 간 거리	비고
	3개	100m	직선
	3개	100m	삼각형
	4개	75m	┌형
	6개	50m	육각형

## 4.2 모의실험 결과

### 4.2.1 사용자 위치 정보 노드 간 1:1 통신

본 논문에서 구현한 시스템으로 두 대의 사용자 위치 정보 노드 간 1:1 통신을 직선거리에서 수행한 결과, <표 4>에 나와 있는 것처럼 약 250m 까지 정상적으로 동작하는 것을 확인하였다. 전체적으로는 중간에 하나 이상의 노드가 사라질 경우, 진동이나 알람으로 경고 시스템을 활성화하며, 이 경우도 직선으로 최대 거리 250m 안에 노드가 있는 경우 전체적인 통신이 끊어지지 않고 데이터를 주고받는 것을 확인하였다.

### 4.2.2 사용자 위치 정보 노드 간 1:N 통신

본 논문이 구현한 시스템으로 하나의 사용자 위치 정보 노드가 여러 노드와 통신을 통해 데이터를 주고받을 수 있는지를 확인하는 사용자 위치 정보 노드 간 1:N 통신을 수행한 결과는 <표 4>에서 확인 할 수 있으며 자세한 결과 분석은 다음과 같다.

먼저, 사용자 위치 정보 노드 간 약 100m 이내의 거리에서 직선으로 3개의 사용자 위치 정보 노드로 통신을 수행한 결과 정상적으로 데이터가 송·수신 되는 것을 확인하였다. 다음으로, 약 100m 이내의 거리에서 삼각형 형태로 3개의 사용자 위치 정보 노드로 통신을 수행한 결과 정상적으로 데이터를 주고받는 것을 확인하였다. 세 번째로, 노드 간 약 75m 이내의 거리에서 4개의 노드로 '┌' 형태로 통신한 결과 역시 정상적으로 데이터를 주고받는 것을 확인하였으며 마지막으로, 노드 간 약 50m 이내의 거리에서 6개의 노드로 통신한 결과, 정상적으로 데이터를 송·수신하는 것을 확인 하였다.

## 5. 결론

본 논문에서는 그룹 내에 존재해야 하는 다수의 이동 가능 대상들을 관리할 때 이에 대한 제어를 효율적으로 할 수 있게 도와 줄 수 있는 시스템으로, Ad-hoc 네트워크 환경에서 DSDV 라우팅 알고리즘을 이용한 위치 정보 시스템을 구현하였다.

본 논문에서 구현한 위치 정보 시스템은 단말 장치 사이에 서로 데이터를 송·수신하며 상호 위치를 파악하여 독립적인 네트워크를 구성한다. 만약 하나 이상의 사용자 위치 정보 노드가 기 구축된 네트워크에서 사라질 경우 진동 혹은 알람 등으로 상황을 알린다. 또한 안드로이드

환경을 지원하는 단말기와의 블루투스 통신을 통해 실시간으로 노드의 위치를 확인 할 수 있게 하였으며, 사용자 정보 노드가 네트워크에서 이탈될 경우 안드로이드 단말에서도 이를 확인할 수 있도록 구현하였다. 본 논문에서 제안된 시스템의 성능을 평가한 결과, 1:1 통신은 최대 250m 직선거리에서 통신이 가능하며, 1:N 통신 및 다양한 형태의 노드 구성에 있어서도 정상적으로 데이터를 송·수신하는 것이 가능하였다.

본 논문에서 구현한 시스템은 3G 혹은 WI-FI와는 다른 독립적인 네트워크를 구성하기 때문에 유치원 아동 관리 시스템, 놀이공원 미아 방지 시스템, 관광 가이드의 관광객 관리 시스템 등 여러 시스템에 유연하게 적용 가능하므로 활용 방안이 다양할 것으로 예상된다.

## 참고문헌

- [1] 강문식, "데이터 통신과 컴퓨터 네트워킹", 한빛미디어, 2009
- [2] Thomas Kunz, S. S. Ravi, "Ad-hoc, Mobile, And Wireless Networks", Springer-Verlag New York Inc, 2006
- [3] 김창환, "Ad-hoc 네트워크 기술 동향", 전자정보센터
- [4] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing(DSDV) for Mobile Computers," Proc. of ACM SIGCOMM, 1994
- [5] <http://developer.android.com/resources/samples/BluetoothChat/index.html>
- [6] Frank Ableson, Charlie Collins, Robi Sen, "Unlocking Android", freelec, 2010