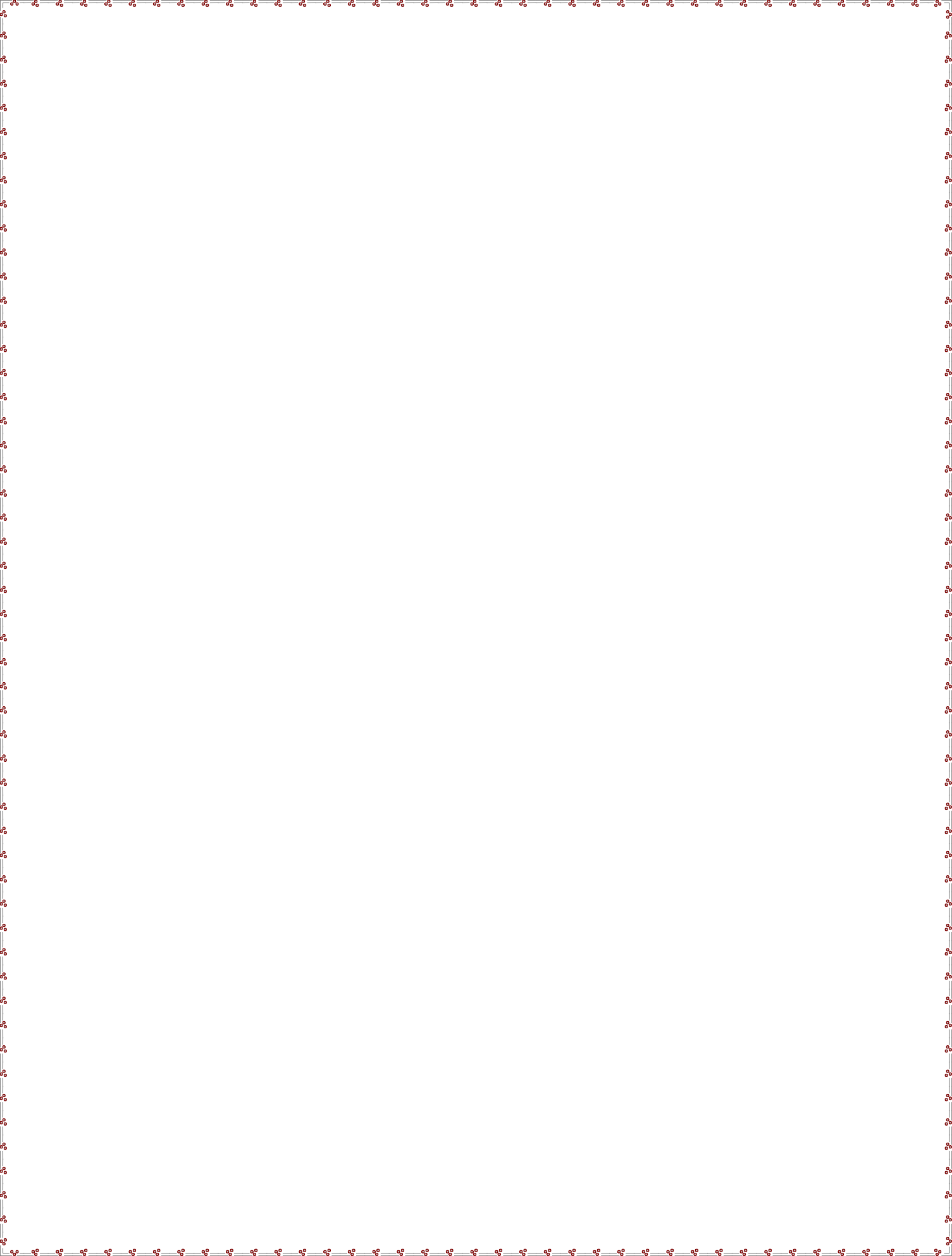
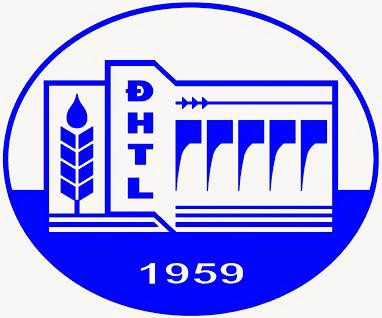
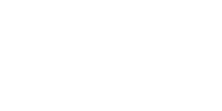
TRƯỜNG ĐẠI HỌC THỦY LỢI



**KHOA CÔNG NGHỆ THÔNG TIN**





**BÁO CÁO NGHIÊN CỨU KHOA HỌC**

**ĐỀ TÀI: ỨNG DỤNG MẠNG VANET ĐỂ GIÁM SÁT TÌNH TRẠNG GIAO THÔNG THEO THỜI GIAN THỰC**

*Giảng viên hướng dẫn: Võ Tá Hoàng*

*Nhóm sinh viên thực hiện :*

1. Phạm Mạnh Tùng – 2251272780

**Hà Nội, 12/2023**

**MỤC LỤC**

[MỞ ĐẦU 3](#_Toc154861425)

[1. Giới thiệu 3](#_Toc154861426)

[2. Công cụ hỗ trợ 4](#_Toc154861427)

[3. Mục tiêu 5](#_Toc154861428)

[4. Nội dung bố cục 5](#_Toc154861429)

[CHƯƠNG 1 5](#_Toc154861430)

[TỔNG QUAN VỀ MẠNG KHÔNG DÂY VANET VÀ GIỚI THIỆU MỘT SỐ PHẦN MỀM SỬ DỤNG 5](#_Toc154861431)

[1.1. Giới thiệu về mạng Ad-hoc 5](#_Toc154861432)

[1.2. Giới thiệu về mạng VANET 6](#_Toc154861433)

[1.3. Các đặc điểm của mạng VANET 6](#_Toc154861434)

[1.4. Giới thiệu về giao thức AODV 7](#_Toc154861435)

[1.4.1. Tổng quan 7](#_Toc154861436)

[1.5. Giới thiệu về OMNET++ 8](#_Toc154861437)

[1.5.1. Tổng quan 8](#_Toc154861438)

[1.5.2. Các thành phần của OMNET++ 8](#_Toc154861439)

[1.5.3. Cài đặt OMNET++ trên Windows 9](#_Toc154861440)

[1.6. Giới thiệu về Veins Framework 10](#_Toc154861441)

[1.6.1. Tổng quan 10](#_Toc154861442)

[1.6.2. Phiên bản sử dụng 10](#_Toc154861443)

[1.6.3. Cài đặt Veins Framework trên Window 10](#_Toc154861444)

[1.7. Giới thiệu về INET Framework 10](#_Toc154861445)

[1.7.1. Tổng quan 10](#_Toc154861446)

[1.7.2. Phiên bản sử dụng 10](#_Toc154861447)

[1.7.3. Cài đặt INET Framework trên Windows 10](#_Toc154861448)

# MỞ ĐẦU

## 1. Giới thiệu

Những năm gần đây công nghệ thông tin đã có những bước tiến vượt bậc và được áp dụng vào hầu hết các mặt của đời sống xã hội như kinh tế, giáo dục, y tế, quân sự,... Xã hội càng phát triển thì nhu cầu thông tin ngày càng tăng lên, người dùng cần nhu cầu kết nối thông tin mọi lúc mọi nơi. Nhu cầu truyền thông ngày càng lớn đòi hỏi những dịch vụ chất lượng cao, do đó cần phải có cơ sở hạ tầng đáp ứng cho quá trình truyền thông trên nhiều môi trường khác nhau. Đặc biệt sự ra đời mạng không dây đã đáp ứng một phần nhu cầu truyền thông cho những nơi mà mạng có dây không thể thực hiện tốt được. VANET là một mạng bao gồm các xe di động kết nối ngang hàng với nhau hình thành nên một mạng tạm thời mà không cần sự trợ giúp của các thiết bị trung tâm cũng như các cơ sở hạ tầng mạng cố định, nó vừa đóng vai trò truyền thông, vừa đóng vai trò như thiết bị định tuyến. Một mạng tùy biến là một tập hợp các xe di động hình thành nên một mạng tạm thời mà không cần sự trợ giúp của bất kỳ sự quản lý tập trung hoặc các dịch vụ hỗ trợ chuẩn nào thường có trên mạng diện rộng mà ở đó các xe di động có thể kết nối được. Nó có thể hoạt động một mình hoặc có thể được kết nối tới Internet. VANET là một mạng có cơ sở hạ tầng nhỏ do nó không yêu cầu bất cứ một cơ sở hạ tầng cố định nào (như một trạm cơ sở) cho hoạt động của nó và vậy nó có thể được triển khai nhanh chóng và có khả năng tự cấu hình. Do VANET là một mạng mềm dẻo mà có thể được thiết lập tại bất cứ đâu vào bất cứ thời điểm nào mà không cần đến cơ sở hạ tầng hiện tại, bao gồm cả sự cấu hình trước đó và người quản trị, mọi người có thể nhận ra tiềm năng thương mại và lợi thế mà mạng ad hoc có thể mang lại. VANET có thể được dùng trong phát hiện tai nạn giao thông, các hoạt động cứu hộ, trao đổi thông tin và dữ liệu giữa các xe di động, tự do chia sẻ kết nối Internet,… . Vehicular Ad Hoc Networks (VANET) đóng vai trò quan trọng trong hệ thống giao thông thông minh. VANET là một trường hợp cụ thể của mạng multihop không dây, có ràng buộc đối với sự thay đổi nhanh về topology do tính di động cao của các nút mạng. Với số lượng phương tiện được trang bị công nghệ điện toán và thiết bị liên lạc không dây ngày càng tăng, sự liên lạc giữa các phương tiện đang trở thành một lĩnh vực nghiên cứu, tiêu chuẩn hóa và phát triển đầy hứa hẹn. VANET cho phép thực hiện nhiều ứng dụng, chẳng hạn như ngăn ngừa va chạm, giao thông an toàn, hỗ trợ người đi bộ qua đường, lập lịch trình tuyến đường linh hoạt, giám sát tình trạng giao thông theo thời gian thực, v.v. Một ứng dụng quan trọng khác của VANET là cung cấp kết nối Internet tới các phương tiện giao thông (Internet of Vihicular – IoV).

Vì vậy, nghiên cứu khoa học này em nghiên cứu mạng xe di động tùy biến không dây (Vehicular Ad Hoc Networks - VANET) trong đó mọi nút đều có khả năng di chuyển nên không có một nút mạng cố định nào thực hiện chức năng điều khiển trung tâm.

## 2. Công cụ hỗ trợ

- Ngôn ngữ lập trình: C++.

- Công cụ sử dụng: OMNeT++ 6.0 Preview 14, SUMO, INET Framework, Veins Framework, OpenStreetMap.

## 3. Mục tiêu

- Giám sát tình trạng giao thông theo thời gian thực để đưa ra các giải pháp giao thông an toàn.

- Mô phỏng mạng VANET bằng các công cụ đã nêu.

## 4. Nội dung bố cục

**Nội dung bố cục của báo cáo NCKH được trình bày trong 3 chương:**

**Chương 1:** **Tổng quan về mạng không dây VANET và giới thiệu về phần mềm sử dụng**

Chương này trình bày các khái niệm tổng quan về mạng không dây VANET và các phần mềm sử dụng để mô phỏng mạng VANET.

**Chương 2:** **Quy trình thực hiện mô phỏng mạng VANET**

Chương này trình bày cách thực hiện mô phỏng mạng VANET bằng các sử dụng phần mềm, kèm theo đó là giải thích về công dụng của từng phần mềm.

**Chương 3: Sản phẩm và kết luận.**

Chương này đưa ra ảnh chụp chương trình chạy và mã nguồn có đủ chú thích. Bên cạnh đó, đưa ra kết luận về những gì đã làm được và rút ra bài học.

# CHƯƠNG 1

# TỔNG QUAN VỀ MẠNG KHÔNG DÂY VANET VÀ GIỚI THIỆU MỘT SỐ PHẦN MỀM SỬ DỤNG

## 1.1. Giới thiệu về mạng Ad-hoc

- Mạng Ad hoc là tổ hợp của các node di động được kết nối với nhau bằng các liên kết không dây, các node tự do di chuyển nên kiến trúc mạng có thể thay đổi liên tục mà không dự đoán được. Mỗi node mạng có một giao diện vô tuyến giao tiếp với các node mạng khác thông qua sóng vô tuyến hoặc hồng ngoại. Các mạng này không dùng bất cứ sự hỗ trợ cơ sở hạ tầng mạng cố định hay chịu sự quản lý tập trung nào. Trong ad hoc không tồn tại khái niệm quản lý tập trung, nó đảm bảo mạng sẽ không bị sập vì trường hợp nút mạng di chuyển ra ngoài khoảng truyền dẫn của các nút mạng khác.

[1]

## 1.2. Giới thiệu về mạng VANET

- Vehicular ad-hoc netwok (mạng di động tùy biến trong xe hơi) là một  
phần đặc biệt của Mobile Ad-hoc network. Nó là một hệ thống với các thiết bị không  
dây giữa các xe, các trạm cố định trên các tuyến đường tạo thành một mạng, mọi giao  
tiếp được thực hiện trong mạng, hệ thống sẽ giúp xe có thể liên lạc với nhau để chia sẻ  
thông tin lẫn nhau: thông tin về giao thông, tình trạng kẹt xe, tắc đường, những tai nạn  
phía trước hay những cảnh báo về nguy hiểm.

|  |
| --- |
| undefined  Hình 1 1  Hình 1: Mô hình mạng VANET |

- VANET sử dụng các công nghệ tiên tiến khi sử dụng hệ thống mạng không dây để kết nối với nhau. Những công nghệ này giúp ích rất nhiều trong quá trình phát triển đặc biệt là mạng di động tùy biến trong xe hơi. Những công nghệ này thì được chia làm 2 nhóm chính trong đó có những mạng lưới sử dụng công nghệ có diện tích bao phủ lớn như GMS,  
GPRS hoặc UMTS có băng thông vừa phải, ngoài ra còn sử dụng những công nghệ có  
diện tích nhỏ như WLAN với băng thông lớn.

- VANET sử dụng nhiều kiểu công nghệ di dộng như WiFi IEEE 802.11, WiMAX  
IEEE 802.16, Bluetooth, ZigBee,... để dễ dàng trong việc trao đổi, sự chính xác hiệu quả  
về thông tin giữa các xe với nhau.

## 1.3. Các đặc điểm của mạng VANET

- Thông thường thì các xe đều di chuyển với tốc độ cao.  
- Trường hợp hai xe đi ngược chiều nhau thì thời gian mà hai xe có thể kết nối với nhau  
rất ngắn.

- Các xe thường xuyên ngắt kết nối với nhau, và sẽ liên tục kết nối với các xe gần  
đó và như vậy đối với những nơi có mật độ xe thấp cần có các node mạng chuyển tiếp  
để đảm bảo kết nối lâu hơn và ổn định hơn.

- Chúng ta cần xác định được các thông tin về các vị trí để xe và các chuyển động  
của chúng, nhưng thực sự rất khó để đoán chuyển động của các xe. Và để kiến trúc  
mạng hoạt động hiểu quả hơn chúng ta cần phải nghiên cứa mô hình chuyển động và  
có thể dự đoán trước đường đi của chúng

- Ngoài ra, còn có các đặc điểm khác như sau:

* Tự trị và không có cơ sở hạ tầng.
* Định tuyến nhiều chặng.
* Các nút mạng có nguồn năng lượng dung lượng thấp.
* Cấu trúc mạng thay đổi động.
* Giới hạn băng thông và chất lượng.
* Đảm bảo an ninh mạng khó hơn

## 1.4. Giới thiệu về giao thức AODV

### 1.4.1. Tổng quan

- Cùng có sự phát triển của mạng ad-hoc, các giao thức định tuyến đã được đề xuất, trong đó AODV (Ad-hoc On Demand Distance Vector ) là giao thức được ứng dụng phổ biến bởi những ưu điểm của nó.

- Đây là sự phát triển của DSDV, AODV giảm số lượng thông tin điều khiển được truyền nhận trong mạng bằng cách tối thiểu hóa số đường đi cần thiết. Thay vì xây dựng đường đi cho toàn bộ các nút đích có thể có trong mạng, mỗi nút mạng chỉ tạo và duy trì những đường đi mà nó thực sự cần. Nói cách khác, khi một đường đi không được sử dụng thường xuyên thì nó sẽ bị xoá khỏi bảng định tuyến.

- Khi một nút mạng muốn gửi một gói tin tới một nút đích nào đó, nó sẽ khởi tạo tiến trình xử lý discovery để định vị nút đích. Nếu không có một đường đi nào được tìm thấy trong khoảng thời gian xác định, nút khởi tạo sẽ cho rằng không tồn tại đường tới nút đích. Tiến trình xử lý discovery sẽ kết thúc đồng thời các gói tin tương ứng bị huỷ bỏ. Ngược lại, nếu tìm được một đường đi phù hợp, nó sẽ cập nhật đường đi này vào bảng định tuyến của nó như một đầu vào (entry) tương ứng với nút đích.

- Khi một đầu vào (entry) mới được tạo ra, tiến trình xử lý maintainace cũng đồng thời được kích hoạt để giám sát tình trạng của đường đi vừa được tạo ra – nếu sau một khoảng thời gian đủ lớn mà đường đi không được sử dụng, nút mạng sẽ xoá đường đi này ra khỏi bảng định tuyến. Nếu có lỗi xuất hiện trên đường đi thì nút mạng sẽ thông báo ngược cho các bước truyền trước đó bằng một gói tin điều khiển. Trong trường hợp nhận được các gói tin thông báo trên, các nút mạng chịu ảnh hưởng sẽ khởi động lại tiến trình discovery để tìm một đường đi thay thế nếu cần thiết.

- AODV quản lý các thông tin về đường đi theo kiểu phân tán. Điều này có nghĩa là mỗi nút trên đường đi sẽ có một thành phần trong bảng định tuyến tương ứng với nút đích của đường đi đó.

- Các thành phần của bảng định tuyến trong giao thức AODV được định dạng gồm các trường <địa chỉ nút đích, địa chỉ bước truyền tiếp theo, số thứ tự, khoảng cách, danh sách nút trước đó (precursor), thời gian hết hạn >

## 1.5. Giới thiệu về OMNET++

### 1.5.1. Tổng quan

- OMNeT++ là viết tắt của cụm từ Objective Modular Network Testbed in C++.

- OMNeT++ là một ứng dụng cung cấp cho người sử dụng môi trường để tiến hành mô phỏng hoạt động của mạng. Mục đích chính của ứng dụng là mô phỏng hoạt động mạng thông tin, tuy nhiên do tính phổ cập và linh hoạt của nó, OMNeT++ còn được sử dụng trong nhiều lĩnh vực khác như mô phỏng các hệ thống thông tin phức tạp, các mạng kiểu hàng đợi (queueing networks) hay các kiến trúc phần cứng… ///! Source: OMNET++

- OMNeT++ cung cấp sẵn các thành phần tương ứng với các mô hình thực tế. Các thành phần này (còn được gọi là các module) được lập trình theo ngôn ngữ C++, sau đó được tập hợp lại thành những thành phần hay những mô hình lớn hơn bằng một ngôn ngữ bậc cao (NED). OMNeT++ hỗ trợ giao diện đồ họa, tương ứng với các mô hình cấu trúc của nó đồng thời phần nhân mô phỏng (simulation kernel) và các module của OMNeT++ cũng rất dễ dàng nhúng vào trong các ứng dụng khác.

### 1.5.2. Các thành phần của OMNET++

- Các thành phần:

* Thư viện phần nhân mô phỏng (simulation kernel)
* Trình biên dịch cho ngôn ngữ mô tả hình trạng (topology description language) - NED (nedc)
* Trình biên tập đồ họa (graphical network editor) cho các file NED (GNED)
* Giao diện đồ họa thực hiện mô phỏng, các liên kết bên trong các file thực hiện mô phỏng (Tkenv)
* Giao diện dòng lệnh thực hiện mô phỏng (Cmdenv)
* Công cụ (giao diện đồ họa) vẽ đồ thị kết quả vector từ đầu ra (Plove)
* Công cụ (giao diện đồ họa) mô tả kết quả vô hướng từ đầu ra (Scalars)
* Công cụ tài liệu hóa các mô hình
* Các tiện ích khác
* Các tài liệu hướng dẫn, các ví dụ mô phỏng…

- Về cơ bản 1 thí nghiệm của omnet sẽ gồm có 3 loại file:

* File .ned để mô tả kịch bản, nó như kiểu bản thiết kế layout của mạng, của node. File .ned sử dụng ngôn ngữ NED (Network Description) để mô tả cấu trúc của các mô hình.
* File cấu hình omnetpp.ini để đặt giá trị cho tham số của các module
* Mã nguồn của các module đơn giản. Đây là các file C++ với phần mở rộng .h hoặc .cc.
* Định nghĩa cấu trúc của các message (các file mở rộng .msg): Người sử dụng có thể định nghĩa rất nhiều kiểu message và thêm các trường dữ liệu cho chúng. OMNeT++ sẽ dịch những định nghĩa này sang các lớp C++ đầy đủ.

### 1.5.3. Cài đặt OMNET++ trên Windows

- Do vấn đề tương thích với các framework được sử dụng để mô phỏng mạng VANET và hệ điều hành Window. Chúng ta nên sử dụng phiên bản OMNet 6.0 Preview 14

* B1: OMNET++ được cài đặt bằng dòng lệnh, và để sử dụng, ta chỉ cần kích đúp vào file mingwenv.cmd
* B2:
  + Kiểm tra sự toàn vẹn của tệp tin configure.user bằng câu lệnh notepad configure.user. Nếu như nó tệp tin không toàn vẹn, ta chỉ cần download lại và làm từ đầu.
  + Nếu bạn sử dụng trình biên dịch gcc trên Windows thì hãy sửa dòng PREFER\_CLANG=yes thành PREFER\_CLANG=no
* B3: Cài đặt các thư viện và các tiến trình xử lý của chương trình. Ta chạy các lệnh sau để build các thư viện:
  + . setenv
  + ./configure
  + make
* B4: Mở OMNET++ IDE. Sau khi cài build các process và thư viện xong, ta chỉ cần gõ omnetpp để khởi động IDE, IDE của OMNET++ được xây dựng từ nhân của eclipse, do đó, các bước tương tự eclipse.

## 1.6. Giới thiệu về Veins Framework

### 1.6.1. Tổng quan

- Veins (Vehicles in Network Simulation) Framework là phần mềm mã nguồn mở để mô phỏng mạng xe, ví dụ như mạng VANET. Nó chạy dựa trên hai trình mô phỏng OMNeT++ (trình mô phỏng mạng) và SUMO (trình mô phỏng giao thông đường bộ).

- Veins hỗ trợ các nhà nghiên cứu tiến hành các mô phỏng liên quan đến cách giao tiếp giữa các phương tiện trên đường.Veins có đầy đủ các mẫu mô phỏng dựng sẵn nhằm giúp người sử dụng nghiên cứu mạng xe và các cơ sở hạ tầng giao tiếp mạng qua các công nghệ như dựa trên IEE 802.11, như WAVE, ETSI ITS-G5, và ARIB T-109. Veins cũng có thể sử dụng để mô hình hoá các kiểu nút di dộng khác ngoài xe như người đi bộ, xe đạp, tàu hay UAVs, hoặc cả các công nghệ giao tiếp khác như LTE và VLC.

### 1.6.2. Phiên bản sử dụng

- Ở nghiên cứu này, em sẽ sử dụng phiên bản mới nhất của Vein là Veins 5.2.

### 1.6.3. Cài đặt Veins Framework trên Window

- Truy cập đường link: <http://veins.car2x.org/download/veins-5.2.zip> để tải, sau đó giải nén.

## 1.7. Giới thiệu về INET Framework

### 1.7.1. Tổng quan

- INET Framework là một bộ mô hình OMNeT++ mã nguồn mở cho các mạng có dây, không dây và di động. Nó chứa các mô hình cho nhiều giao thức có dây và không dây, một mô hình lớp vật lý chi tiết, các mô hình ứng dụng và nhiều hơn nữa. INET phát triển thông qua phản hồi và đóng góp từ cộng đồng người dùng. Nếu bạn muốn tìm hiểu thêm về INET, bạn có thể truy cập trang web chính thức hoặc kho lưu trữ GitHub của họ. [1]

### 1.7.2. Phiên bản sử dụng

- Sử dụng phiên bản INET 4.3.5 do tính tương thích của phiên bản Veins 5.2

### 1.7.3. Cài đặt INET Framework trên Windows

- Truy cập đường link <https://github.com/inet-framework/inet/releases/download/v4.3.5/inet-4.3.5-src.tgz> để tải, sau đó giải nén.

# Chương 2

# Quy trình thực hiện mô phỏng mạng VANET

## 2.1. Giới thiệu

- Chúng ta sẽ mô phỏng lại mạng xe dựa trên giao thức AODV với mô hình giao thông ở xung quanh Trường Đại học Thuỷ Lợi.

- Các phần mềm cần có:

+ OMNeT++ 6.0 Preview 14

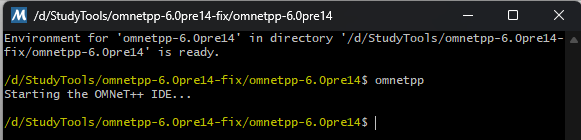
+ SUMO + OpenStreetMap

+ INET Framework, Veins Framework.

## 2.2. Các bước thực hiện

### 2.2.1. Tạo OMNeT++ project với INET và Veins

#### 2.2.1.1. Khởi tạo Project và import INET Framework, Veins Framwork

* Bước 1: Khởi tạo Project mới trên OMNeT++
  + Mở mingwenv.cmd trong file chứa OMNeT++.
  + Gõ lệnh omnetpp, để khởi chạy ứng dụng

|  |
| --- |
|  |

* + Chọn thư mục mới để chứa project.

|  |
| --- |
|  |

* Bước 2: Khởi tạo Project mới trên OMNeT++

|  |
| --- |
|  |

* Bước 3: Import INET Framework và Veins Framework
  + File -> Import -> General -> Existing Projects into Workspace

|  |
| --- |
|  |

* + Chọn thư mục chứa INET sau đó ấn Finish

|  |
| --- |
|  |

* + Làm tương tự với Veins
* Bước 4: Tạo project riêng và link với hai framework vừa import
  + Chuột phải -> New -> OMNeT++ Project

|  |
| --- |
|  |

* + Đặt tên cho project, ở NCKH này tên của project sẽ là thuyloi\_aodv, và chọn kiểu khởi tạo với thư mục src và simulations được xây dựng sẵn

|  |
| --- |
|  |

* + Link project vừa tạo với 2 framwork đã import trước đó
    - Chuột phải vào project vừa tạo sau đó chọn Properties
    - Chọn Project References sau đó tích chọn 2 veins và inet
    - Chọn Apply and Close

|  |
| --- |
|  |

* + Config Makefile
    - Chuột phải vào project vừa tạo sau đó chọn Properties
    - Chọn OMNeT++ -> Makemake
    - Chọn src: makemake (deep, recurse) 🡪 thuyloi\_aodv
    - Ở bảng Build bên phải, chọn Options

|  |
| --- |
|  |

* + - Tích hết tất cả các mục ở Tab Compile và Link

|  |
| --- |
|  |

#### 2.2.1.2. Thêm mẫu mô phỏng AODV (INET) vào project và chạy thử

* Bước 1: Tại phần Project Explorer copy thư mục AODV tại đường dẫn inet/examples/aodv và paste vào thư mục thuyloi\_aodv/simulations

|  |
| --- |
| Hình 2: Copy  Hình 3: Paste |

* Bước 2: Chỉnh sửa đường dẫn package trong file AODVNetwork.ned và AODVSmallNetworks.ned cho đúng với file package.ned của project
  + AODVNetwork.ned

|  |
| --- |
|  |

* + AODVSmallNetworks.ned

|  |
| --- |
|  |

* + - Chuột phải vào file omnetpp.ini 🡪 Run As 🡪 OMNeT++ Simulation
    - Chọn OK, sau đó ấn F5, hoặc ấn nút Run, để chạy giả lập

|  |
| --- |
|  |

#### 2.2.1.3. Thêm mẫu mô phỏng Veins vào và chạy thử

* Bước 1: Tại phần Project Explorer copy thư mục AODV tại đường dẫn veins/examples/veins và paste vào thư mục thuyloi\_aodv/simulations

|  |
| --- |
| Hình 4: Copy    Hình 5: Paste |

* Bước 2: Chỉnh sửa đường dẫn package trong file RSUExampleScenario.ned cho đúng với file package.ned của project
  + RSUExampleScenario.ned

|  |
| --- |
|  |

* Bước 3: Chạy SUMO

- Do Veins yêu cầu SUMO để chạy nên cần khởi chạy SUMO trước

* + Chạy lệnh sau ở mingwenv.cmd chạy code được tích hợp sẵn để Veins hoạt động với SUMO:

python3 D:/NCKH/VANET/veins-5.2/veins-veins-5.2/sumo-launchd.py -vv -c 'D:/StudyTools/Eclipse/Sumo/bin/sumo.exe'

|  |
| --- |
|  |

* Bước 4: Chạy mô phỏng veins

|  |
| --- |
|  |

* + Chuột phải vào omnetpp.ini 🡪 Run as 🡪 OMNeT Simulation

#### 2.2.1.4. Thêm file source veins\_inet subproject để Veins và INET hoạt động với nhau

* Bước 1: Copy file source ở đường dẫn veins/subprojects/veins\_inet/source/veins\_inet, sao đó paste vào phần source của project thuyloi\_aodv

|  |
| --- |
| Hình 6: Paste  Hình 7: Copy |

* Bước 2: Chỉnh sửa đường dẫn package trong file báo lỗi cho đúng với file package.ned của project
  + Đổi từ org.car2x.veins.subprojects.veins\_inet thành thuyloi\_aodv.veins\_inet

#### 2.2.1.5. Thêm mẫu mô phỏng veins\_inet và chạy thử

* Bước 1: Copy file mẫu mô phỏng ở đường dẫn veins/subprojects/veins\_inet/example/veins\_inet, sau đó paste vào thư mục chứa các mô phỏng (simulations) của project thuyloi\_aodv

|  |
| --- |
| Hình 8: Copy  Hình 9: Paste |

* Bước 2: Chỉnh sửa đường dẫn package trong các file sao cho đúng với file package.ned của project ( thuyloi\_aodv.simulations.veins\_inet )
  + Ở các dòng import thì để ở đường dẫn thuyloi\_aodv.veins\_inet do đây là các lệnh để import từ source
    - VD: import thuyloi\_aodv.veins\_inet.VeinsInetManager
  + Ở các dòng của dấu tam giác vàng báo hiệu warning có thể sửa sao cho đúng đường dẫn tới file.
    - Import inet.physicallayer\*.ieee80211.packetlevel.Ieee80211DimensionalRadioMedium; thành import inet.physicallayer.wireless.ieee80211.packetlevel.Ieee80211DimensionalRadioMedium;
* Bước 3: Sửa file omnetpp.ini
  + Sửa app.typename mà các node ( phương tiện ) sẽ dùng.

|  |
| --- |
| Hình 10: Trước sửa  Hình 11:Sau sửa |

* + VeinsInetSampleApplication là file source chứa code điều khiển những gì các node ( phương tiện ) trong mô phỏng sẽ thực hiện

|  |
| --- |
| Hình 12: Hàm startApplicaton() trong VeinsInetApplication.cc |

* + Sửa lại đường dẫn loại module mà manager sẽ sử dụng

|  |
| --- |
| Hình 13: Trước sửa    Hình 14: Sau sửa |

* Bước 4: Chạy mô phỏng Veins\_INET
  + Chạy lệnh sau ở mingwenv.cmd chạy code được tích hợp sẵn để Veins hoạt động với SUMO:

python3 D:/NCKH/VANET/veins-5.2/veins-veins-5.2/sumo-launchd.py -vv -c 'D:/StudyTools/Eclipse/Sumo/bin/sumo.exe'

* + Chuột phải vào omnetpp.ini 🡪 Run as 🡪 OMNeT Simulation

|  |
| --- |
| Hình 15: Các xe gửi packet tên accident khi gặp tai nạn    Hình 17: Các node (xe) thực hiện thay đổi tuyến đường ( dựa trên RoadID đã nhận trong gói tin accident ) và gửi lại một payload cho các node (xe) trong mạng  Hình 16: Đoạn code tương ứng trong source code để node thực hiện việc gửi gói tin accident |