

Modélisation de trafic

Irénée Briquel

2013-2014

Description

L'objectif de ce projet est de modéliser une carte de ville, et ensuite d'une circulation automobile sur cette carte. L'objectif n'est pas de modéliser les interactions entre véhicules ou la gestion des intersections (on supposera que les véhicules se déplacent indépendamment les uns des autres, à vitesse constante sans interruption), mais de repérer les axes les plus empruntés avec une circulation aléatoire.

La carte

On pourra modéliser la carte par un ensemble d'intersections ou carrefours, ayant des coordonnées (latitude et longitude), et de tronçons de rues, menant en ligne droite d'une intersection à une autre. Les intersections serviront donc à marquer les carrefours, mais aussi à tracer des rues courbées.

En outre, chaque tronçon portera les informations suivantes :

- Un sens de circulation (sens unique ou double sens).
- Chaque tronçon appartient à une rue, qui porte un nom, et est constituée d'une suite de tronçons contigus.
- Les numéros de la rue présents sur le tronçon seront supposés répartis de manière équidistante tout le long du tronçon : un tronçon allant du numéro 14 au numéro 18 aura les maisons 14, 15, 16, 17, 18 réparties sur sa longueur.

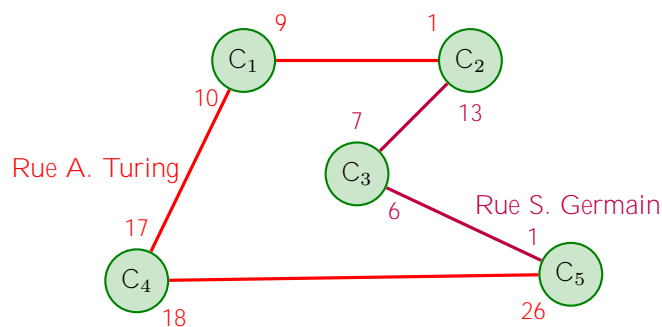


Figure 1: Un exemple de carte avec deux rues et cinq tronçons

Génération de la carte

Pour que ce programme soit capable de travailler sur différentes cartes, il sera nécessaire de fixer un format de fichier décrivant une carte, et de créer une fonction capable de charger un tel fichier et de créer la carte correspondante.

On pourra en outre implémenter une fonction vérifiant que le plan de la ville ainsi chargée ne présente pas de problèmes : tronçons qui se croisent, parties non atteignables, rues non continues ou avec numéros redondants...

Le trafic

Un véhicule sera caractérisé par une vitesse, et des règles de déplacement.

Il pourra par exemple

- Avoir une destination donnée par une adresse (numéro et nom de la rue), et une fonction pour s'y rendre (au cap, par un algorithme de recherche de chemin, aléatoirement...).
- Se déplacer aléatoirement, avec ou non des règles d'arrêt.

La simulation se fera en temps discret, et à chaque étape de temps, on calculera la situation au temps suivant : déplacement des véhicules, apparition de nouveaux et disparition de ceux ayant fini leur déplacement...

Une interface graphique

Une interface graphique permettra de visualiser la carte et un trafic de véhicules.

- Suivant la taille de la carte et le nombre de véhicules impliqués, il sera pertinent soit d'afficher les véhicules individuellement, soit de faire ressortir les tronçons très empruntés par des couleurs différentes pour visualiser les axes potentiellement congestionnés.
- On pourra proposer une interface pour permettre à l'utilisateur d'interagir avec la simulation (lancement/pause, ajout de véhicules, changements de paramètres...). On pourra aussi par exemple afficher une liste des véhicules circulant, et proposer d'ouvrir une fenêtre donnant des informations sur chaque véhicule.

Dépendances entre les tâches et extensions possibles

La première tâche indépendante des autres est la modélisation de la carte. On pourra commencer par créer une petite carte en dur dans le programme, avant de s'attaquer à la lecture d'une carte dans un fichier. L'interface graphique et la modélisation d'un trafic pourront ensuite être ajoutés.

Dans tous les cas, il est nécessaire, avant de se lancer dans la programmation, de définir précisément la spécification fonctionnelle du programme, qui est ici largement libre.

En extension, on pourra notamment travailler sur la création de cartes. On pourra par exemple proposer une interface de création/modification de la carte, et assurer la sauvegarde de cette carte au format défini. On pourra aussi, indépendamment du programme principal, chercher des cartes ou données extérieures (par exemple, une liste de villes et coordonnées GPS trouvée sur internet) et les convertir au format défini pour votre programme.

On pourra aussi rechercher à implémenter pour les véhicules des algorithmes de recherche de chemin optimaux, en veillant à leur complexité de calcul.