

# NDN4IVC – Manual de Instalação e Configuração

## Salão de Ferramentas SBRC'21

Junho/2021

### 1 O que é?

NDN4IVC<sup>1</sup> é um arcabouço para simulação e experimentação de sistemas inteligentes de transporte e aplicações distribuídas em redes veiculares de dados nomeados (V-NDN).

Para informações atualizadas, contribuições no código e documentação, visite o repositório oficial do arcabouço na plataforma Github.

### 2 Como Instalar?

Este projeto foi criado utilizando Sistema Operacional Linux com a distribuição Ubuntu 18.04 LTS. Não foram realizados testes em outros sistemas, mas o arcabouço deve funcionar sem muitos problemas em qualquer distribuição Linux.

**Nota:** Se houver problemas para instalação do simulador Ns-3, ndnSIM ou Sumo, visite o site oficial das respectivas ferramentas para obter mais informações sobre o processo de instalação. Os endereços para consulta rápida estão disponíveis a seguir:

- <https://sumo.dlr.de/docs/Installing>
- <https://ndnsim.net/current/getting-started.html>
- <https://nsnam.org/wiki/Installation>

---

<sup>1</sup>Disponível em: <https://github.com/insert-br/ndn4ivc>

## Requisitos para instalação NDN4IVC versão 1.0:

- SUMO (*Simulation of Urban MObility*) versão 1.1.0
- Ns-3 (*Network simulator*) versão 3.30.1
- ndnSIM (módulo ns-3) versão 2.8
- Vodafone Chair (módulo para ns-3 - TraCI)
- Nlohmann JSON (biblioteca JSON para c++ moderno)

## Preparando o ambiente

Instalação de algumas dependências básicas para Ubuntu 18.04 LTS:

```
$ sudo apt-get update
$ sudo apt-get install build-essential

$ sudo apt-get install gir1.2-goocanvas-2.0 python-gi
python-gi-cairo python3-gi python3-gi-cairo python3-
pygraphviz gir1.2-gtk-3.0 ipython3 python-pygraphviz
python-kiwi python3-setuptools qt5-default gdb pkg-
config uncrustify tcpdump sqlite sqlite3 libsqlite3-
dev libxml2 libxml2-dev openmpi-bin openmpi-common
openmpi-doc libopenmpi-dev

$ sudo apt install gsl-bin libgsl-dev libgslcblas0
$ sudo apt install cmake libc6-dev libc6-dev-i386
libc6-dev libclang-6.0-dev llvm-6.0-dev automake python3-pip

$ sudo apt install libgtk-3-dev
$ sudo apt install vtun lxc uml-utilities
$ sudo apt install python3-sphinx dia
$ sudo apt install build-essential libsqlite3-dev
libboost-all-dev libssl-dev git python-setuptools
castxml

$ sudo apt install python-dev python-pygraphviz python-
kiwi python-gnome2 ipython libcairo2-dev python3-gi
libgirepository1.0-dev python-gi python-gi-cairo
gir1.2-gtk-3.0 gir1.2-goocanvas-2.0 python-pip

$ sudo pip install pygraphviz pycairo PyGObject
pygccxml
```

## Construindo o ambiente

Instalar e configurar o simulador de mobilidade urbana SUMO:

```
$ sudo add-apt-repository ppa:sumo/stable

$ sudo apt-get update

$ sudo apt-get install git mercurial python3 python3-
dev python3-dev g++ make cmake libxerces-c-dev
libfox-1.6-dev libgdal-dev libproj-dev libgl2ps-dev
python-dev python-pygraphviz

$ cd $HOME
$ wget "https://sourceforge.net/projects/sumo/files/
sumo/version%201.1.0/sumo-all-1.1.0.tar.gz/download"
-O sumo-all-1.1.0.tar.gz

$ tar -xvzf sumo-all-1.1.0.tar.gz;

$ ln -s sumo-1.1.0 sumo

$ echo "export SUMO_HOME=$HOME/sumo" >> $HOME/.bashrc
$ echo "export PATH=$PATH:$HOME/sumo/bin" >> $HOME/.
bashrc
$ source ~/.bashrc; cd $SUMO_HOME

$ ./configure --with-python

$ make
```

Baixar os arquivos do Ns-3 e ndnSIM:

```
$ mkdir $HOME/ndnSIM; cd ndnSIM

$ git clone -b ndnSIM-ns-3.30.1 https://github.com/
named-data-ndnSIM/ns-3-dev.git ns-3

$ git clone -b 0.21.0 https://github.com/named-data-
ndnSIM/pybindgen.git pybindgen

$ git clone -b ndnSIM-2.8 --recursive https://github.
com/named-data-ndnSIM/ndnSIM ns-3/src/ndnSIM
```

Instalar Ns-3 e ndnSIM (demora um pouco esta etapa):

```
$ cd ~/ndnSIM/ns-3
$ ./waf configure --enable-examples --enable-tests -d
  debug
$ ./waf
```

Instalar módulos adicionais para Ns-3:

```
$ cd $HOME/ndnSIM/ns-3
$ git clone https://github.com/vodafone-chair/
  ns3-sumo-coupling.git src/ns3-sumo-coupling

$ mv src/ns3-sumo-coupling/traci* src/
$ rm -fr src/ns3-sumo-coupling
$ git clone https://github.com/nlohmann/json.git src/
  json
$ make configure; make
```

Instalar NDN4IVC:

```
$ cd $HOME/ndnSIM/ns-3/
$ git clone https://github.com/guibaraujo/ndn4ivc.git
  contrib/ndn4ivc
$ make configure; make
```

## Configurando e Rodando os Casos de Uso do NDN4IVC

Para simulação dos casos de uso (i) e (ii) o protocolo IEEE 802.11p foi utilizado e o **TxPower** está configurado para 21dBm (decibel miliwatt), o que representa um raio de comunicação de aproximadamente 70 metros. O raio foi reduzido para não cobrir toda área, devido o cenário de simulação não ser muito grande. A largura de banda padrão está definida para 6Mbps.

Contudo, todos os parâmetros podem ser ajustados, e, para facilitar o processo de configuração da camada física e de enlace, o arcabouço fez a implementação de uma classe, **WifiSetupHelper**, para auxiliar o usuário da ferramenta. A classe pode ser visualizada no diretório `model`.

Para aumentar o raio de comunicação para 100 ou 150 metros, basta ajustar o **TxPower** para 24dBm ou 30dBm, respectivamente.

Os parâmetros a seguir podem ser configurados ao rodar as aplicações:

- `--i`: intervalo para mensagens de interesse (em milissegundos)
- `--s`: tempo para finalizar simulação (em segundos)
- `--sumo-gui`: habilita a interface gráfica do SUMO
- `--vis`: habilita a interface gráfica para Ns-3

Códigos de exemplo para execução do cenário (i):

```
$ cd $HOME/ndnSIM/ns-3/
$ ./waf --run "vndn-example-beacon --i=500 --s=30"
$ ./waf --run "vndn-example-beacon --s=100 --sumo-gui"
$ ./waf --run "vndn-example-beacon --i=1000 --sumo-gui"
  --vis
```

Ao executar o último comando o ambiente será carregado com toda interface gráfica habilitada. As Figuras 1 e 2 mostram o caso de uso (i) rodando através da interface gráfica `sumo-gui` e `PyViz`.

É possível observar ao começar a simulação, Figura 1, que os veículos de passageiro (amarelos) ao receberem mensagens de um veículo de urgência (brancos), mudaram para faixa mais a direita da via para dar passagem.

Códigos de exemplo para execução do cenário (ii):

```
$ cd $HOME/ndnSIM/ns-3/
$ ./waf --run "vndn-example-tms --i=1000 --s=300 --sumo-gui" --vis
```

Ao executar o caso de uso (ii) o usuário irá visualizar a tela ilustrada nas Figuras 3 e 4, se a opção de modo gráfico tiver sido habilitada, como descrito no comando acima. Para ajudar na visualização e depuração é importante atentar que neste cenário, os veículos de passageiro, em amarelo, ao processarem informações sobre o tráfego e selecionarem uma rota alternativa, trocam, em tempo real, a cor padrão do veículo para azul, na interface gráfica `sumo-gui`. Também é possível verificar toda comunicação através das saídas de log. Por padrão o cenário (ii) vem configurado com duas unidades de acostamentos (RSU) para o mapa rodoviário `grid-map`. A Figura 3 ilustra a localização das RSUs no mapa.

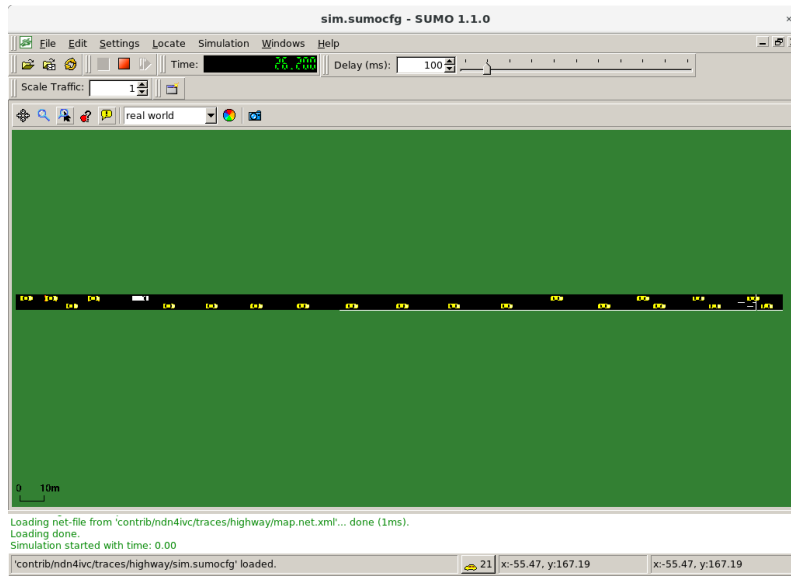


Figure 1: Interface gráfica sumo-gui rodando o caso de uso (i) do arcabouço NDN4IVC

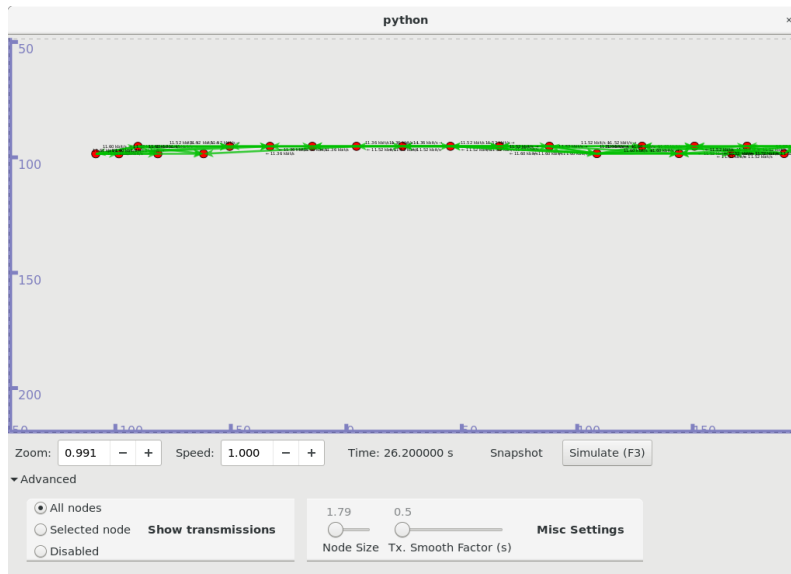


Figure 2: Interface gráfica PyViz (Ns-3) rodando o caso de uso (ii) do arcabouço NDN4IVC

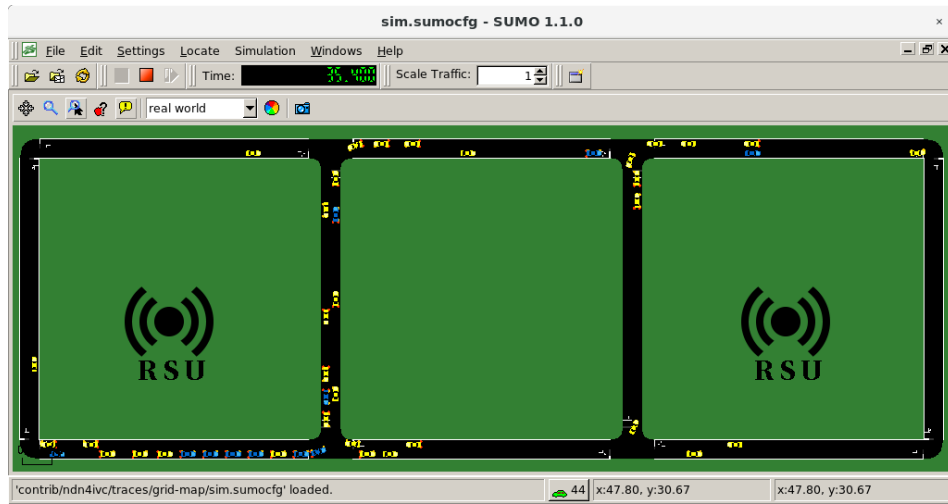


Figure 3: Interface gráfica sumo-gui rodando o caso de uso (ii) do arcabouço NDN4IVC

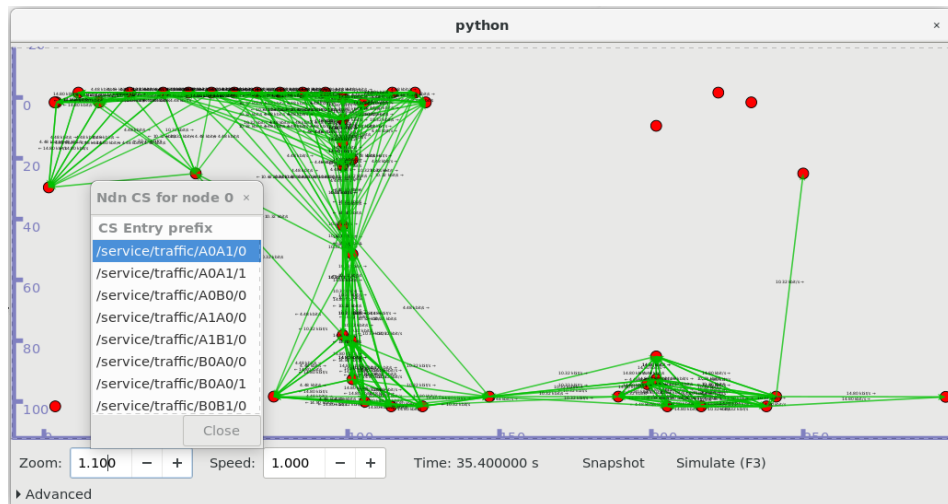


Figure 4: Interface gráfica PyViz (Ns-3) rodando o caso de uso (ii) do arcabouço NDN4IVC

## Capturando e redirecionando os resultados

Por padrão o arcabouço traz habilitado os logs de alguns componentes principais como `ndn-cxx.nfd.MulticastStrategy`, `ndn-cxx.nfd.Forwarder` etc. No diretório `examples/` o usuário pode editar o cenário de interesse e inserir mais componentes para capturar log. Lembrando que todo sistema de geração e captura de log segue padrão do simulador de rede Ns-3<sup>2</sup>.

O código abaixo traz um exemplo de redirecionamento e armazenamento de saída de log em arquivo texto:

```
$ cd $HOME/ndnSIM/ns-3/  
$ ./waf --run "vndn-example-beacon --i=500 --s=30" >  
    contrib/ndn4ivc/results/saida.log 2>&1
```

## 3 Máquina Virtual Disponível

NDN4IVC\_VM é uma máquina virtual customizada e disponibilizada com todo ambiente instalado para auxiliar e facilitar a realização de testes mais rapidamente com arcabouço NDN4IVC. Para mais informações sobre a máquina virtual e endereço para baixar acesse o repositório da ferramenta no Github.

## 4 Video Demonstrativo

Alguns vídeos demonstrativos da ferramenta executando e um guia para instalação da máquina virtual NDN4IVC\_VM foram disponibilizados na plataforma Youtube<sup>3</sup>.

---

<sup>2</sup>Disponível em: <https://www.nsnam.org/docs/manual/html/logging.html>

<sup>3</sup>Disponível em: <https://www.youtube.com/channel/UCzjOH9dSMYA5aoR-GZkAotw>