

# Modifications de topologies lors du découpage de tronçons

## Itinéraire parcourant un seul tronçon

Itinéraire dont la direction est la même que celle du tronçon

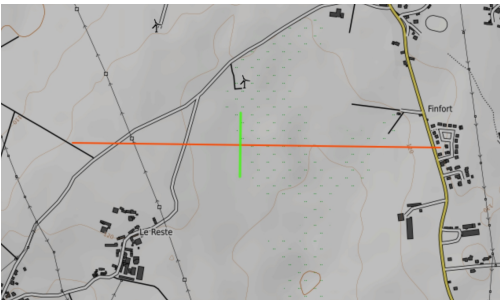
On part d'un tronçon simple allant de gauche à droite :



Sur ce tronçon, on crée un itinéraire allant dans la même direction :



On crée un nouveau tronçon qui coupe le premier tronçon en deux :



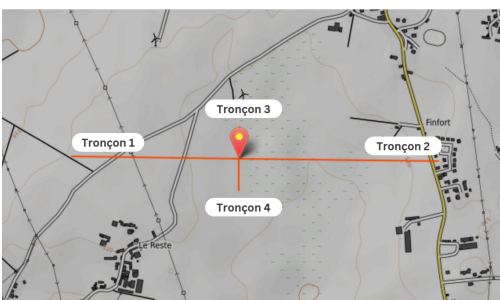
Sur la page de modification de l'itinéraire, il ne peut pas être réaffiché correctement car sa topologie n'a pas été mise à jour correctement.

Sa nouvelle topologie est la suivante :

paths : [2, 1] (liste des tronçons parcourus)

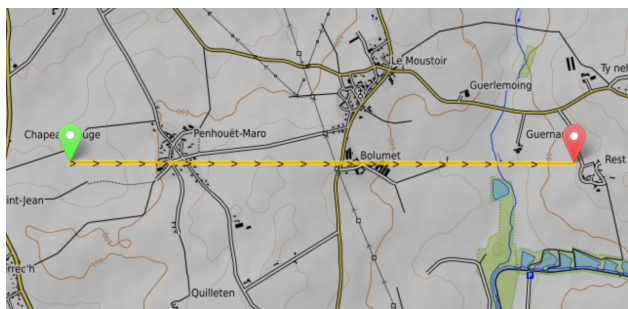
positions : {'0' : [0, 0.8], '1' : [0.1, 1]} (fractions de début et de fin pour chacun des tronçons)

L'erreur vient du fait que l'ordre des PathAggregations est incorrect.



## Itinéraire dont la direction est inversée par rapport au tronçon

Tronçon :



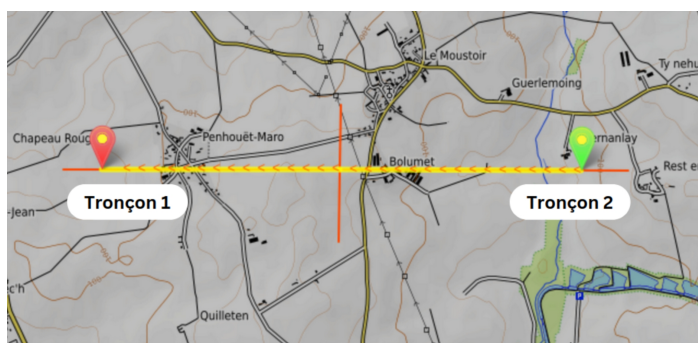
Itinéraire :



La topologie est correcte après sa mise à jour lors du découpage du tronçon :

paths : [2, 1]

positions : {'0' : [0.8, 0], '1' : [1, 0.1]}

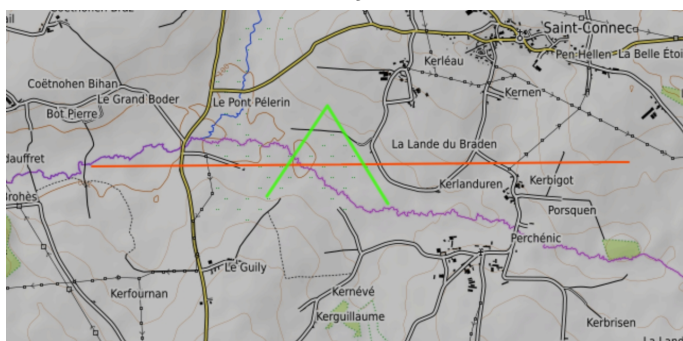


## Itinéraire parcourant un tronçon qui sera coupé plus d'une fois

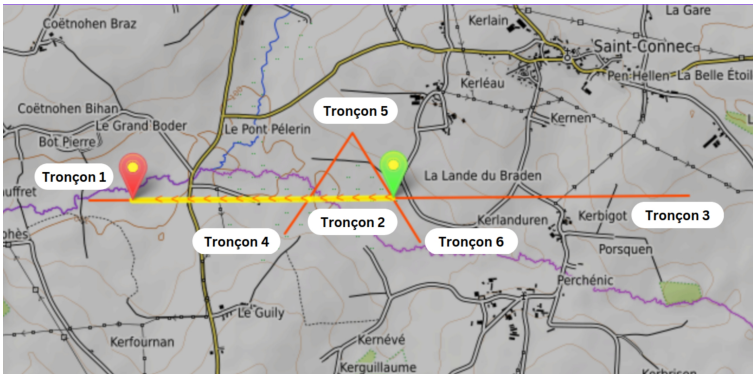
La topologie n'est pas correctement mise à jour si le nouveau tronçon coupe plus d'une fois le tronçon de base. Dans ce cas-ci, le problème survient quelle que soit la direction de l'itinéraire par rapport au tronçon de base.

Exemple :

Avec un seul nouveau tronçon, on coupe en 3 un tronçon sur lequel est créé un itinéraire :



Itinéraire affiché après découpage du tronçon (dans cet exemple la direction de l'itinéraire est inversée par rapport à celle du tronçon) :



Nouvelle topologie :  
 paths : [2, 3, 1]  
 positions : {'0' : [1, 0], '1' : [0.8, 0], '2' : [1, 0.2]}

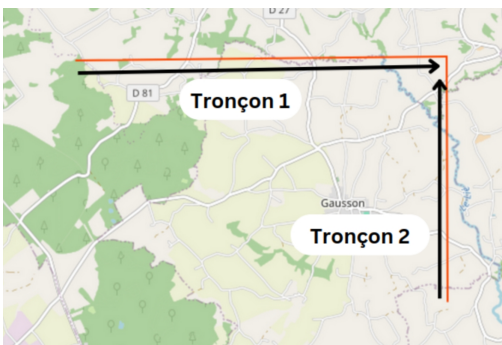
### Itinéraire parcourant plusieurs tronçons

Dans ce cas, la seule situation où la topologie est mise à jour correctement semble être lorsque tous les tronçons parcourus ont une direction opposée à celle de l'itinéraire.

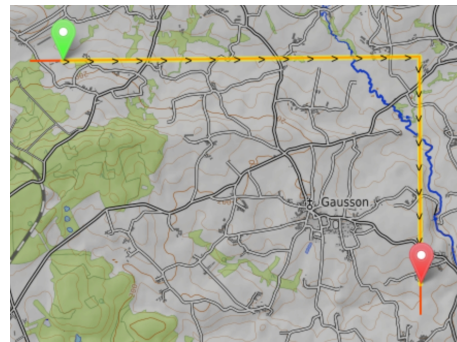
#### Cas à 2 tronçons

Cas 1 : 2 tronçons de directions alternées

Tronçons :

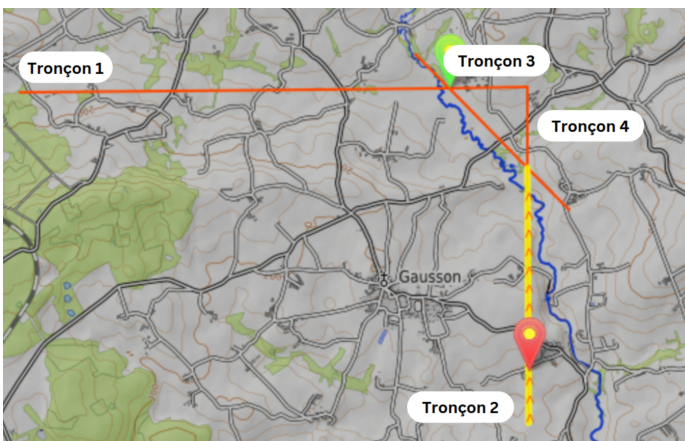


Itinéraire :



Nouvelle topologie :  
 paths : [3, 1, 4, 2]  
 positions : {'0' : [0, 1], '1' : [0.09, 1], '2' : [1, 0], '2' : [1, 0.2]}

Elle n'est correcte que sur les deux derniers tronçons (qui correspondent au tronçon originel dont la direction est inversée par rapport à l'itinéraire).

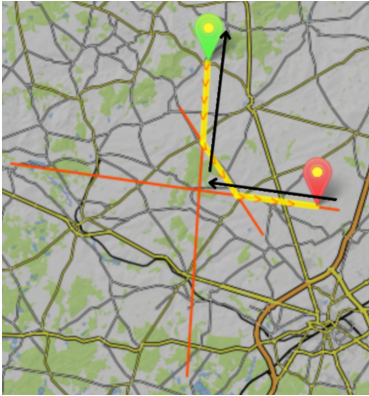


Dans cet exemple, le 1er tronçon parcouru a la même direction que l'itinéraire, et le 2ème a une direction opposée.

La topologie est également mise à jour de manière incorrecte dans les cas suivants :

- 2 tronçons avec 1er tronçon de direction opposée et 2ème de même direction ;
- 2 tronçons dont la direction est la même que celle de l'itinéraire.

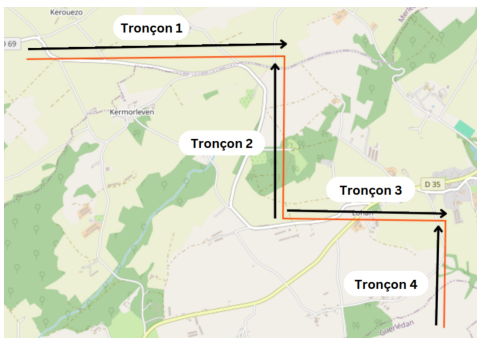
Cas 2 : 2 tronçons de directions opposées à celui de l'itinéraire  
(Les flèches noires indiquent la direction de chaque tronçon)



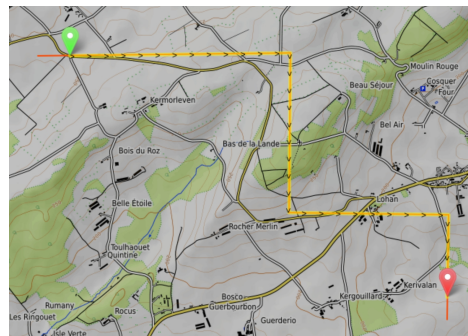
Résultats : la nouvelle topologie est correcte mais la page de modification de l'itinéraire recalcule la route.

### Cas à 4 tronçons dont les directions alternent

Tronçons :



Itinéraire:



Après ajout d'un tronçon intersectant les tronçons 1 et 2, sa topologie devient :

paths : [5, 1, 2, 6, 3, 4]

positions : {'0' : [1, 0], '1' : [0.1, 1], '2' : [1, 0]}, '3' : [1, 0], '4' : [0, 1], '5' : [1, 0.2]

La topologie est erronée sur les tronçons coupés, et ce même sur celui dont la direction est inversée par rapport à celle de l'itinéraire. Elle reste correcte sur les tronçons non modifiés.

