

IETF-115

IPMON Hackathon Project

November 6, 2022

Champions: Jaehoon (Paul) Jeong and Yiwen (Chris) Shen

Members: Bien Aime Mugabarigira, Junhee Kwon, hyeonah Jung, and yoseop Ahn

Department of Computer Science and Engineering at SKKU

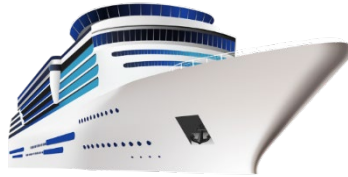
Email: {[pauljeong](mailto:pauljeong@skku.edu), [bienaime](mailto:bienaime@skku.edu), [juun9714](mailto:juun9714@skku.edu), [hyeonah214](mailto:hyeonah214@skku.edu), [ahnjs124](mailto:ahnjs124@skku.edu)}@skku.edu,
chrisshen@ks.ac.kr

IPv6 Moving Object Networking (IPMON)

- IPMON aims at the provisioning of IPv6 networking for moving objects such as terrestrial, aerial, and marine vehicles.
- IPMON fills in the gap of IPv6-related standards to provide those vehicles with the communication among them or with infrastructure nodes for the Internet connectivity.
 - **IPMON Communication Types:** V2X, V2V, and V2I
- IPMON considers wireless multihop communication, high-speed mobility, and optimal packet routing in a temporary network topology.
 - IPMON BoF aims at **developing protocols for moving objects** with **IPWAVE WG's Problem Statement and Use Cases Draft**:
 - [draft-ietf-ipwave-vehicular-networking-30](#)

Moving Objects (MOs) in IPMON

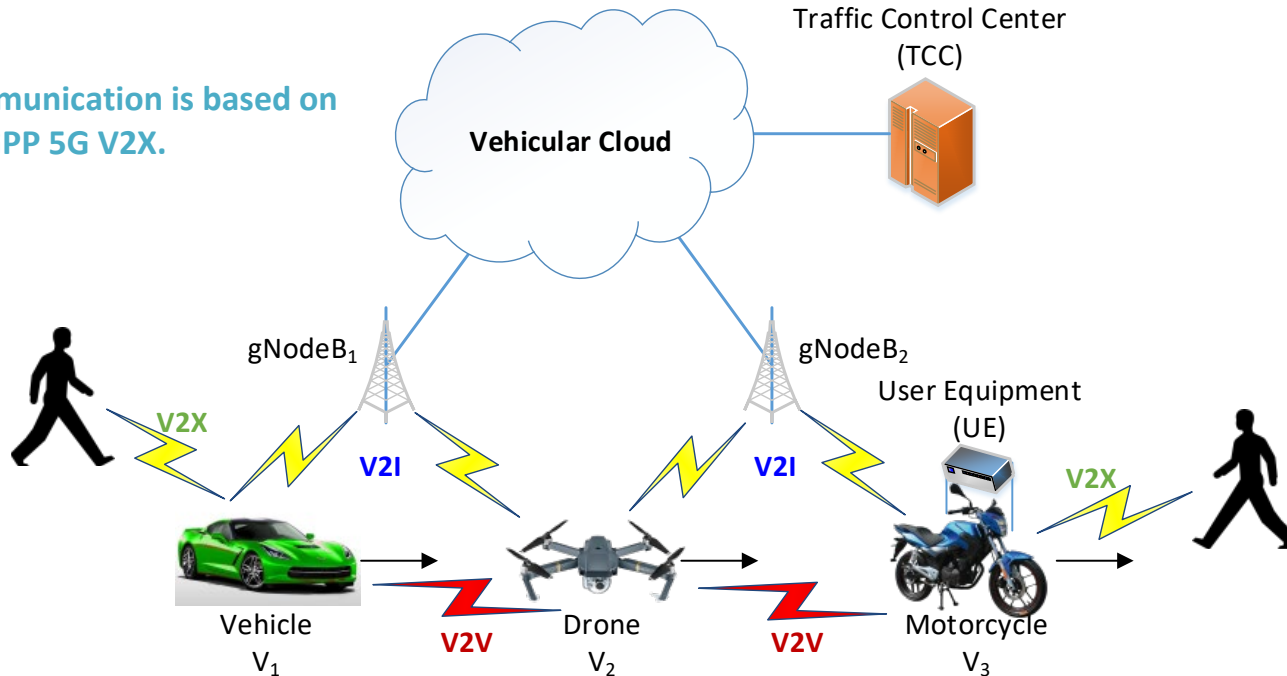
- Vehicle, Motorcycle, Scooter, Pedestrian, Unmanned Aerial Vehicle (UAV), Drone, Urban Air Mobility (UAM), Train, Subway, Boat, Ship, etc.



Vehicular Network Architecture for IPMON

- An IPMON Network consists of Vehicular Ad Hoc Networks (VANET) and Access Networks (AN).
- Wireless Communications supports IEEE 802.11-OCB/802.11-bd, and 3GPP 5G V2X.

Wireless Communication is based on
3GPP 5G V2X.



IPv6 Moving Object Networking (IPMON) Project

Champions: Jaehoon (Paul) Jeong (SKKU) and Yiwen (Chris) Shen (KSU)

IETF-115 IPMON Hackathon Project:
Context-Aware Navigation Protocol (CNP)



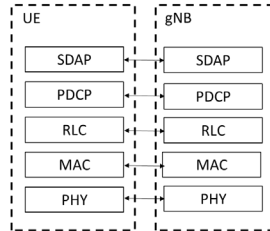
Professors:

- Jaehoon (Paul) Jeong (SKKU)
- Younghan Kim (SSU)
- Yiwen (Chris) Shen (KSU)

Students:

- Bien Aime Mugabarigira (SKKU)
- Junhee Kwon (SKKU)
- Hyeonah Jung (SKKU)
- Yoseop Ahn (SKKU)

5G NR Protocol Stack



5G Drone Networks



Objectives

- To Demonstrate IPMON Basic Protocols
- Context Aware Navigation Algorithm (CNP) for heavy-traffic drone networks
- Drone networks based on 5G V2X

IETF IPMON CNP Draft:

- URL: <https://datatracker.ietf.org/doc/html/draft-jeong-ipwave-context-aware-navigator-06>

Where to get source code:

- GitHub: <https://github.com/ipwave-hackathon-ietf>

How to set up an environment:

- OS: Ubuntu 20.04
- SUMO 1.8.0
- OMNeT++ 5.6.2
- GNU GCC7.3
- INET 4.4

Implementation Contents:

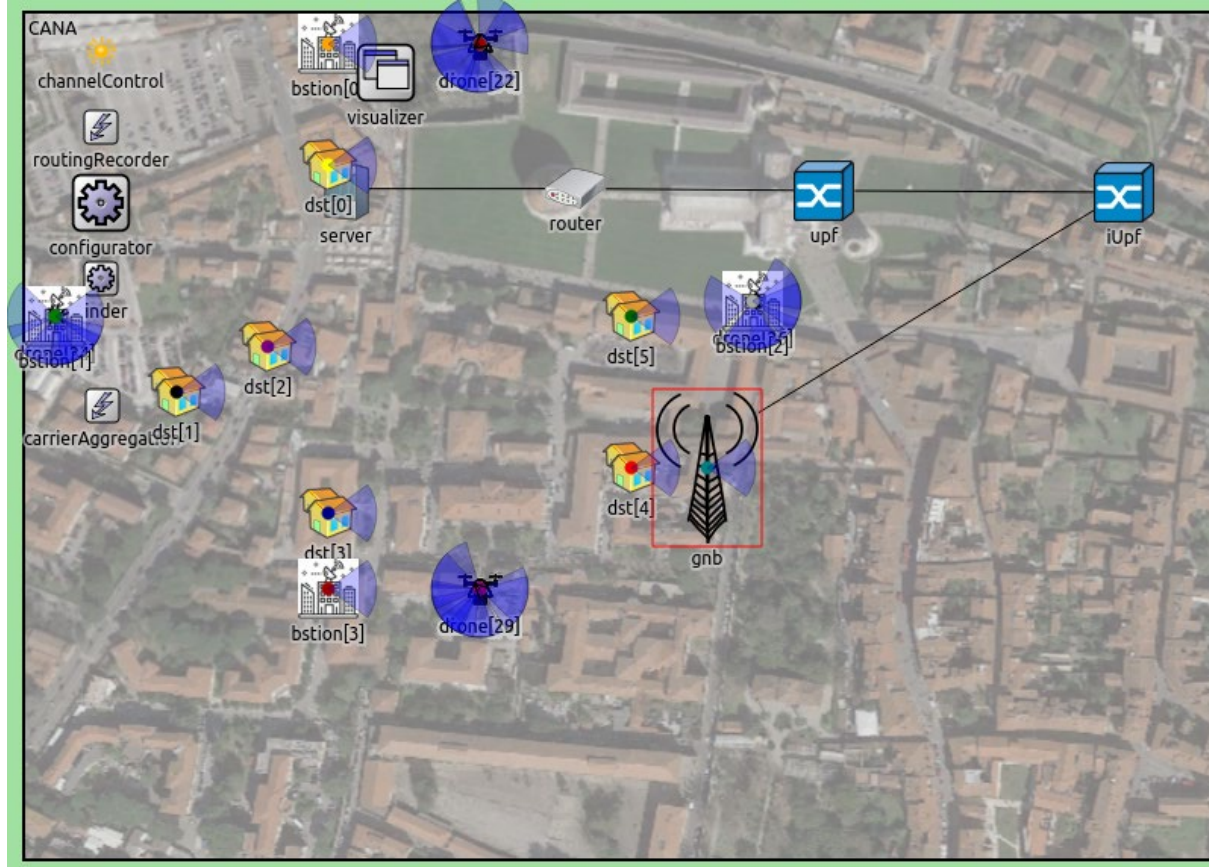
- To extend the CNP implementation to Drones, Personal Mobility (PM), Motorcycles as well as Vehicles
- To change the previously implemented LTE-V2X CNP simulation into 5G-V2X one



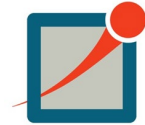
Hackathon Plan

- Draft for this Project
 - Context-Aware Navigation Protocol for IP-Based Vehicular Networks
 - draft-jeong-ipwave-context-aware-navigator-06
 - <https://datatracker.ietf.org/doc/draft-jeong-ipwave-context-aware-navigator/>
- Simulation
 - Context Aware Navigation Algorithm (CNP) for heavy-traffic drone networks
 - Drone networks based on 5G V2X

What got done (1/2)



OMNeT++





What got done (2/2)

- We checked whether the 5G communication feature of the 5G-Simu simulation can be used for CNP or not.
- Based on the 5G-Simu, we newly implemented our CNP simulation on it.



Open Source

URL: <https://github.com/ipwave-hackathon-ietf/ipwave-hackathon-ietf-115>

Search or jump to... Pull requests Issues Marketplace Explore

ipwave-hackathon-ietf / ipwave-hackathon-ietf-115 Private Watch 1 Fork 0 Star 0

<> Code Issues Pull requests Actions Projects Security Insights Settings

main 1 branch 0 tags Go to file Add file Code

File/Folder	Commit	Time
권준희 first commit f5b1b21 19 seconds ago 1 commit		
bin	first commit	19 seconds ago
emulation	first commit	19 seconds ago
images	first commit	19 seconds ago
simulations	first commit	19 seconds ago
src	first commit	19 seconds ago
tests/fingerprint	first commit	19 seconds ago
.gitignore	first commit	19 seconds ago
.nedexclusions	first commit	19 seconds ago
.nedfolders	first commit	19 seconds ago
.oppbuildspec	first commit	19 seconds ago
.oppfeatures	first commit	19 seconds ago
.oppfeaturesstate	first commit	19 seconds ago
.pydevproject	first commit	19 seconds ago
INSTALL.md	first commit	19 seconds ago
LICENSE.md	first commit	19 seconds ago
Makefile	first commit	19 seconds ago
README.md	first commit	19 seconds ago

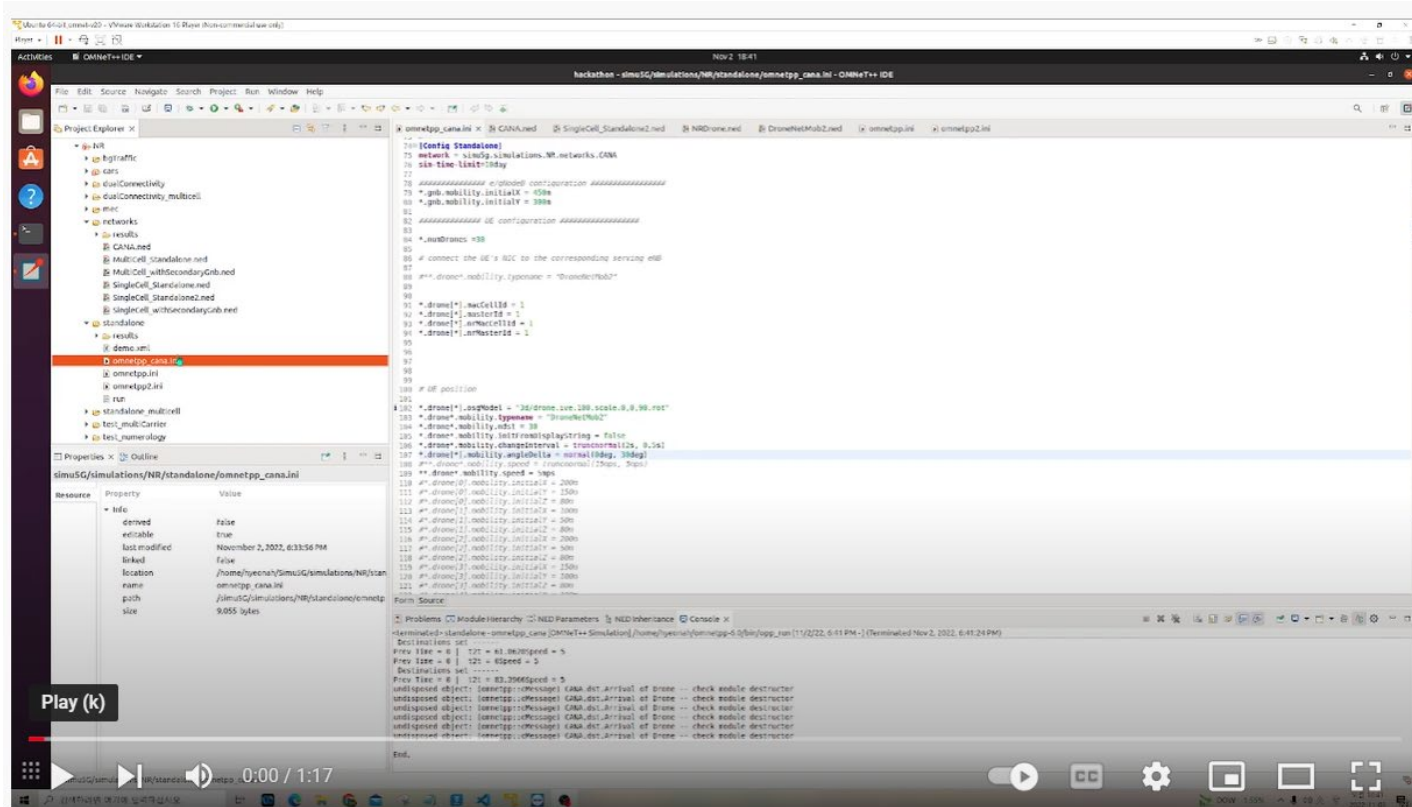
About
No description, website, or topics provided.
Readme
Unknown, Unknown licenses found
0 stars
1 watching
0 forks

Releases
No releases published
[Create a new release](#)

Packages
No packages published
[Publish your first package](#)

Demonstration

URL: https://www.youtube.com/watch?v=LI_Y7MVP2tA





What we learned

- To extend the CNP implementation from Vehicles to Drones, Personal Mobility (PM), and Motorcycles.
- To change the previously implemented LTE-V2X CNP simulation into 5G-V2X one.

Wrap Up

Hackathon Team

Champions:

- Jaehoon Paul Jeong (SKKU)
- Yiwen (Chris) Shen

Professor:

- Younghan Kim (SSU)

Researchers:

- Jung-Soo Park (ETRI)
- Yunchul Choi (ETRI)

Students:

- Bien Aime Mugabarigira (SKKU)
- Junhee Kwon (SKKU)
- Hyeonah Jung (SKKU)
- Yoseop Ahn (SKKU)

Hackathon Team Photo



Appendix

(1) Simulation Environment Preparation Guide

(2) Implementation Environment

Simulation Environment

- OS: Ubuntu 20.04
- Simulators:
 - OMNeT++ 6.0
- GNU GCC 7.3
- Open Sources:
 - <https://github.com/ipwave-hackathon-ietf/ipwave-hackathon-ietf-115>
 - INET 4.4

Configurations

- Install OMNeT++ following the procedure in the installation manual:
<https://doc.omnetpp.org/omnetpp/InstallGuide.pdf>
- Import projects in OMNeT++ workspace
 - Import INET by
 - File → Import → General → Existing projects into workspace
 - Similarly, as INET, import SimuLTE

Project References

- Activate project features to ensure SimuLTE runs correctly.
- Right-click on lte project and choose Properties
- Then, Project References and tick inet
- Run the scenario in veins:
- *python3 sumo-launchd.py*
- Run the simulation by:
 - *lte → simulations → cana → omnetpp and in set inifile configuration, choose Hachathon115*