

VEINS ET SUMO INTRODUCTION

Ce document se présente d'une manière générale Veins.

Il décrit également les fonctions principales de SUMO à travers différents exemples. Ce document ne fait pas le lien entre le simulateur de réseaux OMNeT++ et SUMO. Ce dernier sujet est traité dans un document séparé

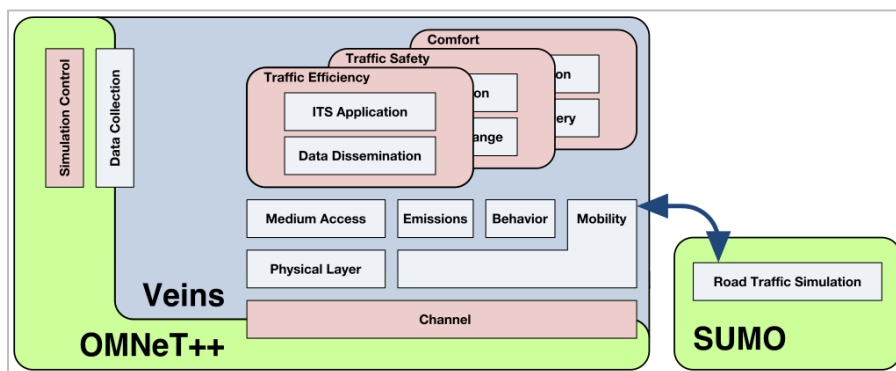
I. Présentation de Veins

Lien pour la documentation de [Veins](#).

VEINS est un framework de simulation de réseau de véhicules.

Les modèles sont exécutés sur le simulateur OMNeT++ (réseau) et interagissent avec le simulateur de trafic routier SUMO. Les deux simulateurs sont connectés avec le protocole TCP. Le mouvement des véhicules dans le simulateur SUMO est reflété comme le mouvement des nœuds dans une simulation OMNeT++.

VEINS est destiné à servir de base à l'écriture de code de simulation spécifique à l'application. Les modèles de simulation de Veins servent de boîtes à outils. Ils sont disponibles gratuitement.



II. Présentation de SUMO

Toute la documentation de [SUMO](#) est répertoriée sur ce lien. Cette partie suit la [vidéo](#) explicative de SUMO.

Afin de réaliser les exemples suivants il est recommandé de télécharger le dossier « SUMO 2020 » présent sur [ce lien](#) et d'extraire son contenu dans C:\Users\nom_user\Sumo.

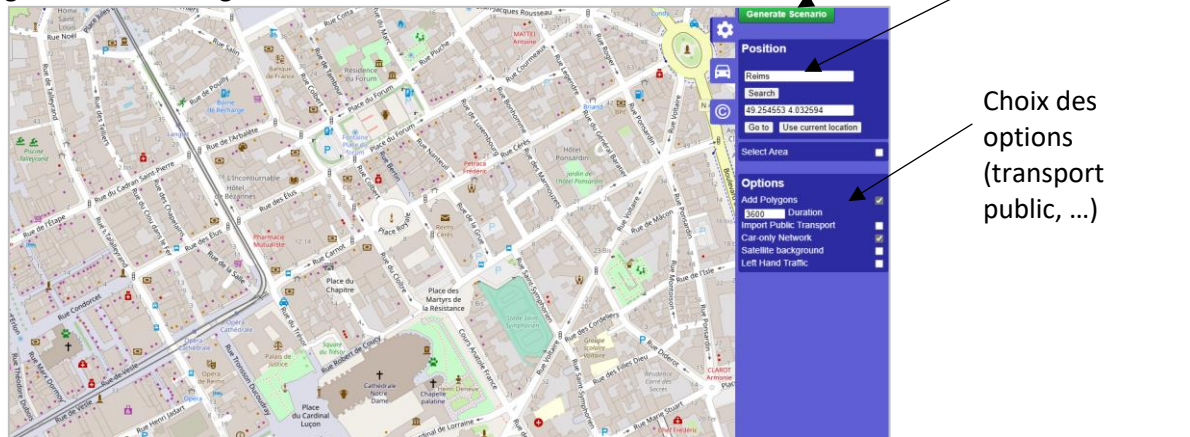
1. osmWebWizard

Utilise la base de données d'OpenStreetMap. Les données d'OpenStreetMap sont complétées par des contributions volontaires de personnes. Il est donc possible que certaines parties de la carte ne soient pas correctes.

a. Génération aléatoire d'une circulation dans Reims

Rentrer la commande suivante : `python %SUMO_HOME%/tools/osmWebWizard.py`

Un nouvel onglet sur votre navigateur devrait s'ouvrir :



Une fois vos paramètres rentrés appuyer sur « Generate Scenario ». Cela vous ouvre sumo-gui :



Cela crée un nouveau dossier ayant pour nom la date du jour où vous avez généré la simulation pour la première fois. Ce nouveau dossier est donc dans : %SUMO_HOME%.

Ce dossier contient le fichier de configuration `osm.sumocfg`, qui est lu/chargé dans `sumo-gui.exe`, ainsi que d'autres fichiers de configurations.

Explication des différents fichiers contenus dans le dossier :

<code>osm.net.xml</code>	Fichier de simulation du réseau.	
<code>osm.passenger.trips.xml</code>	Générer aléatoirement	Fichier pour les passagers des voitures
<code>osm.pedestrian.rou.xml</code>		Fichier pour les piétons
<code>osm_pt.rou.xml</code>		Fichier pour les bus, trams...
<code>osm_stops.add.xml</code>	Fichier qui définit les infrastructures comme les arrêts de bus.	
<code>osm.poly.xml</code>	Fichier qui définit les contours des bâtiments, les points d'intérêts.	
<code>osm.view.xml</code>	Fichier qui reprend la visualisation de la carte (<i>exemple</i> : couleurs, ombres...).	
<code>osm_bbox.osm.xml</code>	Fichiers utiles pour rebuild le scénario.	Raw OSM
<code>osm.netccfg.xml</code>		Reconstruit le réseau et les stops.
<code>osm.polycfg</code>		Reconstruit les formes.
<code>build.bat</code>		Reconstruit la circulation.

b. Améliorer les données d'OpenStreetMap

Cette partie se base sur [la vidéo du Dr. Joanne Skiles](#). Cette méthode permet de modifier et améliorer les données de la carte directement à partir du fichier osm.

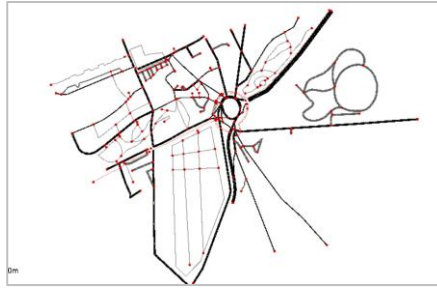
- 1- Ouvrir OpenStreetMap sur un navigateur : <https://www.openstreetmap.org/>
- 2- Trouver le lieu que vous voulez utiliser pour générer la carte. Cliquer en haut sur « Exporter » puis sur « Sélectionner manuellement une autre zone » dans la fenêtre de gauche.
- 3- Sélectionner la zone souhaitée et cliquer sur « Exporter ». Cela vous télécharge un fichier **map.osm** dans votre dossier de téléchargement.
- 4- Pour améliorer la carte il est possible d'utiliser JOSM qui est un éditeur java qui permet d'éditer les données venant d'OpenStreetMap. Voici les étapes à suivre pour télécharger JOSM :
 - a. Sur Windows vous pouvez :
 - i. Télécharger le fichier exécutable pour télécharger JOSM sur [ce site](#).
 - ii. Ou ouvrir l'application Microsoft Store de Windows et chercher « JOSM ». Cliquer dessus puis cliquer sur Installer.
- 5- Ouvrir dans JOSM le fichier **map.osm** téléchargé préalablement. La carte devrait s'afficher :



Il est possible de cliquer sur des bâtiments ou des routes pour avoir leurs informations disponibles. Ces informations sont des contributions rentrées par des utilisateurs. Il est donc possible, en fonction du lieu, que des informations soient manquantes. Pour se déplacer sur la carte utiliser le bouton droit de la souris.



- 6- Pour convertir le fichier **osm** en fichier **net.xml** voici la ligne de commande : `netconvert --osm-files map.osm --output-file C:\Users\nom_user\Sumo\reims.net.xml --geometry.remove --roundabouts.guess --ramps.guess --junctions.join --tls.guess-signals --tls.discard-simple --tls.join`
Il est intéressant de lire les warnings qui sont affichés lors de la conversion du fichier. Il est recommandé de modifier le fichier pour faire une carte plus propre.
- 7- Lancer la simulation du réseau avec la commande `netedit -s reims.net.xml` une fois dans le dossier où le fichier a été créé. Voici la carte obtenue :



8- Créer la circulation :

- a. Se placer dans le dossier qui contient le fichier python permettant de générer la circulation : `cd C:\Users\nom_user\source\sumo-1.8.0\tools`
- b. Rentrer la commande suivante : `python randomTrips.py -n C:\Users\nom_user\Sumo\reims.net.xml -e 1000 -o C:\Users\nom_user\Sumo\reims.trips.xml`
- c. Les convertir en fichier `rou.xml` : `duarouter -n C:\Users\nom_user\Sumo\reims.net.xml --route-files C:\Users\nom_user\Sumo\reims.trips.xml -o reims.rou.xml --ignore-errors`
Duarouter calcule les itinéraires les plus rapides / optimaux directement et itérativement dans le contexte de *Dynamic_User_Assignment*. Le fichier `rou.xml` contient les itinéraires de chaque véhicule.

9- Enfin, il faut créer le fichier de configuration SUMO :

- a. Créer un fichier de nom `reims.sumocfg`
- b. Le compléter ce fichier avec le code xml suivant :

```
<?xml version="1.0" encoding="UTF-8"?>
```

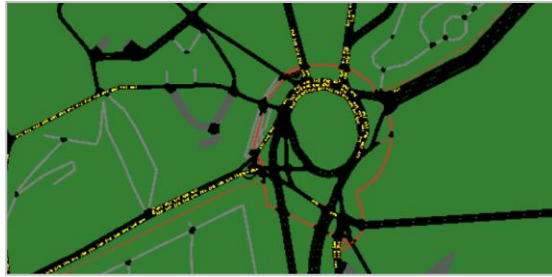
```
<!-- generated on Mon Sep 28 20:45:00 2020 by Eclipse SUMO sumo Version  
v1_7_0+0314-5b825ea2c9  
-->
```

```
<configuration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="http://sumo.dlr.de/xsd/sumoConfiguration.xsd">
```

```
  <input>  
    <net-file value="reims.net.xml"/>  
    <route-files value="reims.rou.xml"/>  
  </input>  
  
  <processing>  
    <ignore-route-errors value="true"/>  
  </processing>  
  
  <report>  
    <verbose value="true"/>  
    <duration-log.statistics value="true"/>  
    <no-step-log value="true"/>  
  </report>
```

```
</configuration>
```

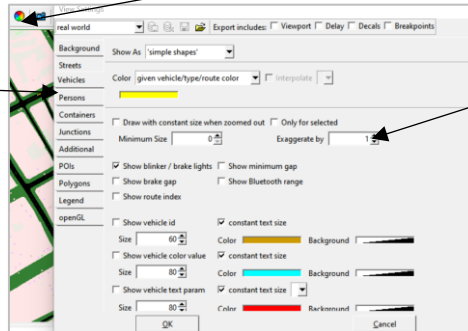
10- Enfin, tester la simulation en ouvrant dans `sumo-gui` le fichier `reims.sumocfg`. Voici le résultat :



c. Changer l'affichage graphique

Pour changer l'affiche (taille, couleur...) des voitures : cliquer sur le logo rond coloré >> vehicles.

Permet de paramétrer les piétons pour les simulations en comprenant.



Permet par *exemple* d'agrandir les voitures.

2. Editer le réseau – supprimer des routes

Les exemples de cette partie viennent de la simulation présente dans le dossier **Sumo2020_tutorial** et plus spécifiquement du sous dossier **02_netedit**.

a. Ouvrir les fichiers

Placer vous dans le dossier : `cd C:\Users\nom_user\Sumo\sumo2020_tutorial\02_netedit`

Puis rentrer la commande suivante : `netedit -s osm.net.xml`

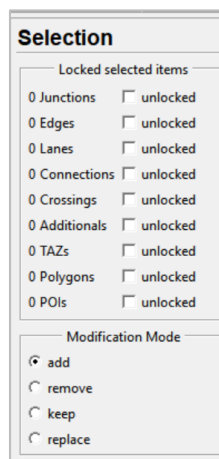
Cela permet de charger le réseau ainsi que les stops.

Si les stops ne sont pas chargés directement cliquer sur **File >> Additional and Shapes >> Load Additional >> osm_stops.add.xml** pour les rajouter sur la carte.

b. Editer des routes

i. Supprimer des routes

Pour rentrer dans le mode de sélection taper sur « **s** » sur votre clavier. Cela devrait ouvrir une fenêtre à gauche de la carte.



Pour sélectionner des morceaux de routes :

- Cliquer sur le morceau que vous voulez. Pour désélectionner recliquer sur le morceau de route.
- Pour une sélection plus générale → **Shift + clique gauche**. Cela vous met a disposition un outil de capture.

Quand une route est sélectionnée elle devient bleue.

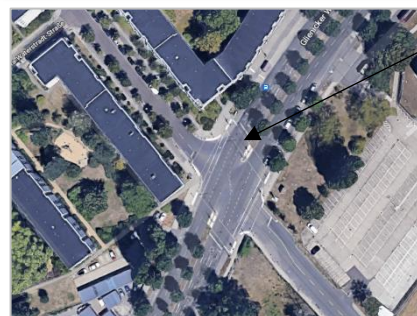
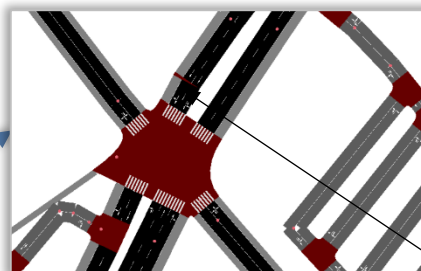
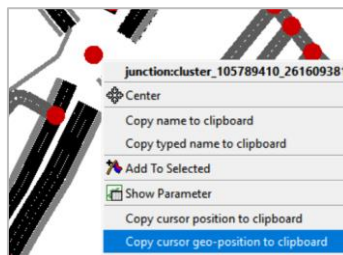


Pour supprimer ces routes il suffit de cliquer sur **Suppr.**

ii. Changer le réseau/les routes

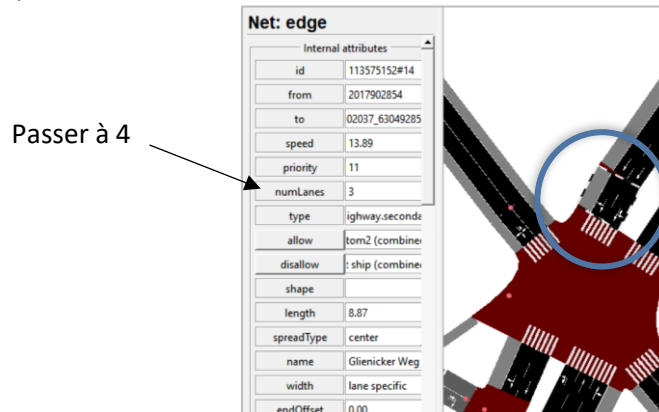
Se mettre dans le mode « Sélection » en cliquant sur « **i** ».

Se placer sur l'intersection encadrée. Pour information il est possible de récupérer les coordonnées sur le plan avec : clic droit sur l'endroit qu'on souhaite récupérer >> **Copy cursor geo-position to clipboard**.

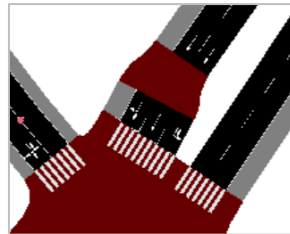


Normalement
3 voies (une
pour tourner)
et pas
seulement 2.

Cliquer sur le morceau de la route qui doit être modifié et changer le champ « numLanes » pour rajouter une voie de circulation routière. Le paramètre aura donc pour valeur 4 puisque la voie piétonne compte comme une voie :

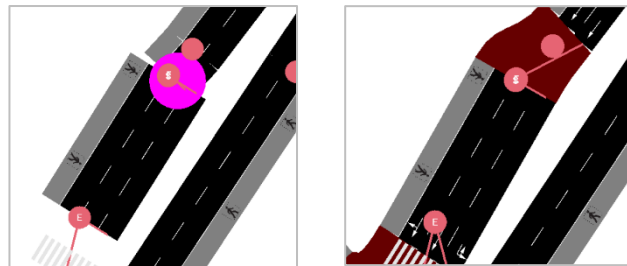


Pour réarranger l’affichage suite à la nouvelle dimension de la route, cliquer en haut sur **Processing >> Compute Junctions**. Cela donne :



Il est également possible de changer la géométrie :

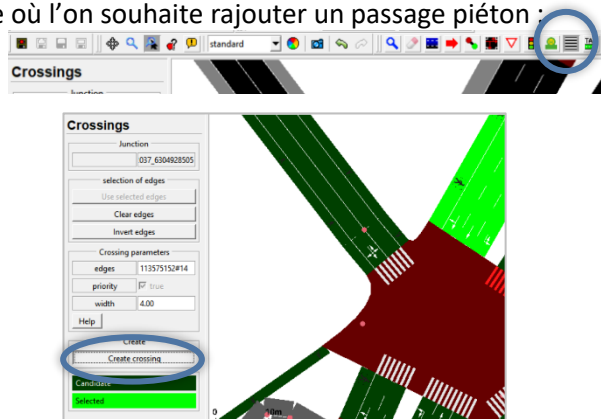
- Cliquer sur « m »
- Faire les modifications souhaitées (comme agrandir le morceau que l’on vient de créer) (image de gauche)
- Terminer avec **Processing >> Compute Junctions** (image de droite)



c. Rajouter des passages piétons

Dans la barre du haut cliquer sur le logo zébré pour afficher la fenêtre permettant le paramétrage des passages piétons : « **Crossings** ».

Il faut cliquer sur la route où l'on souhaite rajouter un passage piéton :



d. Editer les feus

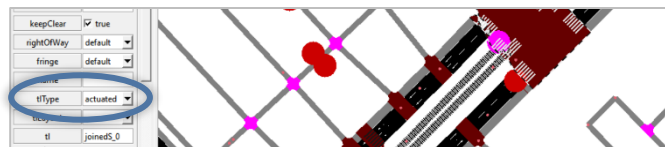
Pour cette partie les exemples sont tirés du sous-dossier 03_netedit. Rentrer la commande suivante une fois dans le sous-dossier correspondant : `netedit -s osm.net.xml`

Pour rentrer dans le mode « Edit Traffic Light » taper « t ».



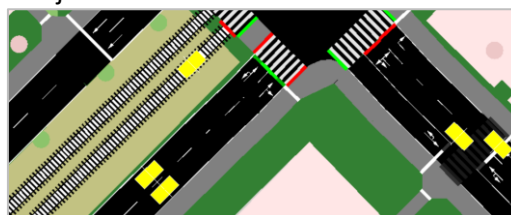
Puis aller dans le mode « Inspect », en tapant sur la touche « i ». Cela permet en cliquant sur l'intersection d'avoir le type de feu :

Ici le type est « actuated » ce qui signifie qu'il y a des détecteurs. Ces détecteurs permettent de remarquer qu'il y a une longue queue de voitures qui attendent au feu. Le feu passera donc au vert plus longtemps pour dégorger la route embouteillée.



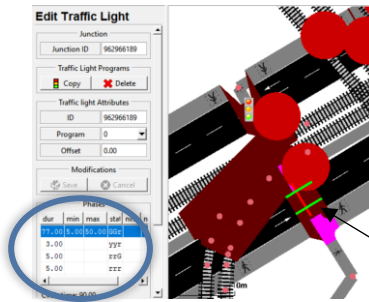
Pour voir ces détecteurs :

- Ouvrir `sumo-gui` et ouvrir le fichier `osm.sumocfg` du sous-dossier 03_netedit.
- Zoomer jusqu'à voir apparaître des bandes :
 - o Vertes : feu verts
 - o Rouges : feu rouges
- Cliquer gauche sur la bande de couleur souhaitée >> **Show Detectors**
- Les détecteurs s'affichent en jaune :



Le type « actuated » fonctionne mieux qu'un cycle statique.

Il est également possible d'avoir accès dans le mode « t » aux différents cycles du feu, avec leur temps :



Les flèches (haut et bas) permettent de visualiser sur la carte les feux (lignes rouges / vertes...). Il est possible dans le tableau de changer le temps de chaque étape, ou même de changer les feux.

e. Sauvegarder et régénérer une circulation aléatoire

Pour sauvegarder :

- Sauvegarder le réseau : File >> Save Network
- Sauvegarder les fichiers additionnels : File >> Additional and Shapes >> Save Additional

Il faut ensuite régénérer la circulation créée aléatoirement puisque certaines routes n'existent plus. Pour cela double cliquer sur le fichier

C:\Users\nom_user\Sumo\sumo2020_tutorial\02_netedit\build.bat . Attention, sumo-gui ne doit pas être ouvert pendant l'exécution de ce fichier.

3. Editer la circulation

Il y a trois éléments importants qui interagissent dans une simulation de circulation :

- Les voitures : leur origine, leur vitesse, leur destination...
- Les transports en commun : leur trajet, leurs arrêts...
- Les piétons : leur destination, leur moyen de déplacement (*Exemple* : voiture puis train puis marche) ...

a. Circulation provenant des comptages

Nous nous situons toujours dans le sous-dossier 03_netedit.

Méthode :

- Dans un premier temps passer en mode « Data » en cliquant sur « F4 ».
- Puis passer en mode « EdgeData » en cliquant sur le logo entouré ci-dessous :

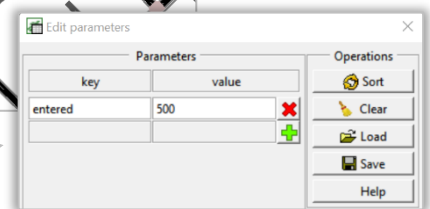


Créer une nouvelle base de données en cochant cette case dans la fenêtre de gauche. Lui donner un nom puis cliquer sur « Create DataSet ». Cliquer sur cette « Create new interval » pour définir le laps de temps où cette configuration prendra effet.



En dernier sélectionner les morceaux de routes qui seront affectées en cliquant dessus.

Choisir les paramètres en cliquant sur le bouton « Edit Parameters ». Une nouvelle fenêtre s'ouvre où il est possible de rajouter des paramètres avec le bouton « + »



⇒ Il faut donc trouver 500 voitures dans l'intervalle de temps choisi et dans les bords/morceaux de routes sélectionnés.

- Sauvegarder les données : File >> Data Elements >> Save Data Elements >> tmp.xml

([edgecounts.xml](#) est le fichier qui équivaut au fichier tmp.xml créé précédemment mais qui est fourni dans le dossier sumo2020_tutorial en guise d'exemple.)

Pour générer le nouveau trafic : (Ne fonctionne pas pour moi erreur)

- Il est possible d'exécuter une commande (ici dans le dossier 04_traffic.final) :
python %SUMO_HOME%/tools/routeSampler.py -r vehroutes.xml --edgedata-files [edgecounts.xml](#) -o passenger.sampled.rou.xml
- Ou exécuter directement le fichier de configuration 04_traffic.final/1.rscfg

Pour tester les configurations rentrées (comptage) ouvrir sumi-gui et charger le fichier 04_traffic.final/1.sumocfg.



On remarque que les véhicules sont principalement sur les axes choisis mais pas seulement. La simulation se termine quand 500 véhicules sont passés par les axes choisis.

b. Circulation provenant du comptage des tournants

Nous retournons pour faire l'exemple dans le sous-dossier 03_netedit et rentrer la commande `netedit -s osm.net.xml` si vous aviez fermé la fenêtre.

Méthode :

- Dans un premier temps passer en mode « Data » en cliquant sur « F4 ».
- Puis passer en mode « EdgeRelation » en cliquant sur le logo entouré ci-dessous :

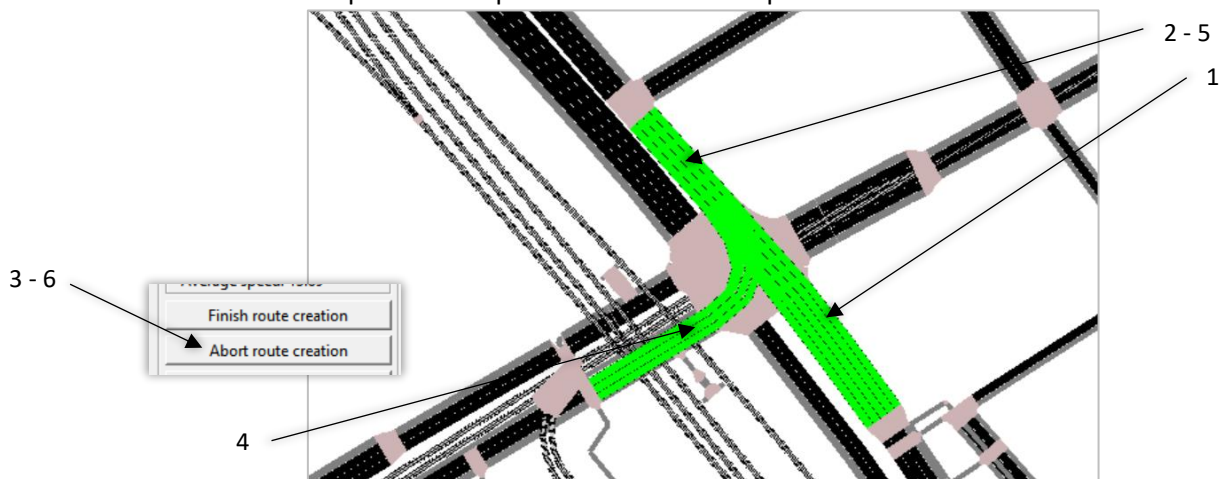


La méthodologie est exactement la même que pour la partie précédente (II.3.a) seul l'édition des paramètres changent :

Parameters		Operations
key	value	
count	500	Sort
		Clear
		Load

Puis, il faut choisir les tournants où la configuration s'appliquera. La sélection des tournants se fait toujours par la connexion de deux morceaux.

Suivre l'ordre des cliques suivant pour construire l'exemple :



Enregistrer les configurations.

Enfin, comme pour la partie précédente il faut générer le nouveau trafic avec la commande : `python %SUMO_HOME%/tools/routeSampler.py -r vehroutes.xml --edgedata-files edgecounts.xml -turn-files turncounts.xml -o passenger.sampled.rou.xml`

On remarque que `turncounts.xml` est le fichier qui reprend les configurations sur les tournants que nous venons de réaliser. Ce fichier est préalablement fourni dans le dossier `sumo2020_tutorial` en guise d'exemple.

Il est possible de charger les deux configurations en même temps (`comptage des morceaux de routes` + `tournants`)

Pour tester les configurations rentrées (comptage des tournants + comptages des edges/morceaux de route) ouvrir `sumi-gui` et charger le fichier `04_traffic.final/2.sumocfg`.



On remarque que la circulation est différente de la simulation précédente. En effet, il y a de la circulation sur le tournant sélectionné.