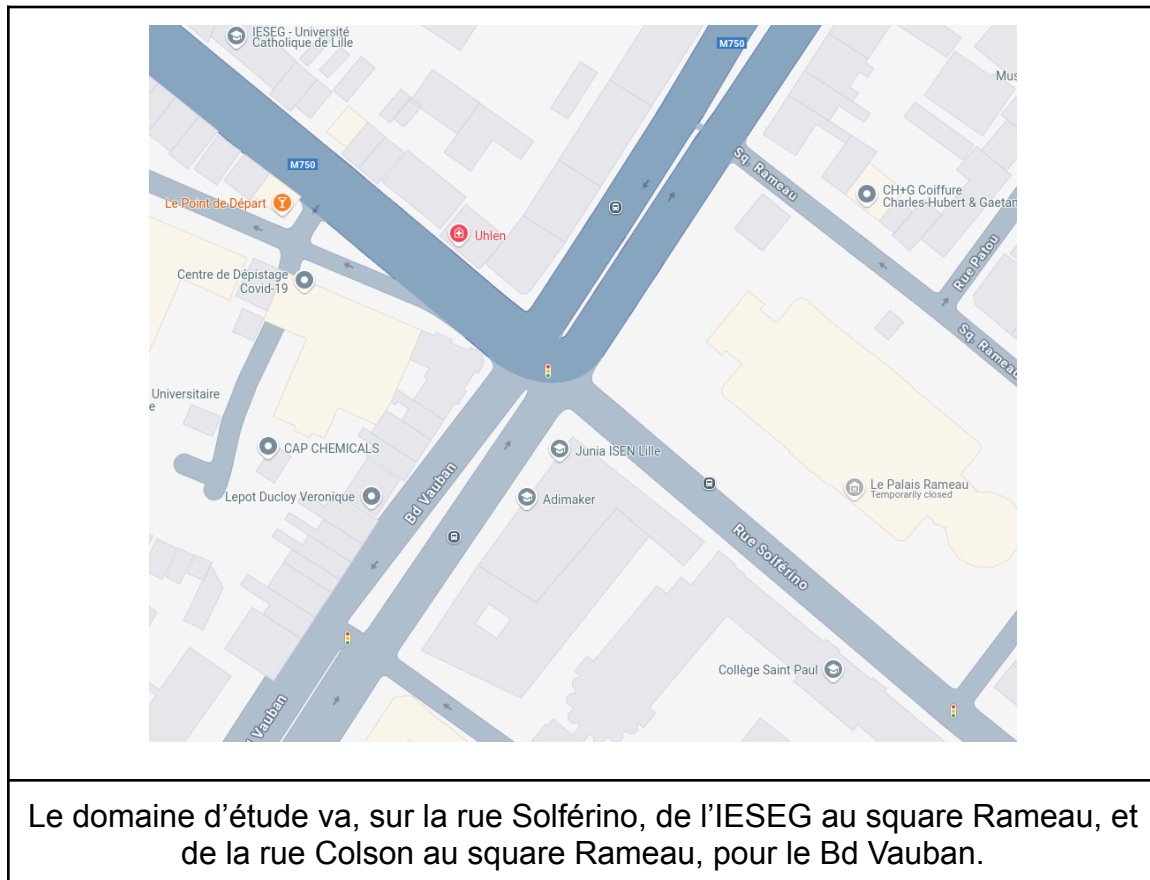


# Modélisation simplifiée de la circulation sur un espace public

## Présentation du projet

L'objet de ce projet est de modéliser le comportement d'utilisateurs du domaine public empruntant la rue Solférino, le Bd Vauban et le carrefour/rue Solférino (Figure ci-dessous).



Quatre types d'utilisateurs sont présents sur ces tronçons : piétons, cyclistes, véhicules légers et bus.

Le modèle de gestion proposé :

- Repose sur une supervision synchronisée par les feux de circulation (gérée par un thread) ;

- Suppose que chaque usager a un fonctionnement relativement autonome et circule sur les voies qui lui sont dédiées : routes pour les voitures, couloir de bus pour les bus, pistes cyclables pour les vélos, et trottoirs et passages protégés pour traverser les routes pour les piétons (chaque usager peut être modélisé par un « thread ») ;
- Le flux entrant à la fois sur le Bd Vauban et sur la rue Solférino est aléatoire. Il ne peut se faire que si un espace est disponible pour faire entrer un usager sur son domaine d'utilisation (trottoirs, routes, pistes cyclables) ;
- Le flux sortant se fait sans encombre ;
- Les acteurs circulant sur le domaine respectent le code de la route et s'arrêtent aux feux de circulation le cas échéant ;
- Les voitures et bus gèrent leur accélération (décélération) ainsi que leur vitesse.
- Les usagers se suivent sur les voies qui leur sont dédiées en conservant une distance de sécurité.

Le déplacement de chaque véhicule, véhicule léger et bus, est régi par une phase d'accélération, une phase éventuelle de parcours à une vitesse de fonctionnement constante, une décélération pour conserver une distance de sécurité, un arrêt au feu, le croisement d'une route.

L'accélération et la décélération sont considérées comme constantes en valeur absolue sauf en cas d'arrêt d'urgence (scénario à déterminer éventuellement) lors de laquelle la décélération peut être plus brutale.

Graphiquement, il est proposé d'utiliser une carte fixe sur laquelle les différents usagers se déplacent et qui montre graphiquement la position et l'état de chaque feu.

### **Travail à effectuer**

Ce projet laisse beaucoup de liberté sur son implémentation et certains aspects des comportements peuvent être simplifiés s'ils s'avèrent trop complexes à implémenter.

Les principales contraintes reposent sur l'autonomie des différents acteurs et la synchronisation de leurs actions respectives.

Comme nous l'avons vu en cours, il est possible de modéliser chaque acteur par un « thread » qui partage ses informations avec tous les autres par des variables, vecteurs, tableaux ... Par exemple, chaque usager (acteur) connaît l'état des feux ainsi que leurs positions respectives. Egalement, comme précisé ci-dessus, les comportements de chaque acteur, feux, véhicules, piétons, sont visibles graphiquement et en continu.

Pour faciliter l'intégration de ces interfaces et la gestion de ce projet, des outils/bibliothèques comme CMake, git, SFML, thread.... devront être utilisés. Des exemples d'utilisation des bibliothèques dans le contexte de ce projet sont fournis et seront complétés au fil des demandes de précision.

Vous pouvez également utiliser les bibliothèques de votre choix si elles sont disponibles pour les systèmes d'exploitation les plus courants comme Windows, Linux, MacOSX et si elles sont programmées en C++. N'hésitez pas à poser vos questions afin de pouvoir y répondre collectivement et générer une liste de type FAQ le cas échéant.

Quelques indications pour débiter le projet :

- Dans un premier temps, définir les données pour chaque acteur et les informations à échanger avec les autres acteurs.
- Par exemple, une voiture peut avoir comme caractéristiques : son identifiant, sa position, vitesse, elle doit se positionner par rapport aux autres véhicules et respecter les feux de circulation.
- Chaque feu de circulation est caractérisé par son état (vert, orange et rouge) et sa position. Ils sont synchronisés par un thread ;
- Parallèlement, s'informer de l'utilisation des outils graphiques et créer un dépôt Git.

## **Attendus du projet**

Ce projet doit répondre au cahier des charges ci-dessus et être réalisé dans le temps imparti (fin 2024). Il faut adopter une démarche assez conservatrice et ne pas se lancer dans des procédures/projets trop complexes. A l'issue de ce projet, il est impératif que vos programmes suivent le cahier des charges. Au-delà du respect total ou partiel du cahier des charges, il est indispensable que vos programmes fonctionnent correctement.

Utiliser les dépôts git pour conserver les branches de projet qui fonctionnent correctement avant de vous lancer dans de nouvelles implémentations.

Le projet est réalisé par un groupe de 2 étudiants. L'ensemble des programmes devra pouvoir être compilé en C++ avec des outils standards et les bibliothèques ci-dessus indifféremment sous Linux, MacOSX, Windows. En outre, il est impératif d'utiliser l'outil cmake (multiplateforme) pour la gestion de la compilation des projets. Enfin, les programmes seront complétés par la réalisation d'une capsule vidéo présentant votre projet.