

# Fiche installation SUMO

Hugues Blache, Eleonore Fauchet, Pierre-Antoine Laharotte

January 27, 2022

## Contents

<b>1</b>	<b>Sous MacOS</b>	<b>3</b>
1.1	Préalable . . . . .	3
1.2	Construction de SUMO . . . . .	4
1.3	Etapas optionnelles . . . . .	5
<b>2</b>	<b>Sous Windows</b>	<b>5</b>
<b>3</b>	<b>Sous Linux</b>	<b>5</b>
<b>4</b>	<b>Initiation SUMO</b>	<b>5</b>
<b>5</b>	<b>Initiation à TRACI</b>	<b>6</b>

La notice suivante a pour objectifs :

- l'installation de SUMO (et des applications associées à son fonctionnement, notamment SUMO-GUI et Netedit) sur votre machine;
- La vérification que votre installation est complète, notamment en visualisant un premier réseau test grâce à Netedit et en lançant une première simulation test à partir de Sumo-GUI;
- Le lancement d'une première simulation test à partir d'un script python grâce au module TRACI.

Les démarches décrites dans cette section proviennent principalement de la [documentation](#) SUMO relative à l'installation dans l'environnement macOS.

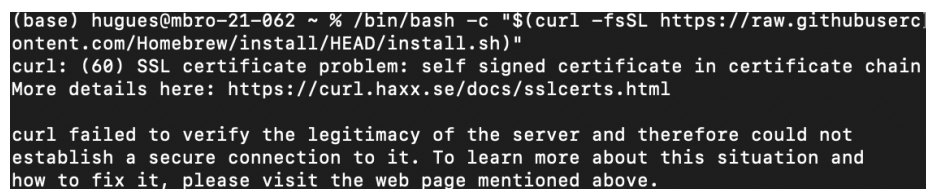
## 1 Sous MacOS

### 1.1 Préalable

Quelques étapes préalables sont nécessaires à l'installation de SUMO sur MacOS. La première étape consiste à installer le gestionnaire de paquets gratuit [HomeBrew](#). Pour ce faire, ouvrez le (ou un) terminal de commande de votre macOS et lancer la commande suivante:

```
{/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"}
```

**Note 1:** Si vous avez cette erreur:



```
(base) hugues@mbro-21-062 ~ % /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
curl: (60) SSL certificate problem: self signed certificate in certificate chain
More details here: https://curl.haxx.se/docs/sslcerts.html

curl failed to verify the legitimacy of the server and therefore could not
establish a secure connection to it. To learn more about this situation and
how to fix it, please visit the web page mentioned above.
```

Figure 1: Erreur HomeBrew

Tapez la commande suivante :

```
/bin/bash -c "$(curl -k https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Tapez ensuite votre mot de passe utilisateur.

```
(base) eleonore@mbro-21-061 ~ % /bin/bash -c "$(curl -k https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
% Total    % Received % Xferd  Average Speed   Time    Time     Current
           Dload  Upload   Total   Spent    Left     Speed
100 28027 100 28027    0     0  226k    0 --:--:-- --:--:-- --:--:-- 228k
==> Checking for `sudo` access (which may request your password)...
Password:
```

Figure 2: Nouvelle commande HomeBrew

**Note 2:** Si votre environnement possède déjà Homebrew, il est judicieux de le mettre à jour avec la commande:

```
brew update
```

**Note 3:** Quelques autres outils peuvent être utiles lors des utilisations de SUMO. La commande suivante:

```
xcode-select --install
```

permet d'installer le compilateur Clang, lequel permet de compiler les codes sources C++ de SUMO.

La commande suivante:

```
brew install cmake
```

permet d'installer CMake qui est un processus de compilation du logiciel

Afin d'exécuter et de compiler SUMO, les commandes suivantes peuvent être exécutées. Ces commandes permettent d'installer des *dependencies* dans l'environnement:

```
brew install --cask xquartz
brew install xerces-c fox proj gdal gl2ps
brew install python swig eigen pygobject3 gtk+3 adwaita-icon-theme
python3 -m pip install texttest
```

## 1.2 Construction de SUMO

La commande suivante permet de cloner dans le répertoire `./sumo`

```
git clone --recursive https://github.com/eclipse/sumo
export SUMO_HOME="$PWD/sumo"
```

Créez ensuite le répertoire qui contiendra les librairies et fonctions SUMO activées par le processus de configuration de CMake en exécutant les codes suivants :

```
cd $SUMO_HOME
mkdir build/cmake-build
cd build/cmake-build
cmake ../.
```

Lancez le processus de construction de SUMO depuis CMake :

```
cd $SUMO_HOME/build/cmake-build
cmake --build . --parallel $(sysctl -n hw.ncpu)
```

### 1.3 Etapes optionnelles

D'autres commandes optionnelles au lancement de SUMO sont disponibles dans la [documentation](#), comme l'installation de Traas (TRaci As A Service).

## 2 Sous Windows

Il existe quatre *packages* différents pour l'installation de SUMO selon votre plateforme (32 ou 64 bit, à regarder sur votre [version](#)) disponibles sur la [documentation](#). Il faut également spécifier les chemins des paramètres de SUMO dans [windows](#) pour pouvoir lancer correctement des commandes depuis le terminal. Au besoin, pensez également à télécharger un éditeur de texte (comme [Notepad++](#) ou BlocNotes) pour ouvrir et modifier vos fichiers XML par la suite.

## 3 Sous Linux

Les étapes à suivre sont complètement détaillées dans la [documentation](#).

## 4 Initiation SUMO

Il existe diverses documentations sur l'utilisation de SUMO, voici quelques références:

- [SUMO User Documentation](#), avec un [exemple](#).
- Guide pour le logiciel SUMO du cours [CIV8740](#)

Le lancement d'une simulation SUMO via l'application SUMO\_GUI requiert:

- Un fichier .net définissant le réseau. On peut visualiser et modifier le réseau via l'application Netedit, ou également via les éditeurs de texte (comme TextEdit sous Mac et BlocNotes sous Windows).

- Un fichier .rou définissant la demande, à savoir le nombre et le type de véhicules sur le réseau. On peut également directement modifier ce fichier via les éditeurs de texte.
- Un fichier .sumocfg qu'on lance via SUMO-GUI pour faire tourner la simulation, aussi modifiable via les éditeurs de texte. Il contient notamment les noms des fichiers définissant le réseau et la demande.

Pour vérifier que l'installation de SUMO est complète et que toutes les applications associées fonctionnent bien vous pouvez:

- Lancer le fichier reseau.net via l'application Netedit et observer comment le réseau est défini.
- Lancer le fichier demande.rou via un éditeur de texte (TextEdit ou Bloc-Notes) pour regarder les informations qu'il contient
- Enfin lancer le fichier simulation.sumocfg via SUMO-GUI ou encore via le terminal de commande avec la commande "sumo -c simulation.sumocfg".

## 5 Initiation à TRACI

"Traffic Control Interface" permet d'accéder, de consulter et/ou de modifier l'état des variables (véhicules / boucles électromagnétiques / infrastructures) propres à une simulation, à chaque pas de temps. En pratique, cela permet de consulter et modifier de façon dynamique la simulation, alors même qu'elle est en train de tourner. Ceci est particulièrement utile pour récupérer les données des objets simulés et contrôler leur comportement pendant la simulation.

Pour lancer la simulation test avec le script python, les étapes sont les suivantes :

- Ouvrir le fichier TP\_ITS.py;
- Modifier la Ligne 10 avec le chemin qui mène au dossier sumo :

```
os.environ['SUMO_HOME'] = '/YourPath/sumo'
```

- Ouvrir le terminal de commande et indiquer le chemin au dossier contenant votre script python, le fichier réseau, le fichier demande et le fichier simulation à l'aide de la commande suivante :

```
cd /YourPath/
```

- Lancer la commande suivante :

```
python TP_ITS.py
```