NS3 - 과제 3

20186889 권용한

- 1. 각 mobility model 에 대한 설명을 정리해서 pdf로 제출
 - constant-acceleration-mobility-model

현재의 가속도가 변화하지 않는 Mobility를 위한 모델입니다.

가속도가 설정되면 명시적으로 새 값이 다시 설정되기 전까지는 변경되지 않습니다.

• constant-position-mobility-model

현재의 위치가 변화하지 않는 Mobility를 위한 모델입니다.

위치가 설정되면 명시적으로 새 값이 다시 설정되기 전까지는 변경되지 않습니다.

constant-velocity-mobility-model

현재의 속도가 변화하지 않는 Mobility를 위한 모델입니다.

속도가 설정되면 명시적으로 새 값이 다시 설정되기 전까지는 변경되지 않습니다.

• gauss-markov-mobility-model

Gauss-Markob Mobility를 위한 모델입니다.

다른 mobility 모델과는 달리 memory와 variablity를 가집니다.

알파 parameter는 얼마나 많은 메모리와 랜덤성을 결정합니다.

각 객체는 mean속력, 방향, pitch angle에 상응하는 특정 속력과 방향, pitch angle를 가지고 시작하며, timestep마다 새로운 속력, 방향, pitch angle이 현재 값과 평균, gaussian random variable에 기반해 생성됩니다.

간단한 비행기와 같은 모델에 적합합니다.

• random-direction-2d-mobility-model

객체의 움직임이 random direction에 기반하도록 하는 모델입니다.

각 객체가 특정 딜레이마다 멈추고 랜덤 방향과 속력을 선택해 경계 값에 도달할 때까지 특정 방향으로 이동합니다.

경계에 도달하면 새로운 방향과 속력을 선택합니다.

• random-walk-2d-mobility-model

각 instance가 고정된 거리를 이동하거나 고정된 시간이 지날때까지 사용자가 제공한 random 변수를 바탕으로 랜덤으로 선택된 속력과 방향으로 이동합니다.

만약 경계에 도달하면 반사각과 속력으로 반동합니다.

• random-waypoint-mobility-model

각 객체는 time zero 에서부터 'pause' 랜덤 변수에 의해 통제되는 기간동안 멈추면서 시작합니다.

Pausing이 끝난 후 객체는 새로운 waypoint를 선택하고 새로운 랜덤 속도를 랜덤변수 'Speed'에 기반해 선택합니다.그리고 waypoint를 향해 고정된 속도로 이동합니다.

목적지에 도달하면 이 과정을 다시 시작합니다.

• steady-state-random-waypoint-mobility-model

이 모델은 속도, pause, 위치가 uniformly distributed random variable일 때의 random waypoint mobility model입니다.

차이는 parameter의 초기 값이 uniform distibution이 아니고 random waypoint mobility의 stationary distribution입니다.

이 모델은 2d specific하며, 노드의 속도는 0이 아닙니다.

• waypoint-mobility-model

각 객체는 Waypoint 개체 집합에서 지정된 시간에 속도와 위치를 결정합니다.

과거 waypoint는 현재 simulation time 후에 소모됩니다.

Default로 각 객체의 초기 위치는 first waypoint에 대응되며, 각 객체의 초기 속도는 0입니다. 마지막 waypoint에 도달하면, 객체의 위치는 고정되고 속력은 0이 됩니다.

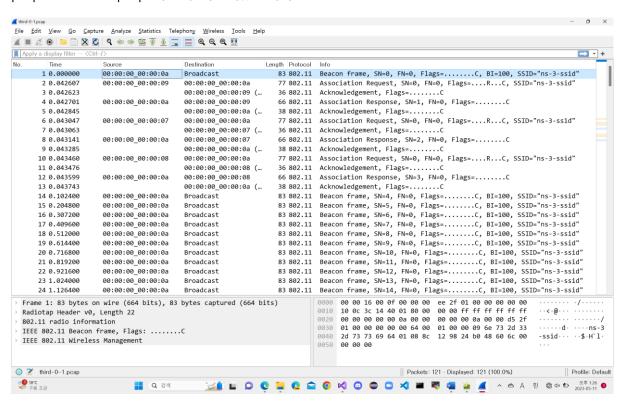
노드가 waypoint time 사이에 있다면, 고정된 속력으로 과거 waypoint와 현재 waypoint 사이를 이동합니다. 노드를 특정 위치에 고정시키기 위해서는 같은 위치에 두개의 waypoint를 설정해야 합니다.

2. Wifi AP 노드 pcap 분석

실험 결과

```
yhliSyhliSyhliSi-Instron-3670:-/workspace/ns-allinone-3.35/ns-3.355 ./waf --run "scratch/mythird --tracing=true"
this control of the control
```

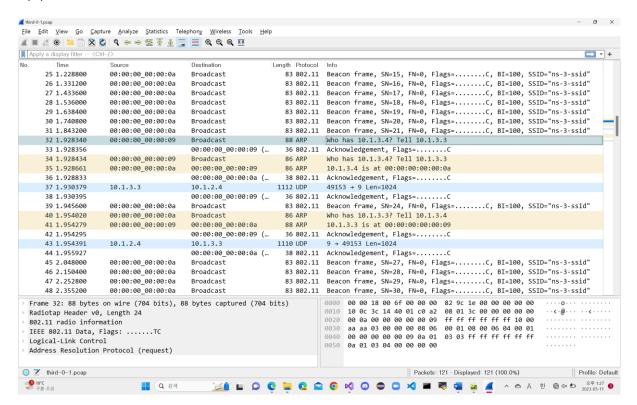
또한 p2p 연결된 모든 노드의 pcap을 구하였습니다. 이중, 0번 노드가 wifi AP이며, 해당 pcap은 third-0-1.pcap 파일에 저장되었습니다.



시작과 동시에 broadcast로 신호를 보내고(No.1), 이를 통해 3개의 단말이 AP와 통신하며 연결된 것을 확인할 수 있습니다. (No.2~No.13)

이후 0.1초 간격으로 꾸준하게 AP에서 broadcast로 신호를 보내는 것을 확인할 수 있습

니다.



이후 2초에 10.1.3.3(노드7, Wifi 연결된 단말)에서 10.1.2.4(노드4, LAN(csma) 연결된 단말)로 데이터를 보내며, 때문에 ARP를 통해 10.1.3.3(Wifi 단말)와 10.1.3.4(Wifi AP)가 서로의 MAC주소를 전달받고 통신을 진행하는 것을 확인할 수 있습니다.

(2초에 데이터를 전송하므로, 1.9초 즈음 ARP를 보내는 것을 확인할 수 있습니다.)

이후 약 10초까지 0.1초 간격으로 AP에서 broadcast로 신호를 보내며, 프로그램이 종료 하게 됩니다.