



HAL
open science

Introduction aux techniques de navigation autonome pour les véhicules intelligents

Philippe Bonnifait, Clément Zinoune

► **To cite this version:**

Philippe Bonnifait, Clément Zinoune. Introduction aux techniques de navigation autonome pour les véhicules intelligents. Techniques de l'Ingénieur, 2021, Technologies de l'information, Réf: S7819 v1, pp.1-20. 10.51257/a-v1-s7819 . hal-03110973

HAL Id: hal-03110973

<https://hal.science/hal-03110973>

Submitted on 14 Jan 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Introduction aux techniques de navigation autonome pour les véhicules intelligents

Introduction to Autonomous Navigation Techniques for Intelligent Vehicles

par **Philippe BONNIFAIT**

Professeur des Universités

Heudiasyc UMR CNRS 7253, Université de Technologie de Compiègne (UTC), France

par **Clément ZINOUNE**

Ingénieur de recherche

Renault S.A.S., Centre Technique d'Aubevoye, le Val d'Hazey, France

Résumé

Cet article présente les notions de base d'autonomie opérationnelle et décisionnelle des véhicules intelligents évoluant sur des routes ouvertes à la circulation publique. Il décrit les principaux systèmes nécessaires à la navigation autonome de ces véhicules : représentation de l'espace, localisation, cartes, perception, décision, planification et contrôle. La plupart des notions sont liées au domaine de la robotique mobile, au contrôle et aux systèmes autonomes. Elles sont décrites d'une façon générale et des exemples concrets les illustrent. Un lecteur novice dans ce domaine pourra ainsi appréhender les différentes facettes de cette problématique complexe. Nous donnons également des éléments bibliographiques qui permettront au lecteur d'aller plus loin en fonction de ses centres d'intérêt.

Abstract

This article presents the basic concepts of operational and decision-making autonomy of intelligent vehicles operating on roads open to public traffic. It describes the main systems required for the autonomous navigation of these vehicles : world modelling, localisation, maps, perception, decision, planning and control. Most of the concepts are related to the field of mobile robotics, control and autonomous systems. They are described in general and concrete examples illustrate them. A reader novice at this domain will thus be able to understand the different facets of this complex issue. We also provide bibliographic elements that will allow the reader to go further according to his centers of interest.

Mots-clés

Robotique, véhicules intelligents, véhicules autonomes.

Keywords

Robotics, Intelligent Vehicles, Autonomous Vehicles

Table des matières

1	Introduction	3
2	Représentation de l'espace	4
2.1	Systèmes de coordonnées	4
2.2	Cartes de navigation vectorielles	5
3	Se localiser dans l'espace	7
3.1	Pose d'un véhicule	7
3.2	Localisation et estimation de pose	8
3.3	Intégrité	9
3.4	Localisation sur carte et map-matching	11
4	Percevoir et localiser les entités d'importance	12
4.1	Percevoir	12
4.1.1	Perception avec capteurs extéroceptifs	13
4.1.2	Perception avec des systèmes de télécommunication	13
4.2	Contextualiser les éléments perçus	14
4.3	Prédire la situation pour décider du comportement optimal	14
5	Naviguer dans l'espace	15
5.1	Décider de la manœuvre à adopter	15
5.1.1	Méthodes par apprentissage	16
5.1.2	Modélisation probabiliste	16
5.1.3	Système expert	17
5.2	Planifier une trajectoire	18
5.2.1	Problématique	18
5.2.2	Calcul du chemin de consigne	18
5.2.3	Profil en vitesse le long du chemin de consigne	23
5.3	Exécuter la trajectoire	24
5.3.1	Contrôle latéral	25
5.3.2	Contrôle longitudinal	27
6	Conclusion et perspectives	29
7	Glossaire	29

1 Introduction

Un véhicule intelligent autonome est un véhicule robotisé à capacité de décision et d'action capable de réaliser des tâches ou une mission qui lui sont confiées sans intervention d'un conducteur humain ou bien avec des interventions minimales, et ce sur des routes ouvertes à la circulation publique. L'adjectif « intelligent » fait référence aux systèmes de transport intelligents. Les principales classes sont les navettes autonomes ou les voitures autonomes même si les camions et les bus sont des véhicules qui peuvent bénéficier des fonctionnalités de navigation autonome.

Les premiers travaux fondateurs ont été réalisés dans les années 80 avec les prototypes « Navlab » de l'université américaine Carnegie-Mellon. L'Europe de son côté s'y est intéressée assez vite avec le programme européen Prometheus (1987-1996). Ces travaux ont donné lieu à beaucoup d'innovations en termes d'aide à la conduite. C'est dans les années 2000 que plusieurs challenges américains de la DARPA (*Defense Advanced Research Projects Agency*) en 2004, 2005 et 2007 ont montré que la technologie permettait de réaliser des « véhicules autonomes » sans personne à bord. Ainsi, depuis le début des années 2010, on assiste à des démonstrations de plus en plus abouties qui ont marqué le domaine comme l'*Intercontinental Autonomous Challenge* de l'université de Parme qui a fait rallier à plusieurs véhicules l'exposition universelle de Shanghai en 2010. A la même époque Google lançait son projet *Self-Driving Car* qui est en 2020, sous la marque Waymo, à sa cinquième génération. Puis, en 2014 le constructeur allemand Daimler a réalisé 100 Km avec très peu d'intervention du conducteur de sécurité lors d'un événement pour fêter le parcours historique de Bertha Benz [1]. Même si on pourrait citer beaucoup d'autres exemples, on constate que les véhicules particuliers se transforment progressivement en véhicules autonomes, c'est-à-dire qu'ils sont capables de prendre le contrôle des actionneurs manipulés habituellement par le conducteur dont le rôle, dans certaines tâches de conduite, se transforme en superviseur.

Il faut dire que le véhicule autonome peut apporter des réponses à des attentes sociétales : sécurité routière, confort de conduite, situations répétitives et ennuyantes, meilleure utilisation du temps passé à voyager, amélioration du trafic, optimisation de l'énergie embarquée et de l'usage de l'infrastructure, réduction des temps de parcours, amélioration de la mobilité des personnes âgées ou avec handicap, etc. Il existe également une volonté politique puisqu'en avril 2016 l'Union Européenne (UE) a établi la déclaration d'Amsterdam (engagement des états membres de l'UE de travailler à l'élaboration de règles et standards communs pour permettre la circulation des véhicules autonomes sur les routes européennes).

En robotique mobile, l'autonomie dépend principalement de trois facteurs :

- la capacité à évoluer indépendamment d'un humain,
- la complexité de la tâche ou de la mission à réaliser,
- la complexité de l'environnement d'évolution.

La *Society of Automotive Engineers* (SAE International) a publié en 2014 une classification en 5 niveaux d'automatisation de la conduite, le niveau 0 correspondant à une conduite complètement manuelle et le niveau 5 à un rêve idéal où le véhicule serait capable de naviguer de façon complètement autonome pour toutes les missions et dans tous les environnements [2]. Conceptuellement, plus le niveau est élevé, plus le véhicule a de capacités d'autonomie mais ces capacités sont définies dans des domaines de fonctionnement opérationnel clairement définis (*Operational Design Domain - ODD*). Le niveau d'automatisation qui nous intéresse le plus dans cet article est le niveau 4. Si nous prenons l'exemple d'une tâche de délégation de conduite, cela signifie que le véhicule doit rendre la main au conducteur en l'avertissant à l'avance lorsqu'il sort de son ODD ou bien qu'un aléa est apparu. Si le conducteur ne réagit pas, le véhicule doit réaliser une manœuvre de mise en sécurité. Dans la suite de cet article, nous passons en revue les fonctionnalités clés de l'autonomie des véhicules intelligents. Nous présentons les problèmes associés et nous présentons des solutions issues de notre expérience. Notre objectif est qu'un novice dans ce domaine puisse appréhender les différentes facettes de cette problématique complexe.

2 Représentation de l'espace

2.1 Systèmes de coordonnées

Bien maîtriser les systèmes de coordonnées est fondamental car la moindre erreur peut avoir un impact significatif sur la localisation et la gestion des données géoréférencées.

Quand on utilise des systèmes GNSS (*Global Navigation Satellite Systems*), les repères sont globaux à l'échelle de la Terre. Ils s'appuient sur des systèmes de référence géodésiques définis par :

- un repère géodésique cartésien attaché au globe terrestre,
- un ellipsoïde de révolution aplati modélisant le globe terrestre,
- un système de coordonnées et d'unités.

Le repère géodésique a son origine proche du centre de masse de la Terre, un axe z proche de l'axe des pôles, un axe x qui passe proche du méridien de Greenwich et l'axe y est tel que la base soit directe. On appelle généralement ce repère ECEF (*Earth Centered, Earth Fixed*) et on le note R_E [c5010].

Les différents GNSS (GPS, Glonass, Galileo et Beidou) ont des repères géodésiques proches mais différents. Les récepteurs GNSS sont capables de gérer les écarts et en général fournissent des résultats dans le système américain GPS (appelé WGS84).

Bien souvent les coordonnées sont fournies en coordonnées géographiques (longitude, latitude et hauteur ellipsoïdale - à ne pas confondre avec l'altitude définie par rapport au niveau moyen des mers dans une zone donnée).

En robotique, on préfère manipuler des coordonnées cartésiennes. Les procédures de passage des coordonnées géographiques aux coordonnées cartésiennes (et vice versa) sont parfaitement connues et il existe de nombreuses bibliothèques de calcul.

Par contre, manipuler des coordonnées ECEF n'est pas adapté aux véhicules intelligents et on préfère travailler dans un repère de navigation tangent à l'ellipsoïde géodésique de façon à manipuler facilement des coordonnées. En général, on choisit son origine proche de la zone d'évolution. Son axe z vertical est colinéaire à la gravité terrestre, son axe des abscisses x est orienté vers l'est et les ordonnées y vers le nord. On le note R_0 . Ce choix d'un repère ENU (*East-North-Up*) est très généralisé car il permet de représenter la situation vue d'oiseau comme lorsqu'on regarde une carte en papier.

Il faut veiller à manipuler un repère de navigation R_0 qui n'induit pas d'erreur de transformation. Il faut donc prohiber les projections. L'usage suivant tend à se généraliser. L'ellipsoïde s'approche localement à une sphère (ce qui est vrai à l'ordre deux) et on rajoute comme contrainte que les centres de l'ellipsoïde et de la sphère sont les mêmes (cf figure 1). A cause de cette simplification, le plan 2D n'est pas parfaitement tangent à l'ellipsoïde mais ceci n'a aucune conséquence. Dans ce cas, la matrice d'attitude ${}^E A_0$ de R_0 dans R_E conduit à une transformation simple et réversible sans erreur.

Transformation d'un point défini dans R_E (ECEF) dans R_0 (ENU)

En notant λ la longitude, ϕ la latitude, S le sinus et C le cosinus, la matrice d'attitude est :

$${}^0 A_E = \begin{bmatrix} -S\lambda & C\lambda & 0 \\ -S\phi.C\lambda & -S\phi.S\lambda & C\phi \\ C\phi.C\lambda & C\phi.S\lambda & S\phi \end{bmatrix}$$

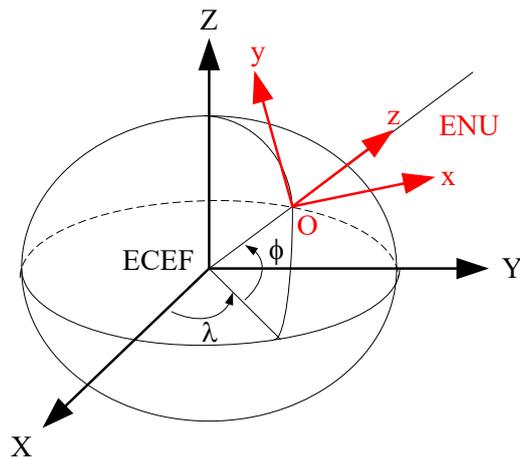


Figure 1 – Repères R_E (ECEF) dans R_0 (ENU). On peut noter l'approximation sphérique de la terre dans ce modèle qui n'introduit aucune erreur lors des changements de repères.

Les coordonnées du point ${}^E [X \ Y \ Z]^T$ en ECEF deviennent en ENU :

$${}^0 \begin{bmatrix} x \\ y \\ z \end{bmatrix} = {}^0 A_{E,E} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} X_0.S\lambda - Y_0.C\lambda \\ X_0.S\phi.C\lambda + Y_0.S\phi.S\lambda - Z_0.C\phi \\ -X_0.C\phi.C\lambda - Y_0.C\phi.S\lambda - Z_0.S\phi \end{bmatrix}$$

où ${}^E [X_0 \ Y_0 \ Z_0]^T$ sont les coordonnées de l'origine de R_0 dans le repère R_E
La transformation inverse est immédiate.

2.2 Cartes de navigation vectorielles

Les cartes de navigation numériques sont des composantes importantes pour le véhicule intelligent. L'ouverture des signaux GPS aux applications civiles ainsi que la miniaturisation de l'électronique embarquée a permis l'apparition de systèmes d'aide à la navigation dans les véhicules particuliers dès les années 1990. Le but de ces systèmes était initialement de calculer l'itinéraire et aider le conducteur à le suivre tout au long du trajet. Le réseau routier était alors numérisé sous forme de bases de données contenant les routes, les adresses, les points d'intérêt. La couverture de ces cartes est désormais mondiale et les informations associées aux routes se sont peu à peu enrichies (limitations de vitesses, restrictions de circulation, ...). Cependant, ces cartes sont toujours limitées à une représentation géométrique de la chaussée comme l'illustre la figure 2. Elles sont couramment désignée comme carte SD (*Standard Definition*) par opposition aux formats de cartes HD (*High Definition*) qui représentent géométriquement chaque voie de circulation du réseau routier [3].

L'apparition de fonctions d'aide à la conduite portant sur des tâches spécifiques de conduite (par exemple, l'aide au changement de voie ou le respect des limitations de vitesses [se3772]) et la recherche sur le véhicule autonome ont suscité le développement de cartes numériques haute définition portant sur de grandes zones géographiques. Le but est de représenter finement le réseau routier avec, par exemple, la géométrie (en trois dimensions) des voies de circulation, des marquages au sol, les limitations de vitesse, la présence de panneaux de signalisation, la présence de repères permettant au véhicule de se localiser, les points d'intérêt (zones de stationnement, zones piétonnes, ralentisseurs...). La figure 3 illustre ce concept. Il faut noter que dans certaines approches, l'acronyme HD est employé pour *High Density*. Dans ce cas, des nuages de points sont stockés comme une couche

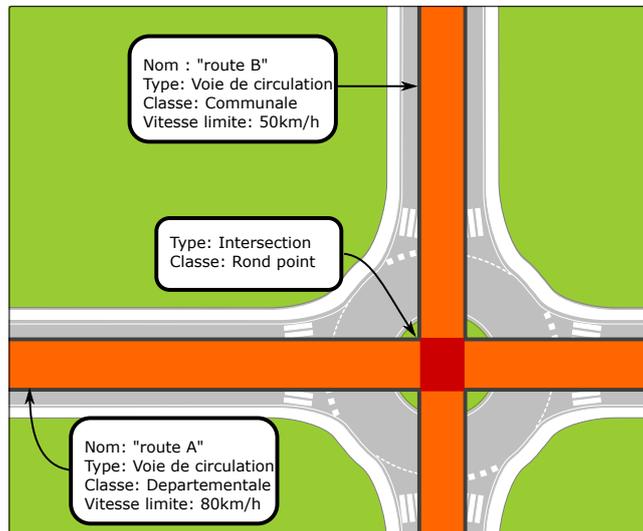


Figure 2 – Principe de représentation des données dans une carte de navigation en définition standard. Chaque chaussée est représentée par une polyligne (en orange) à laquelle sont associés des attributs (nom, type, classe...). Les intersections entre les chaussées (en rouge) possèdent leurs propres attributs

de la carte de navigation. Cela permet d'appliquer des méthodes de localisation dérivées du SLAM (introduit au paragraphe 3.2) mais représente un volume de données considérable.

Il n'existe pas encore de consensus sur la mise en œuvre de ces cartes qu'il s'agisse de format de fichier ou de contenu. Les formats *Navigation Data Standard* (NDS) [4] et *OpenDrive* [5] apparaissent comme largement utilisés par les fournisseurs de cartes. Ils permettent la gestion et le déploiement de grandes quantités de données. Cependant, ils nécessitent des infrastructures logicielles conséquentes au niveau des moteurs des bases de données.

Il est aussi possible de stocker les informations de la carte numérique dans des fichiers de texte structurés (tels qu'XML) ce qui facilite leur exploitation et permet une plus grande flexibilité au prix d'un volume de stockage plus important. L'ajout d'une nouvelle catégorie d'attribut entraîne simplement l'ajout d'une nouvelle balise dans le fichier. Il est alors possible de structurer les données contenues dans la carte pour faciliter son exploitation. Par exemple, l'approche définie par [6] propose une structuration efficace des voies par chaussées (les tronçons de voies sont nommés *Lanelets*). L'originalité de cette approche réside aussi dans le fait que les direction des vecteurs tangents à la voies sont stockées comme attributs des points formant la géométrie des voies. Cela permet d'éviter les discontinuités lors de la mise en correspondance de points avec la carte.

Certaines approches, inspirées du domaine des systèmes d'information géographique, utilisent des bases de données très compactes telles que *Spatialite* (voir chapitre 2 de [7]). Les éléments du réseau routier sont alors stockés dans des tables et leurs attributs dans des champs de ces tables. Ce format de base de données permet de réaliser efficacement des requêtes spatiales (sur le modèle de requêtes SQL). Ces fichiers de base de données sont autonomes (ne nécessitent pas de moteur de base de données). Ils permettent d'allier flexibilité et efficacité.

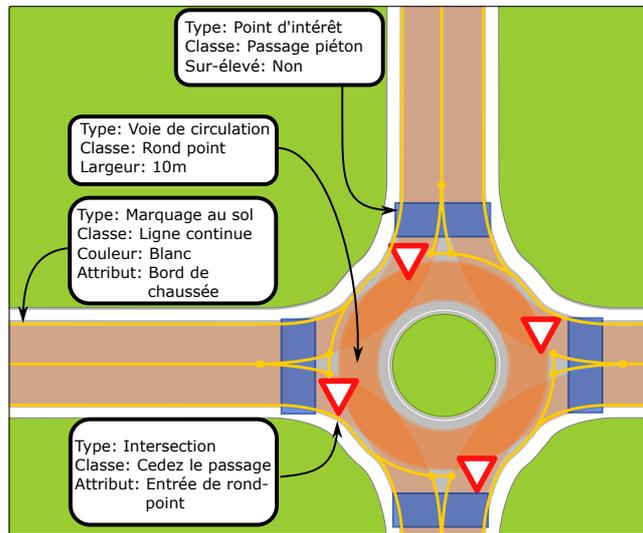


Figure 3 – Principe de représentation des données dans une carte de navigation en haute définition. Chaque voie de circulation (en rose) possède sa géométrie et son ensemble d'attributs propres. Les marquages au sol (en jaune), les points d'intérêts comme les passages piétons (en bleu) et les intersections peuvent être finement décrits.

À retenir

- Il faut être très vigilant sur les systèmes de coordonnées pour ne pas induire d'imprécision.
- Les projections sont à proscrire et il est recommandé d'utiliser un repère de travail.
- Les cartes numériques décrivant le réseau routier sont indispensables à la mise en œuvre d'un système de navigation.

3 Se localiser dans l'espace

Se localiser dans l'espace permet au véhicule de savoir où il se trouve, en particulier par rapport à sa carte de navigation. D'une façon générale, localiser un mobile signifie déterminer la position et l'orientation d'un repère qui lui est rigidement attaché dans un repère de navigation fixe dans l'environnement.

3.1 Pose d'un véhicule

Le repère lié au véhicule est appelé « repère mobile ». On le note R_M . Il est généralement situé au milieu de l'essieu arrière pour les véhicules routiers comme le montre la figure 4. L'axe des abscisses du repère R_M est colinéaire au vecteur vitesse \vec{v} , en négligeant les glissements.

La pose représente la position et l'attitude du mobile par rapport au repère de navigation R_0 . La position est exprimée en coordonnées cartésiennes. L'attitude est souvent notée avec 3 angles comme les angles de lacet, tangage et roulis du mobile par rapport au repère tangent local, mais il existe de nombreuses représentations comme les quaternions très utilisés en robotique notamment quand le mobile peut subir des rotations dans toutes les directions (comme un drone aérien par exemple). En effet, les quaternions permettent d'éviter le blocage de cadran des représentations utilisant des angles d'Euler ce qui constitue des singularités. En représentant une rotation par un quaternion unitaire (un angle de rotation autour d'un vecteur unitaire), il est facile de manipuler des matrices de rotation pour faire des calculs rapide et efficaces.

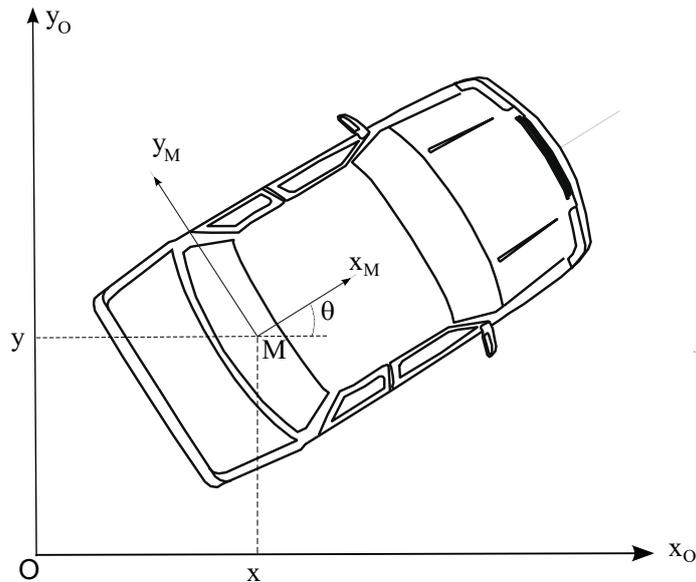


Figure 4 – Repère R_M lié au véhicule et repère R_0 tangent à la terre

Pour la navigation d'un véhicule intelligent, il est suffisant de travailler en 2D pour la très grande majorité des tâches à réaliser. Dans ce cas, on représente simplement la pose par un vecteur q à trois composantes sans utiliser de quaternion :

$$q = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \quad (1)$$

où θ est l'angle de lacet. En pratique, on manipule souvent des vecteurs d'état cinématiques qui s'obtiennent en ajoutant à la pose, la vitesse linéaire v du point M et la vitesse de rotation ω .

3.2 Localisation et estimation de pose

Le problème de la localisation est le problème de l'estimation de la pose du véhicule à partir de mesures de capteurs et de données géoréférencées statiques ou dynamiques comme des satellites par exemple. Pour les véhicules intelligents, ce problème est davantage abordé sous l'angle de la localisation aidée par la carte [8] que de celui de la localisation et de la cartographie simultanée [s7738, s7785] (SLAM en anglais) [9]. Les principales raisons sont liées aux nombreuses cartes disponibles et à la difficulté de trouver des caractéristiques statiques qui soient détectables par des modalités de perception embarquées assez divers (caméras et lidars par exemple) et en général différentes de celles qui ont été utilisées dans la phase de cartographie. Lorsque les cartes de navigation disponibles n'ont pas la précision et la qualité requises dans les zones d'intérêt, une phase de cartographie préalable est nécessaire [10].

Les méthodes de calcul sont actuellement toutes dynamiques, c'est-à-dire qu'elles ont un temps de convergence suite à une phase d'initialisation. On distingue actuellement deux grandes approches :

- celles par filtrage, typiquement filtrage de Kalman non linéaire [11],
- celle par optimisation, lissage et formalisation graphique de contraintes [12] [13].

Ces dernières permettent en particulier de corriger efficacement les estimations de pose lors de fermetures de boucles (c'est-à-dire quand le véhicule repasse en un point qu'il a déjà visité) par exemple dans un parking souterrain.

Les systèmes de localisation actuels combinent des capteurs de localisation à l'estime (comme des centrales inertielle et des systèmes odométriques à roues ou basés vision) et des capteurs de localisation absolue comme des récepteurs GNSS avec des antennes de qualité (souvent en mode différentiel par le code et par la phase via des corrections transmises sur des réseaux cellulaires) et des caméras et des Lidars capables de mesurer des angles et des distances sur des amers géoréférencés comme les panneaux ou bien les marquages routiers.

3.3 Intégrité

L'intégrité de la localisation consiste à fournir une mesure en temps réel du niveau de confiance à accorder aux estimations de la localisation lorsque le véhicule navigue de façon autonome. Elle fournit des informations qui permettent de savoir si les estimations de pose sont utilisables pour la tâche de navigation autonome actuelle. Différentes sources d'erreur existent. Une approche classique consiste à borner l'erreur d'estimation en fonction d'un niveau de risque admissible pour l'application. Ce niveau de risque peut être défini par un *Target Integrity Risk (TIR)* qui dépend du contexte de navigation. Le niveau de protection (*Protection Level PL*) est défini comme la limite statistique de l'erreur de position (*Position Error PE*) telle que la probabilité que l'erreur dépasse le *PL* (c'est le risque d'intégrité empirique *IR*) est inférieure ou égale au *TIR*. Par conséquent, pour un *TIR* choisi, le système satisfait à l'exigence d'intégrité si le *IR* est inférieur au *TIR* :

$$Pr(PE > PL) \leq TIR. \quad (2)$$

De plus, on définit une limite d'alarme (*AL* comme *Alarm Limit*) qui indique quelle est l'erreur maximale autorisée pour la tâche courante. Si $PL > AL$ alors le système de localisation est déclaré non disponible et le véhicule doit réaliser une manœuvre de sécurité comme un arrêt d'urgence. La figure 5 donne une illustration des différentes notions. Dans le domaine de l'aéronautique, on utilise des diagrammes de Stanford pour évaluer l'intégrité d'un système. Il s'agit d'un histogramme en dimension deux qui affiche le niveau de protection *PL* en fonction de l'erreur de position *PE*. On peut donc réaliser un tel diagramme quand on dispose d'une vérité terrain (c'est-à-dire un moyen de localisation beaucoup plus précis que le système en cours d'évaluation).

Pour les véhicules intelligents, les limites d'alarme sont très variables et pas clairement définies à ce jour. On utilise donc des diagrammes simplifiés comme celui de la figure 6. De plus, au lieu d'afficher un histogramme discret, on affiche ici un nuage de points. Les points qui se trouvent en dessous de la première bissectrice correspondent à des informations trompeuses, c'est-à-dire que la vraie position n'est pas contenue dans le domaine de confiance calculé par le système de localisation. De plus, pour que le diagramme soit pertinent il faut qu'il contienne un nombre d'échantillons très élevé pour qu'il soit statistiquement représentatif.

Les méthodes pour calculer les niveaux de protection *PL* en temps réel utilisent une première étape de détection de défauts (intégrité interne) pour éliminer les mesures aberrantes et une étape d'estimation de la borne de l'erreur de position (intégrité externe) [15]. Lorsque le *TIR* est assez grand (par exemple $10e-2$), cette dernière étape peut se faire avec des distributions gaussiennes ce qui est simple à réaliser (en multipliant l'écart type estimé par un bon coefficient). Quand le *TIR* devient très petit et que les exigences sont élevées, l'étape de calcul du *PL* se fait en général avec des distributions d'erreur à queue lourde comme les lois de Student.

Comparaison de l'intégrité de trois systèmes de localisation :

- La figure 6 montre que le système qui produit des points verts est plus précis que les autres car la distribution verte sur *PE* présente un pic plus proche de zéro et elle est plus concentrée que les deux autres. On voit aussi que ce système calcule des intervalles de confiance plus petits que les autres car la distribution de la borne d'erreur *PL* est la plus basse. Ce système lèvera donc moins d'alarmes que les autres. Dans ce cas, la

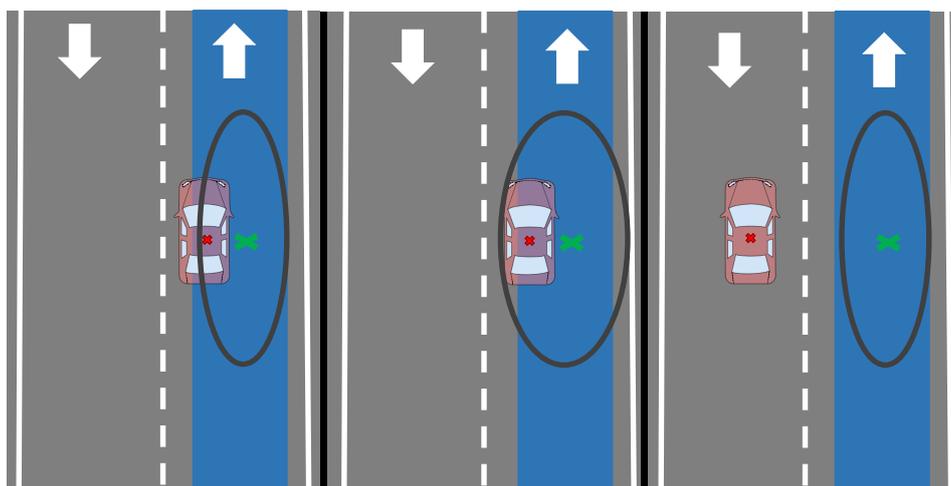


Figure 5 – Illustration des situations rencontrées avec un contrôle d'intégrité. Le couloir dans lequel le point de référence du véhicule doit se situer est représenté en bleu. Il définit la limite d'alarme. En vert, la position estimée et en rouge la position réelle. L'ellipse représente le domaine de protection. A gauche, la situation est normale. Au centre, l'incertitude est trop grande et le système de localisation est jugé non disponible. A droite, l'information est trompeuse (et la situation dangereuse) car le système de localisation pense que la voiture est bien localisée dans son couloir de navigation alors qu'elle est sur la voie de gauche.

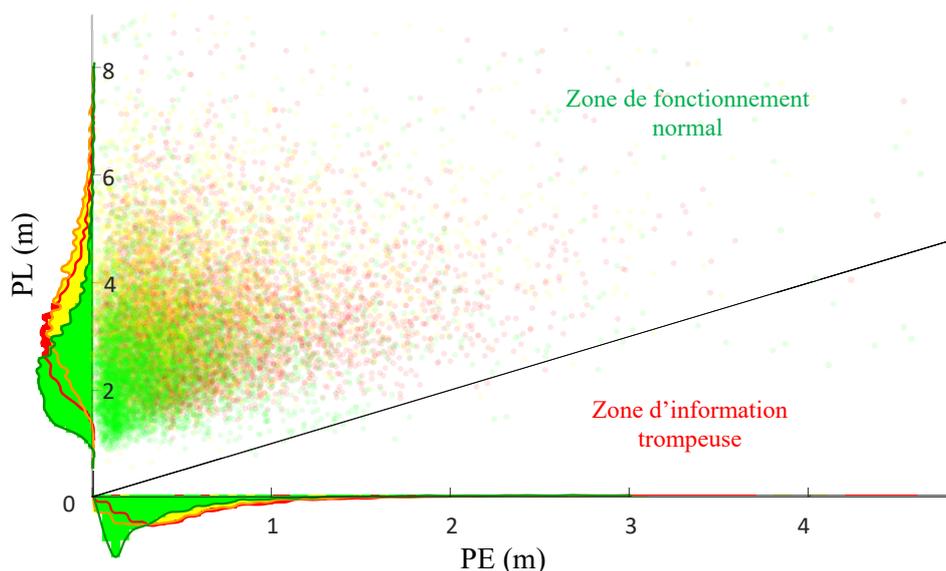


Figure 6 – Superposition de trois diagrammes de Stanford simplifiés (vert, jaune et rouge). Extrait des travaux d'Elwan Héry [14]. Un point au dessus de la droite indique que la vraie position est bien comprise dans le domaine de confiance calculé. Dans le cas contraire, l'information est trompeuse. Comme les nuages de points se superposent, on trace les distributions marginales sur les axes des abscisses et d'ordonnées. La comparaison est ainsi plus facile.

disponibilité du service de localisation sera plus élevée.

- Un diagramme de Stanford est utile également pour évaluer si les échantillons sont bien distribués dans le plan 2D. On voit qu'ici les distributions sont satisfaisantes et donc que l'évaluation de l'intégrité du système pourra être faite correctement (pour un TIR pas trop exigeant).
- Pour évaluer l'intégrité des différents systèmes, il faut estimer le pourcentage de points qui se trouvent dans la zone d'information trompeuse et vérifier que ce risque IR est inférieur au TIR choisi par l'application. Pour ce faire, il suffit de compter ces échantillons. Pour cette phase, le diagramme de Stanford n'est pas utile.

3.4 Localisation sur carte et map-matching

Lorsque la pose du véhicule est estimée de façon intègre dans le repère de travail, il faut souvent la mettre en correspondance avec la carte HD. C'est un problème d'association de données particulier qui consiste à chercher le segment d'une polyligne d'intérêt (comme par exemple le chemin de consigne ou bien la voie de navigation) qui est le plus proche de la pose estimée du véhicule comme le montre la figure 7.

Nous présentons ici une méthode simple qui consiste

- à adresser une requête (limitée à une zone géographique) au système qui gère la carte pour récupérer les polygones,
- à déterminer un critère à optimiser,
- à parcourir les segments de la polyligne d'intérêt,
- à sélectionner le segment qui minimise ce critère.

En général, le critère se base sur la distance et l'écart angulaire entre la pose estimée et chaque segment $[AB]$ extrait de la polyligne.

Formulation du calcul : Soit la pente α du segment : $\alpha = \text{atan2}((y_b - y_a), (x_b - x_a))$

Pour calculer l'écart signé e , il suffit de considérer le repère R_A d'origine A et orienté vers B .

La matrice homogène qui exprime ce repère dans R_0 est :

$${}^0T_A = \begin{bmatrix} \cos \alpha & -\sin \alpha & x_a \\ \sin \alpha & \cos \alpha & y_a \\ 0 & 0 & 1 \end{bmatrix}$$

Les coordonnées homogènes de M dans R_0 sont ${}^0M = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$

Dans R_A , on obtient ${}^AM = {}^AT_0 {}^0M$

L'écart signé e est alors simplement l'ordonnée de M dans R_A .

Quand le projeté orthogonal n'est pas situé sur le segment, la distance au segment n'est pas la norme $\|e\|$. C'est la plus petite des distances $\|MA\|$ et $\|MB\|$.

Pour une pose q estimée et pour un segment $[AB]$ donné, un critère simple (sans pondération) est

$$J(q) = |e| + |\text{submodpi2}(\alpha, \theta)| \quad (3)$$

La fonction « submodpi2 » calcule un écart angulaire entre $-\pi$ et π .

Soit S l'ensemble de tous les segments considérés, le segment map-matché est celui qui minimise le critère :

$$\text{map_matched_segment} = \arg \min_{s \in S} \{J\} \quad (4)$$

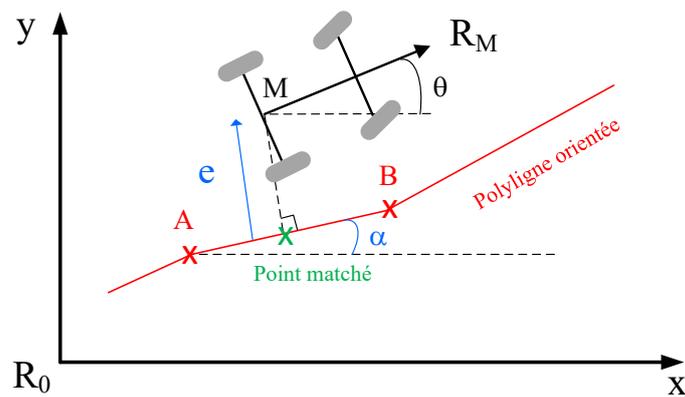


Figure 7 – Le problème du map-matching. Il s'agit de déterminer le point de la polyligne le plus proche de l'origine du repère du véhicule.

Pour aller plus loin

1. Parfois le map-matching est ambigu et plusieurs segments sont candidats compte tenu de l'incertitude de pose. Un algorithme qui renvoie en sortie tous ces segments est dit intègre [7]. C'est l'application cliente qui décide ou pas d'utiliser le résultat.
2. La méthode présentée utilise une projection orthogonale sur la polyligne. On peut améliorer grandement la précision longitudinale en utilisant une projection de type lanelet comme décrit dans [16].

À retenir

- La précision de la localisation doit être adaptée à chaque tâche de navigation.
- L'information de localisation ne doit pas être trompeuse.
- La localisation doit se faire en relation avec la carte de navigation.

4 Percevoir et localiser les entités d'importance

4.1 Percevoir

Le problème de la perception consiste à caractériser en temps réel ce qui ne peut pas être enregistré dans la carte (autres usagers comme des cyclistes ou des véhicules, l'état de la signalisation à message variable, etc.). Le but final est de caractériser l'espace libre roulant dans lequel le véhicule peut naviguer.

En d'autres termes, il s'agit de

- détecter et localiser les objets dynamiques,
- identifier l'état de la signalétique variable, comme l'état des feux,
- caractériser l'espace navigable.

Le premier problème fait référence à la notion de « carte locale dynamique » (CLD). Il s'agit d'une représentation dans laquelle les objets sont en général caractérisés comme suit :

$$o = \{x, P, e, c, z\} \quad (5)$$

où x est la pose cinématique de l'objet, P la matrice de covariance de l'erreur associée, e la probabilité d'existence, c la classe (voiture par exemple) et z un ensemble de caractéristiques comme la longueur, la largeur, etc. Une CLD est centrée sur le véhicule et contient tous les objets en interaction avec lui.

Le système de perception utilise des informations provenant :

- des capteurs extéroceptifs embarqués,
- des capteurs embarqués dans d'autres véhicules ou sur l'infrastructure dont les données sont transmises par télécommunication.

4.1.1 Perception avec capteurs extéroceptifs

Cette problématique est très large et a fait l'objet de nombreuses recherches depuis un trentaine d'années [17]. Les principaux capteurs extéroceptifs sont

- des caméras matricielles de type CMOS,
- des capteurs radars hyperfréquence,
- des capteurs à ultrasons,
- des capteurs lidars (lasers à balayage),
- des capteurs de vision non conventionnelle comme les caméras thermiques ou les nouvelles caméras à évènements donc chaque pixel n'émet un signal que lorsqu'il change d'état.

Les trois premières catégories sont bien maîtrisées et on trouve actuellement beaucoup de systèmes d'aide à la conduite commerciaux qui les utilisent. Pour la détection de feux tricolores, on renvoie le lecteur à l'article [18]. La perception détecte les objets mais les catégorise également (voiture, piéton, végétation, route, etc.) [19]. Actuellement, la très grande majorité des systèmes de perception à base de caméras utilise de l'apprentissage profond pour détecter les objets [20]. Il y a aussi une tendance à effectuer de la segmentation sémantique de la scène de conduite de façon dense au niveau des pixels [21] afin d'avoir une information de perception la plus riche possible.

Les capteurs lidars ont considérablement progressé depuis les années 2010. Il est maintenant très courant d'utiliser des capteurs rotatifs sur 360 degrés multi-faisceaux (plus d'une trentaine). On obtient en temps réel des points avec leurs coordonnées en trois dimensions dans le repère du véhicule. Les approches géométriques pour traiter les nuages de points (comme [22]) sont actuellement les plus performantes mais avec l'arrivée de lidars très denses, il faut s'attendre à ce que les méthodes par apprentissage conduisent à des performances bien supérieures [23]. La technologie lidar est très populaire actuellement car, comme elle est active, elle est moins sensible aux perturbations lumineuses que la vision (mauvais temps, éblouissement, etc.). De plus, ces capteurs permettent de détecter très précisément les objets et en même temps de caractériser le sol ce qui permet de détecter des trous ou des nids de poules par exemple.

En général, un système de perception fournit une CLD dans un repère capteur qui se déplace avec le véhicule. Grâce à la connaissance de la pose du véhicule et de la pose du capteur dans le repère mobile, la CLD peut être transformée dans le repère de travail R_0 ce qui facilite les traitements de la prise de décision. Dans ce cas, un étalonnage extrinsèque très précis du capteur est nécessaire.

4.1.2 Perception avec des systèmes de télécommunication

Il existe des standards de communication (notamment le SAE J2735) pour les systèmes de transport intelligent. On parle de *Dedicated Short Range Communication* (DSRC 802.11p) ou de *Cellular V2X* (C-V2X) [24][te7620], ce dernier étant compatible avec la 5G.

Les principaux messages utiles pour les véhicules intelligents sont :

- les messages des feux communicants et autres panneaux intelligents (messages MAP/SPaT),

- les messages de localisation des autres usagers avec leur vitesse (*Cooperative Awareness Message* (CAM) et *Basic Safety Message* (BSM))
- les messages concernant les travaux ou les accidents (*Decentralized Environment Notification Message* (DENM))
- les messages d'échange d'information de perception (*Cooperative Perception Message* (CPM))

Utiliser des informations reçues par communication sans fil permet de voir plus loin et d'anticiper mieux la situation. Cela facilite aussi certaines tâches comme l'identification de l'état des feux tricolores. Enfin, c'est utile pour éviter des oscillations et des phénomènes de conduite en accordéon quand le flux est chargé.

Par contre, la fiabilité des informations reçues est une question centrale. Si ces informations proviennent de l'infrastructure, en général, elle sont dignes de confiance car les systèmes sont opérés par des organisations officielles. De même, si le véhicule navigue en convoi avec d'autres et qu'un protocole sécurisé a été établi, il peut faire confiance aux données reçues. Par contre, lorsqu'il interagit avec d'autres usagers dans un environnement ouvert, il faut mettre en œuvre des mécanismes pour estimer la confiance des données reçues et ainsi se prémunir des messages malveillants [25].

4.2 Contextualiser les éléments perçus

Les fonctions de perception telles que décrites dans le paragraphe précédent sont centrées sur l'obtention d'une mesure et d'un suivi le meilleur possible. En sortie du module de perception, on dispose d'une liste d'objets présents dans les environs proches du véhicule ainsi que d'un espace navigable et d'une description de l'état des feux tricolores. La qualité (dans l'espace et le temps) sont au centre des problématiques de perception. Le nombre d'éléments perçus peut être élevé, en particulier en milieu urbain. Il est nécessaire de ne prendre en compte que les éléments perçus pertinents pour la tâche de conduite du véhicule autonome afin d'alléger et de faciliter le processus de décision. Par exemple, une allée bordée d'arbres peut générer un grand nombre d'obstacles statiques en sortie de la perception mais qui sont peu importants pour le processus de conduite autonome.

Il est donc utile de déterminer l'importance de chaque élément perçu au regard de la situation de conduite du véhicule en le contextualisant. La connaissance fine du réseau routier grâce à l'usage d'une carte HD est essentielle pour cela.

Illustrons ces concepts sur la base d'un exemple. On considère un lidar 32 nappes situé sur le toit du véhicule qui scanne l'environnement à 10 Hz. A chaque acquisition d'un scan lidar, une transformation est appliquée pour transformer les coordonnées de tous les points du balayage à l'instant du dernier point reçu. Cela compense la distorsion du scan due au mouvement du véhicule pendant l'acquisition des données. Grâce à la connaissance de la pose du véhicule, le nuage de points est projeté dans la carte HD. Après avoir éliminé les points du sol, ils sont regroupés en clusters. Grâce à la carte HD, les clusters en dehors de la chaussée sont éliminés [26].

4.3 Prédire la situation pour décider du comportement optimal

Disposer d'une connaissance statique de la scène de conduite n'est pas suffisant pour prendre des décisions dans un environnement dynamique. Il faut que le véhicule soit capable d'extrapoler les positions des autres agents de la scène. Cela peut se faire grâce au pistage des objets qui permet de prédire sur un court horizon de temps où ils se situeront (typiquement sur 2 secondes). Le pistage classique consiste à appliquer un modèle d'évolution dépendant du type d'objet perçu selon un mécanisme reposant sur du filtrage de Kalman. L'avantage d'un tel formalisme probabiliste est qu'il est possible de gérer l'incertitude qui fait que les objets extrapolés occuperont davantage d'espace au fur et à mesure que la prédiction sera lointaine du fait de l'accumulation des erreurs.

Le pisteur peut également utiliser les polygones des voies centrales de la carte HD, sous hypothèse que les objets dynamiques restent dans leur voie de navigation. Cela permet d'augmenter la qualité

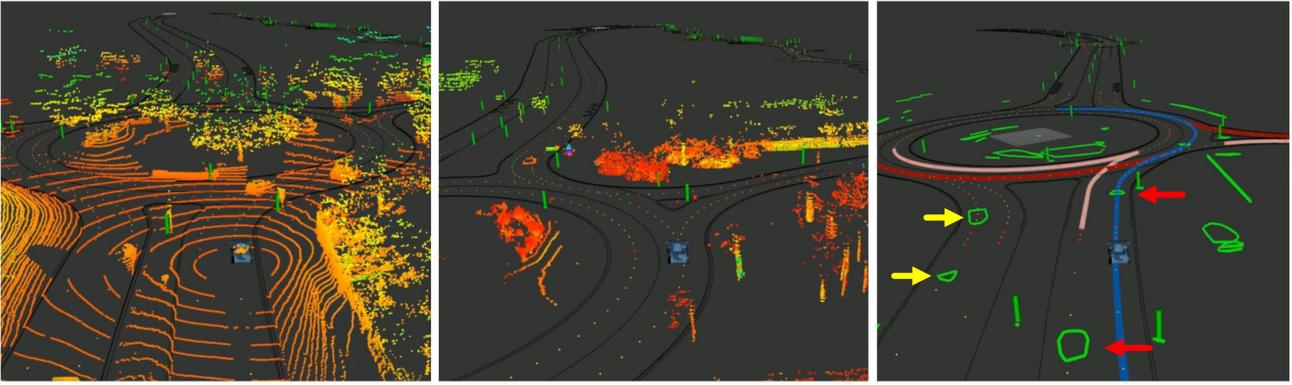


Figure 8 – Perception contextualisée. Le véhicule autonome est vu de l'arrière depuis un point haut. Il est affiché en bleu sur les figures (données réelles du laboratoire Heudiasyc affichées par Corentin Sanchez). De gauche à droite : nuage de points lidar, élimination des points du sol, regroupement des points en clusters verts. Sur la figure de droite, les clusters sur la chaussée sont indiqués avec des flèches. Les flèches rouges pointent sur deux véhicules d'intérêt pour le véhicule autonome : un à l'arrière gauche en train de doubler et un autre à l'avant sur le chemin à suivre.

de la prédiction sur un horizon de temps plus élevé. Quel que soit le pisteur utilisé, ce sont les pistes valides qui seront utilisées par le planificateur. En effet, un pisteur est capable en général de suivre de nombreuses hypothèses mais il ne transmet en sortie que celles qui sont de confiance suffisamment élevée pour réduire le taux de fausse alarme.

Le problème exposé dans ce paragraphe est en fait d'une plus grande complexité. En effet, il existe de nombreuses interactions entre les agents qui partagent l'espace routier. Ainsi, le véhicule autonome du fait de ses décisions et de son comportement agit d'une façon ou d'une autre sur le comportement des autres usagers voisins. Pour qu'une méthode prédictive basée pistage soit adaptée, il faut que le cadencement des différents processus de navigation soit suffisamment rapide et réactif par rapport à la dynamique de la scène de conduite.

À retenir

- Un système de perception fiable est fondamental.
- La perception peut utiliser des capteurs embarqués et des informations reçues par télécommunication.
- Comprendre la scène de conduite, focaliser l'attention sur les objets importants et prédire leur comportement permet d'adopter une navigation plus proche d'une conduite humaine.

5 Naviguer dans l'espace

5.1 Décider de la manœuvre à adopter

L'étape de décision consiste à fixer un objectif intermédiaire pour le véhicule qui soit cohérent avec la situation courante et avec l'objectif global. Il est généralement formulé de manière sémantique sous forme de manœuvres telles que *rester dans la voie*, *changer de voie*, *dépasser un autre véhicule*, *céder le passage*, *s'arrêter au stop*, *accoster un trottoir*, *effectuer une manœuvre de stationnement*, etc. Cela permet de réduire le champ des possibles et donc la complexité des calculs de l'étape de planification (détaillée au paragraphe 5.2) ainsi que d'appliquer des méthodes de raisonnement et de planification d'actions abstraites. Comme toute application embarquée, la ressource de calcul nécessaire à l'exécution des algorithmes de décision est une contrainte importante. Cependant,

l'explicabilité des décisions ainsi que le domaine opérationnel dans lequel le système fonctionne sont des axes clés pour comparer les approches.

5.1.1 Méthodes par apprentissage

Dans cette catégorie de méthodes, la machine est programmée pour construire et exploiter une correspondance entre un ensemble de décisions et un ensemble d'observations. L'apprentissage est dit supervisé quand on fournit à la machine des couples {Observation, Décision} corrects. Dans le cas opposé de l'apprentissage non supervisé, la machine va déterminer des groupes de décisions sur la base d'un ensemble d'observations sans pour autant pouvoir les catégoriser explicitement [27]. L'apprentissage peut finalement se faire par renforcement [s7793] en attribuant un score aux décisions prises en analysant les actions réellement effectuées par le conducteur ou via des indicateurs tels que sortie de route, collision, inconfort.

Ces méthodes présentent l'avantage de ne pas nécessiter une modélisation fine des processus d'interaction et de décision. Cependant elles nécessitent un grand nombre d'itérations d'apprentissage pour converger vers un comportement convenable. La catégorisation de séquences comme étant convenables ou non est alors fastidieuse et coûteuse. La représentativité des séquences choisies pour l'apprentissage au regard de l'ensemble des situations que le véhicule peut rencontrer sur le réseau routier est difficile à prouver. De plus, ces approches ne permettent que très difficilement d'anticiper le comportement du systèmes en fonction de leur niveau d'apprentissage.

5.1.2 Modélisation probabiliste

Dans ce cadre de raisonnement, les interactions entre le véhicule (vu comme un agent) et son environnement sont considérées. A chaque instant, les actions prises par l'agent modifient l'état de son environnement. L'agent choisit alors ses actions en fonction de ce qu'il observe de son environnement et de l'estimation des conséquences qu'auront ces actions dans le futur. Les conséquences des actions ainsi que les transitions d'états de l'environnement sont décrites par des fonctions de probabilité dans les processus de décision markoviens (MDP) [28]. L'agent est alors programmé pour maximiser la probabilité d'un ou plusieurs objectifs tels que l'atteinte d'une destination, la non collision, le respect des règles de circulation. Cette approche suppose que l'agent peut mesurer tous les effets de ses actions sur l'environnement, ce qui n'est pas le cas pour des applications réelles. En effet, certaines grandeurs de l'environnement ne peuvent physiquement pas être mesurées par l'agent. D'autre part, les capteurs employés par l'agent induisent des erreurs de mesures dont les caractéristiques peuvent être variées. Les processus de décision markoviens partiellement observables (POMDP) adressent cette problématique en modélisant l'observation de l'environnement par une densité de probabilité [29] [30]. Les conséquences des actions de l'agent sur l'environnement sont elles aussi traduites par des fonctions probabilistes. Une fonction de récompense est aussi définie pour différencier les conséquences des actions (par exemple, les actions menant au but seront récompensées et celles créant des collisions seront pénalisées). Le processus de décision consiste alors à explorer le POMDP pour en déduire la séquence d'actions dont il est le plus probable de maximiser la récompense.

Cette approche permet de formellement tenir compte des modèles d'erreurs associés aux données perçues, ce qui est un atout important car les imprécisions sont souvent significatives dans le domaine de la perception embarquée. Cependant, la modélisation de toutes les conséquences possibles en une fonction de récompense unique reste un sujet ouvert : comment ordonner un franchissement de ligne continue, une collision avec un véhicule, une collision avec un piéton, le gain de quelques mètres sur la distance parcourue, ... ? Il est par conséquent difficile d'expliquer la décision prise par le système en situation de conduite complexe impliquant plusieurs acteurs.

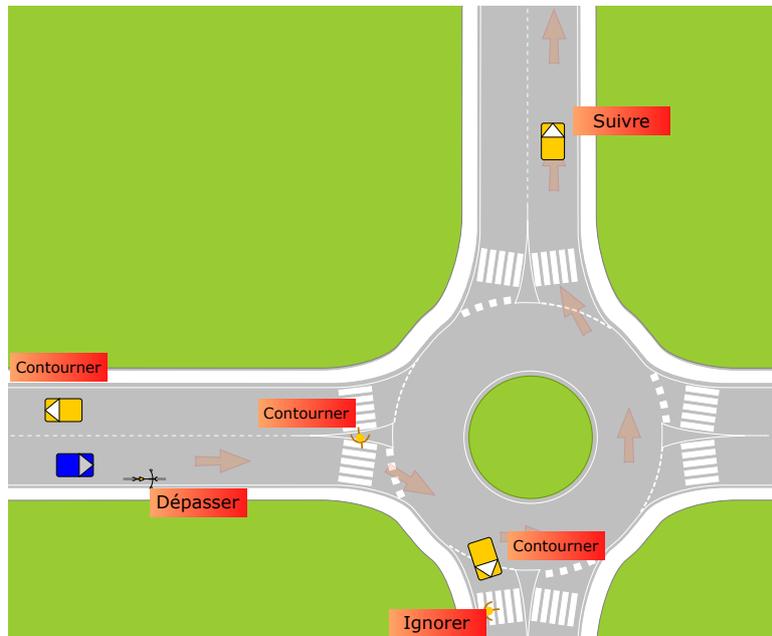


Figure 9 – Décisions résultantes d'un exemple de scène routière. L'ego-véhicule est en bleu à gauche de la figure. L'itinéraire qu'il va suivre est représenté par les flèches rouges. Les autres acteurs présents sur le réseau routiers sont représentés en jaune (véhicules, piétons et cycliste).

5.1.3 Système expert

La formalisation du processus de décision sous forme de système expert consiste à établir une correspondance entre les situations rencontrées possibles et les actions ou manœuvres que le véhicule autonome devra prendre. Du point de vue du système de décision, les données d'entrée sont des objets perçus pouvant entrer en interaction avec le véhicule. La programmation du système expert consiste donc à établir la manœuvre la plus appropriée étant donné la position, la vitesse, la direction, le type d'obstacles reçus ainsi que le contexte routier courant. Ces tables de correspondance sont créées lors de la conception du système de décision. Elles sont renseignées manuellement sur la base d'avis d'expert (d'où l'appellation *système expert*) qui se concrétisent, dans le cadre du véhicule autonome, en l'application du code de la route et en la transcription des usages habituels de conduite. Bien que le nombre de combinaisons possibles parmi les paramètres d'entrée soit assez élevé, cette méthode permet de maîtriser, lors de la conception du système, les décisions prises par le véhicule autonome. Cela permet d'envisager la certification de ce système. D'autre part, lors de l'exécution ou *a posteriori* dans le processus d'analyse d'incident, il est possible d'expliquer de manière déterministe les conditions qui ont mené à chaque décision prise par le véhicule. L'exécution de systèmes experts a, pour finir, l'avantage d'être très rapide.

Nous prenons ici comme exemple l'application d'une approche « système expert » pour la scène routière illustrée par la figure 9. A cette étape, le véhicule connaît l'itinéraire qu'il va suivre. Chaque obstacle est donc traité au vu de cet itinéraire et des mesures disponibles (position, vitesse, cap, classe...). Nous faisons l'hypothèse que les manœuvres disponibles pour le véhicule sont telles que décrit ci-dessous.

- *Suivre* : rester dans la voie et adapter la vitesse pour préserver une inter-distance suffisante avec l'objet,
- *Dépasser* : suivre puis doubler l'objet si le contexte le permet,
- *S'arrêter* : stopper le véhicule à une distance convenable de l'objet,

vecteur vitesse \ position relative	sur la voie de circulation	sur la voie opposée	sur le trottoir de droite	sur le trottoir de gauche
dans la même direction	dépasser	contourner	ignorer	ignorer
dans la direction opposée	s'arrêter	ignorer	ignorer	ignorer
vers bâbord	s'arrêter	ignorer	s'arrêter	ignorer
vers tribord	contourner	s'arrêter	ignorer	s'arrêter
immobile	s'arrêter	ignorer	ignorer	ignorer

Tableau 1 – Exemple de table de décision pour les interactions avec des objets de type *Cycliste*. Ici on choisit de contourner le cycliste s'il est sur la voie mais se dirigeant vers la droite (tribord) afin d'anticiper le fait qu'il sera bientôt sur le trottoir de droite.

- *Contourner* : conserver la vitesse prévue en s'écartant si nécessaire pour éviter une collision,
- *Ignorer* : ne pas prendre en compte l'objet dans le calcul de trajectoire.

Il est alors possible de programmer le comportement à adopter pour le cas d'un objet perçu de type cycliste tel que décrit en table 1. Dans l'exemple de scène routière choisie ici, la décision choisie pour le vélo se trouvant immédiatement devant l'ego-véhicule (c'est à dire le véhicule sujet) est par conséquent *Dépasser*. En remplissant chaque cellule de ces tables, on s'assure qu'une décision est attribuée pour chaque cas de figure possible.

On constate ici que la contextualisation de l'objet est essentielle pour adapter la décision à prendre. En effet, une table de décision pour une classe *Automobile* sera significativement différente de celle pour une classe *Piéton*. De plus, l'intention estimée de l'objet (*a minima* son placement sur la chaussée et la direction de sa vitesse) permet d'adapter finement les décisions prises et de s'approcher ainsi d'un comportement humain.

5.2 Planifier une trajectoire

5.2.1 Problématique

La planification d'une trajectoire se traduit par la recherche d'un ensemble optimal de poses futures pour le véhicule. Étant donné que l'environnement change au cours du temps, que sa dynamique peut évoluer rapidement et que les mouvements des différents agents sont souvent corrélés, cette optimisation est complexe et sa résolution rarement compatible avec les contraintes temps réel.

Le problème est donc traité en deux étapes, à savoir la recherche d'un chemin optimal puis l'attribution d'une dimension temporelle à ce chemin (profil de vitesse par exemple) [31].

5.2.2 Calcul du chemin de consigne

Pour déterminer ce chemin, il est nécessaire de convertir la connaissance que l'on a de l'environnement du véhicule en un ensemble de contraintes et de rechercher une solution optimisant la satisfaction de ces contraintes. Ces contraintes peuvent être de type et de portée variés pour le véhicule autonome. En effet elle doivent permettre, entre autres :

- la non collision avec les obstacles perçus à leur position actuelle et estimée dans le futur,
- le respect des règles de circulation (non franchissement de lignes blanches continues par exemple),
- le confort à bord (adaptation progressive de la vitesse pour les virages serrés ou le franchissement des intersections par exemple),
- l'atteinte de la destination finale désirée pour le véhicule.

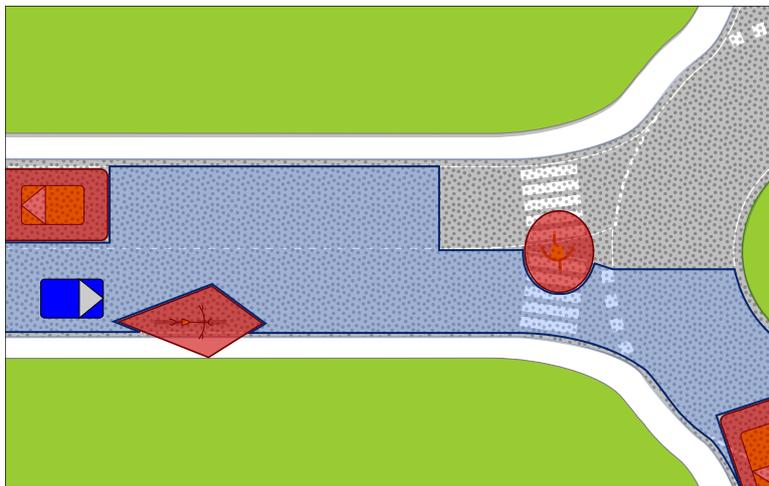


Figure 10 – Construction de l'espace navigable pour le véhicule. L'espace roulant est en noir pointillé, les zones de collision sont en rouge et l'espace navigable résultant est en bleu.

L'espace disponible pour le véhicule est identifié comme l'espace navigable. Il se définit comme étant l'espace roulant (bandeau routier) duquel sont soustraites les zones interdites (franchissement de lignes continues par exemple), les zones de collision avec les objets et les voies ne faisant pas partie de l'itinéraire prévu comme illustré par la figure 10.

La méthode des champs de potentiels est une approche directe pour trouver un chemin [32]. Le but est de pouvoir utiliser une transposition des lois physiques régissant le mouvement d'une boule sur une surface inclinée pour déterminer le chemin ralliant la destination. Un champ de potentiel est créé pour représenter l'espace dans lequel évolue le véhicule avec les caractéristiques suivantes : le point de départ possède un potentiel haut, et le point d'arrivée possède un potentiel bas, chaque obstacle ou contrainte est traduit par la présence de pics de potentiel. Cette approche permet d'appliquer des algorithmes efficaces pour la recherche du chemin (comme la descente de gradient). La configuration de l'espace peut cependant mener à la présence de minima locaux qui doivent alors être pris en compte de manière spécifique pour ne pas que le véhicule s'y retrouve bloqué.

Pour permettre une exploration plus stratégique, il est utile de procéder en deux temps : discrétiser l'espace sous forme d'un graphe puis explorer ce graphe pour trouver le chemin optimal.

Représentation de l'espace sous forme de graphe

La tessellation de Dirichlet (aussi appelée diagramme de Voronoï) décompose l'espace selon des frontières d'équidistance aux obstacles. Elle permet ainsi de représenter les voies les plus sûres à travers un ensemble d'obstacles comme illustré par la figure 11. Cependant, elle peut pénaliser la distance globale à parcourir pour atteindre la destination voulue [33].

Il est possible d'obtenir une granularité plus fine en créant le graphe à partir d'une décomposition de l'espace navigable en cellules, chaque cellule portant un attribut décrivant la présence d'un obstacle (avec un état libre ou occupé par exemple). La taille de ces cellules peut être variable ce qui permet d'obtenir un compromis entre nombre de cellules et finesse de représentation des obstacles.

Il est finalement possible d'échantillonner l'espace navigable avec un ensemble de point (nœuds) disposés arbitrairement (sous forme de grille par exemple). Les connections entre les nœuds (branches) sont des transitions autorisées pour le véhicule. Dans l'approche par graphe de visibilité chaque nœud possède une branche vers tous les nœuds *visibles*, c'est-à-dire atteignables en ligne droite en n'intersectant aucun des obstacles connus. La figure 12 montre un exemple de graphe de visibilité. Cette décomposition permet d'effectuer des transitions directes entre des nœuds qui n'appartiennent pas nécessairement à deux rangées successives.

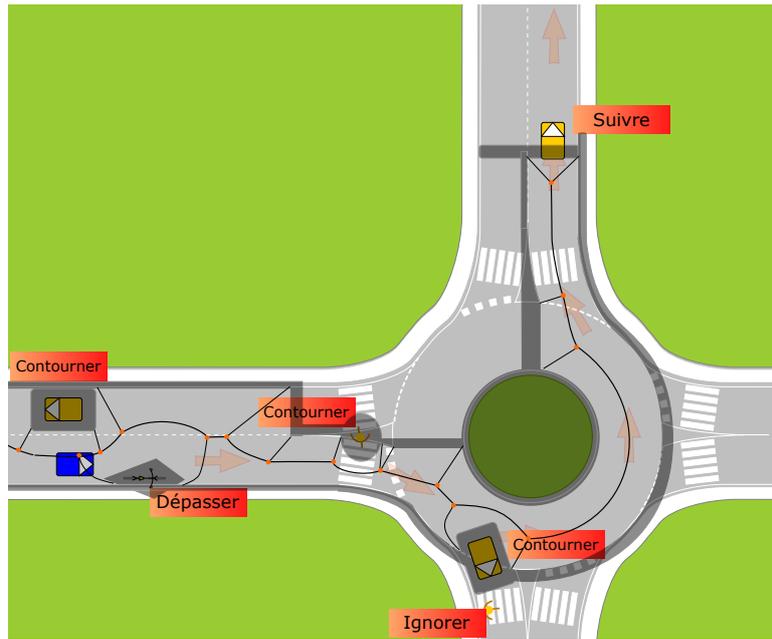


Figure 11 – Décomposition de Dirichlet (diagramme de Voronoï). Les limites de l'espace navigable sont indiquées par les zones noires.

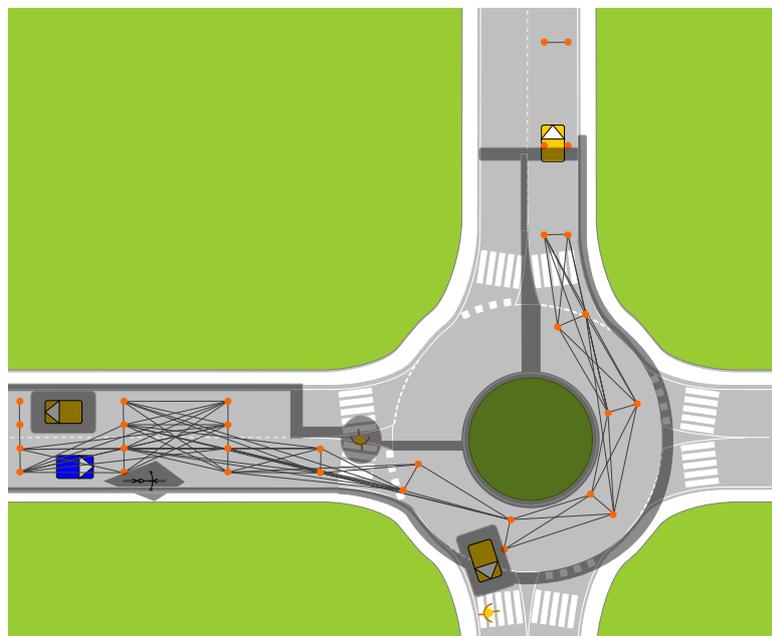


Figure 12 – Graphe de visibilité. Les nœuds (en orange) sont connectés par des branches (en noir) seulement si elles sont contenues dans l'espace navigable dont les bordures sont en noir.

Ces approches ont été présentées en se basant sur l'espace navigable en deux dimensions (coordonnées x et y) mais elles peuvent être appliquées plus généralement dans l'espace des configurations du mobile (incluant l'état complet du mobile à chaque nœud $[x \ y \ \theta \ v \ \dots]^T$). Cela modélise de manière plus fidèle l'accessibilité des nœuds mais complexifie les calculs nécessaires à la création du graphe.

Ces différentes approches permettent une première décomposition de l'espace en vue d'effectuer une recherche de chemin optimal telle que décrite dans le paragraphe suivant. Cependant, elles ne prennent pas en compte certaines contraintes inhérentes liées au véhicule autonome. Par exemple, les décompositions en graphes (par visibilité, décomposition de Dirichlet ou en cellule) assurent la connectivité des nœuds deux à deux mais pas la faisabilité des transitions entre les branches. En effet, le véhicule possède des limitations géométriques, cinématiques et dynamiques qui lui sont propres et qui peuvent rendre impossible la réalisation d'angles trop aigus entre deux branches par exemple. Ces contraintes peuvent être prises en compte dans l'algorithme de recherche ou dans le cadre d'un algorithme dédié.

Pour un véhicule autonome évoluant sur le réseau routier, il est important de considérer l'environnement à différentes échelles. Le véhicule doit rejoindre une destination possiblement très lointaine, gérer sa position sur la chaussée (changements de voies par exemple) à des distances modérées ainsi que gérer les obstacles dans son environnement proche. Il est donc souvent judicieux de décomposer la représentation de l'espace selon ces différentes échelles (de temps et/ou d'espace). Les exemples présentés dans ce paragraphe illustrent particulièrement la représentation de l'espace à petite et moyenne échelles. Pour un véhicule effectuant un grand trajet, le graphe pour la planification dite «stratégique» est souvent fourni par une carte routière de navigation. Chaque route est alors convertie en branche tandis que les lieux forment les nœuds du graphe.

Algorithmes de recherche

Le deuxième volet du calcul du chemin consiste à trouver la combinaison optimale de branches permettant de rallier un point de départ à un point d'arrivée au sein du graphe précédemment établi. Il s'agit donc d'explorer le graphe d'une part et d'évaluer les chemins candidats.

L'algorithme *Rapidly Exploring Random Tree* (RRT) [34] fait grandir un arbre représentant les chemins possibles. A chaque itération, un tirage aléatoire est réalisé pour choisir un nœud du graphe. Si la transition entre ce nœud et le nœud le plus proche de l'arbre est faisable, alors on l'ajoute à l'arbre existant. Le processus est répété si la destination n'est pas atteignable depuis le nouveau nœud de l'arbre. Il est possible de choisir des fonctions aléatoires qui privilégient une direction de recherche (cette direction de recherche est alors appelée biais heuristique) ou, à l'inverse, obtenir une couverture la plus homogène possible de l'espace. L'algorithme RRT est un exemple de méthode non déterministe car reposant fortement sur des tirages aléatoires. Le chemin résultat n'est pas garanti comme étant optimal et plusieurs exécutions de cet algorithme dans des conditions identiques ne fourniront pas le même chemin résultat.

L'algorithme génétique repose lui aussi sur des tirages aléatoires [35]. Il repose sur le principe darwinien de la sélection naturelle selon laquelle les individus les mieux adaptés perdurent. Un ensemble de solutions candidates (chromosomes) est choisi aléatoirement au sein du graphe construit précédemment. Un score est ensuite affecté à chacune d'elles sur la base d'un ensemble de critères (longueur, intersection avec des obstacles, faisabilité cinématique, ...). La troisième étape consiste à créer une nouvelle population de résultats (la seconde génération). Pour ce faire, un tirage aléatoire est effectué dans la première génération, la probabilité qu'un candidat de la première génération a d'être choisi étant proportionnelle à son score. La dernière étape consiste à introduire arbitrairement des mutations (modifications partielles) et des croisements (combinaison de deux chromosomes) au sein

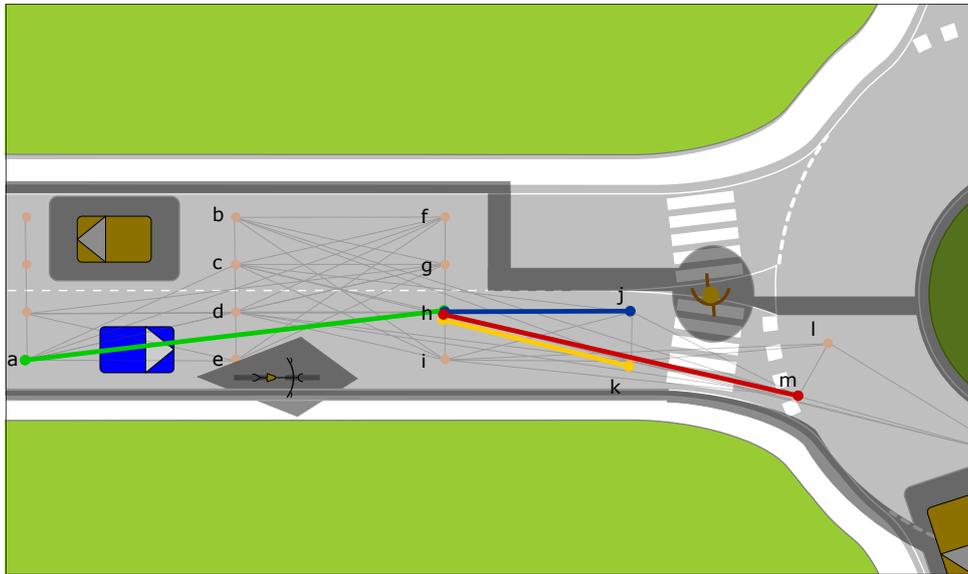


Figure 13 – Algorithme Dijkstra. Le véhicule est actuellement sur le segment $[ah]$. A la première itération, le point j est retenu. A la seconde itération, tout l'itinéraire ahk est plus court que tous ceux passant par j , l'exploration se poursuit donc via $[hk]$. De même à la troisième itération qui privilégie la branche $[hm]$.

de certaines solutions candidates. Les étapes de ré-échantillonnage et de mutation sont répétées un nombre fini de fois (générations) qu'on choisit suffisamment élevé pour espérer converger vers une population contenant des solutions proches de l'optimal. Cette approche nécessite un grand nombre de chromosomes pour couvrir l'ensemble des possibles et converger vers un optimal. L'évaluation du score par chromosome et les mutations représentent un grand nombre d'opérations à réaliser à chaque boucle de calcul mais peuvent être significativement parallélisées si le calculateur le permet (via l'usage de GPU par exemple).

Les algorithmes Dijkstra et A* ont pour principe d'explorer un graphe depuis un nœud de départ. L'exploration peut se faire sur l'entièreté du graphe ou s'interrompre lorsque le nœud de destination a été trouvé. Les branches possèdent un coût qui illustre la satisfaction des différentes contraintes que l'on se fixe pour le calcul de chemin (distance parcourue, non collision par exemple). Dijkstra comme A* explorent le graphe selon le chemin de moindre coût. Lorsque le coût cumulé du chemin en cours d'exploration dépasse celui d'un second chemin du graphe, ces algorithmes procèdent à l'exploration du second chemin. La particularité de l'algorithme A* est de comprendre une heuristique globale dans le calcul du coût des branches pour privilégier une direction de recherche.

Si les fonctions de coût sont positives, Dijkstra et A* permettent de trouver le chemin optimal du graphe, si la solution existe.

Nous développons ici la recherche du chemin optimal à l'aide de l'algorithme de Dijkstra dans le graphe de visibilité. Afin d'éviter de construire un chemin qui serait difficilement exécutable pour le véhicule d'un point de vue cinématique, il est possible de choisir une fonction de coût pour un nœud candidat n_{i+1} telle que :

$$f_c(n_{i+1}) = \begin{cases} \|\vec{n_i n_{i+1}}\| & \text{si } \angle(n_{i+1} n_i n_{i-1}) < \alpha_{max} \\ \infty & \text{sinon} \end{cases} \quad (6)$$

où α_{max} est l'angle maximal toléré entre la branche candidate et la branche parente.

La figure 13 illustre quelques étapes du processus. L'algorithme couvre progressivement l'espace navigable en développant la branche de moindre coût. On note que les branches $[hi]$, $[hg]$ et

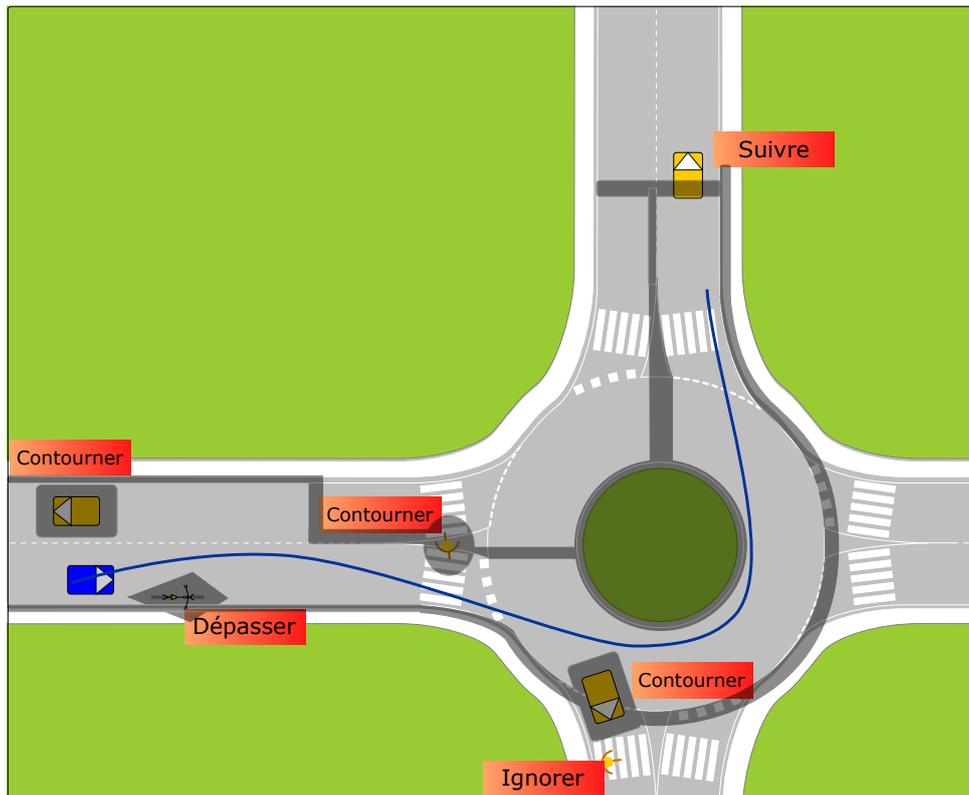


Figure 14 – Chemin (en bleu) obtenu après lissage du résultat obtenu par l’algorithme Dijkstra dans le graphe de visibilité.

$[j/k]$ ne sont ici pas retenues car considérées comme non faisables par le véhicule. Si l’échantillonnage spatial choisi lors de la création du graphe est trop grossier, il est nécessaire de lisser le résultat obtenu (avec des méthodes polynomiales ou B-spline par exemple) pour obtenir le chemin final comme illustré en figure 14.

5.2.3 Profil en vitesse le long du chemin de consigne

Un fois le chemin déterminé, il s’agit d’établir la temporalité que l’on veut y associer. Cela peut se formaliser en attribuant aux points du chemin des instants ou des vitesses de passage désirés. Le choix de l’une ou l’autre des approches peut se faire en fonction des attendus du système client, à savoir le contrôle du véhicule. Nous développons ici une méthodologie de détermination d’un profil de vitesse associé au chemin qui est généralement plus robuste aux erreurs de contrôle.

Le calcul du chemin précédemment développé garantit la faisabilité géométrique en fonction des dimensions du véhicule et de l’espace disponible pour naviguer. Le profil en vitesse, quant à lui, influence directement le confort ressenti à bord et la faisabilité dynamique pour le véhicule. L’usage du véhicule autonome au sein du trafic routier habituel, et particulièrement en zones urbaines et péri-urbaines, nécessite d’avoir un comportement sûr et réalisant un bon compromis entre le confort et l’efficacité. Le véhicule autonome peut alors être inclus dans le trafic sans en devenir un élément perturbateur. Les contraintes à prendre en compte dans le calcul du profil de vitesses peuvent être catégorisées comme suit :

- La libre circulation. Elle contient les contraintes de vitesse que l’on applique sans prise en compte des autres usagers de la route. La courbure des voies de circulation, la vitesse maximale autorisée, les points de prudence cartographiés (passages piétons, ralentisseurs...) sont compris dans cette catégorie. L’avantage de ces éléments est qu’ils sont invariants en fonction de la position courante du véhicule. Il peuvent donc être pris en compte une fois pour toutes.
- L’interaction avec les autres usagers de la route. A chaque instant, la manœuvre choisie et les

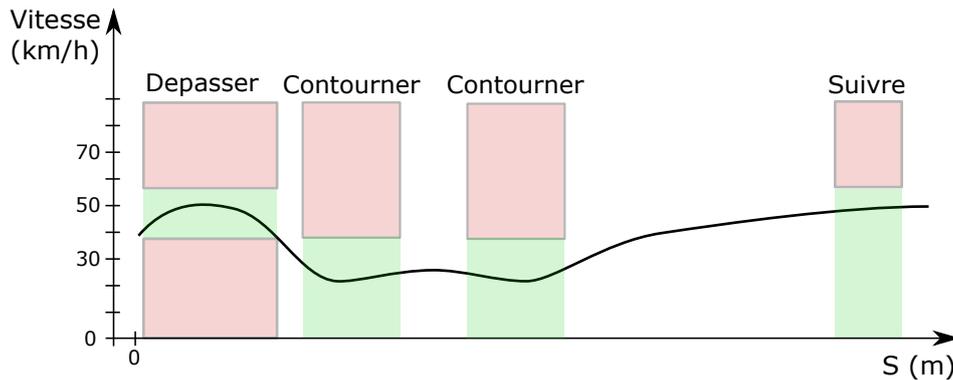


Figure 15 – Profil de vitesse associé au chemin calculé illustré en figure 14. L'axe horizontal indique l'abscisse curviligne le long du chemin (0 étant la position courante). Les contraintes induites par les manœuvres choisies apparaissent en rouge (valeur interdites) et en vert (valeurs permises).

états (position, vitesse, cap) mesurés des autres usagers de la route fournissent des contraintes de vitesse à prendre en compte. Par exemple lors d'un suivi de véhicule ou d'un arrêt pour un piéton.

- Les contraintes temporaires. Les panneaux à message variables peuvent imposer des vitesses maximales autorisées qui ne sont valables que pour une période de temps courte. Les feux tricolores imposent aussi, selon leur cycle, l'arrêt complet à certains instants.
- L'anticipation des limites de perception. Le module de décision peut finalement, appliquer une contrainte de vitesse supplémentaire sur la base de la connaissance de ses propres capacités de perception. Il est en effet préférable de ralentir lors du dépassement d'un bus à son arrêt du fait de l'impossibilité de percevoir d'éventuels piétons traversant devant le bus.

La figure 15 illustre ce que pourrait être le profil de vitesse pour la scène de conduite courante.

Il est fréquent qu'un véhicule soit amené à gérer une priorité à droite, à s'insérer dans une file ou bien à traverser un rond-point. Dans ce cas, on peut utiliser le concept de véhicule virtuel [36] comme illustré par la figure 16. Les véhicules en interaction à cause d'un carrefour en T par exemple déterminent le point intersection de leurs chemins respectifs (rond vert sur la figure). Chaque véhicule peut ramener ensuite virtuellement la position de l'autre sur son chemin. Plusieurs décisions peuvent en découler. Sur la figure 16, le véhicule rouge est prioritaire et le gris évalue son inter-distance avec lui pour évaluer si l'autre véhicule lui apporte une contrainte. Si maintenant le véhicule rouge est le véhicule autonome, en appliquant la méthode et si le gris a une vitesse du même ordre de grandeur, il peut décider qu'il a le temps de tourner à droite et que le véhicule gris ne lui apporte pas de contrainte. Ce genre de raisonnement avec des véhicules virtuels est très général et peut s'appliquer à de nombreux problèmes d'intersection.

5.3 Exécuter la trajectoire

Ce paragraphe traite de la réalisation du contrôle d'exécution de la trajectoire calculée précédemment. On suppose que cette trajectoire est faisable (notamment en ce qui concerne la contrainte de non holonomie due à l'essieu arrière non directionnel [s7755]). On présente l'approche classique qui découple le problème du contrôle latéral (angle au volant) qui réalise le suivi de chemin de celui du contrôle longitudinal qui utilise le profil en vitesse. Celui-ci effectue aussi le contrôle en position pour que le véhicule s'arrête à un endroit précis ou bien qu'il respecte une inter-distance avec un

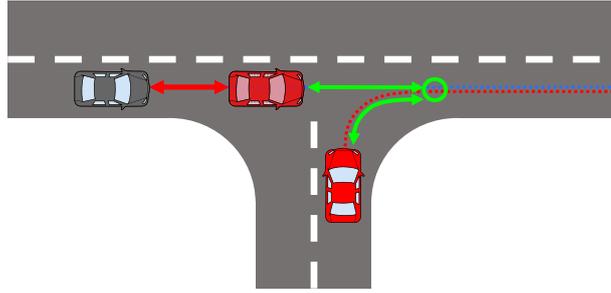


Figure 16 – Illustration du concept de véhicule virtuel. La voiture rouge a la priorité par rapport à la grise. En ramenant virtuellement la voiture rouge sur son chemin de consigne, la grise constate que l'autre voiture la précède et donc la laisse passer.

véhicule qui le précède par exemple. Il est possible d'aborder de façon conjointe le contrôle latéral et le contrôle longitudinal par exemple avec des approches de contrôle prédictif basé modèle [37]. On se limite dans ce paragraphe à une étude cinématique. Dans ce cadre, l'ordinateur contrôle :

- l'angle au volant qu'on suppose ici être l'angle δ de la roue avant virtuelle d'un modèle bicyclette conformément à la géométrie directionnelle d'Ackermann.
- les couples de freinage et de traction. Pour un véhicule à motorisation électrique ou hybride, le freinage peut être en partie réalisé par le ou les moteurs électriques liés aux roues pour recharger les batteries. En pratique, la partie bas-niveau du contrôleur doit être en mesure d'adresser une commande au système de freinage à friction si la force de freinage demandée l'exige.

5.3.1 Contrôle latéral

Considérons un véhicule de type « traction avant ». La commande est δ , la vitesse v de la roue avant étant régulée par ailleurs.

En notant L l'entraxe et θ l'orientation du véhicule, le modèle d'évolution cinématique 2D s'écrit :

$$\begin{cases} \dot{x} = v \cdot \cos \delta \cos \theta \\ \dot{y} = v \cdot \cos \delta \sin \theta \\ \dot{\theta} = \frac{v \sin \delta}{L} \end{cases} \quad (7)$$

Pour synthétiser la commande, il est intéressant de se placer dans un repère de Frenet [s7755]. On définit un vecteur tangent unitaire à la courbe orientée qui suit le chemin à suivre.

Supposons dans un premier temps que la ligne soit horizontale, la dimension en x n'importe plus. Donc, on peut supprimer la première équation :

$$\begin{cases} \dot{y} = v \cdot \cos \delta \sin \theta \\ \dot{\theta} = \frac{v \sin \delta}{L} \end{cases} \quad (8)$$

Supposons qu'on connaisse parfaitement l'état $X = [y \ \theta]^T$.

Une commande intéressante est facile à obtenir par linéarisation du modèle (8) pour des angles de braquage δ et d'écart angulaire θ petits. Au premier ordre, on a :

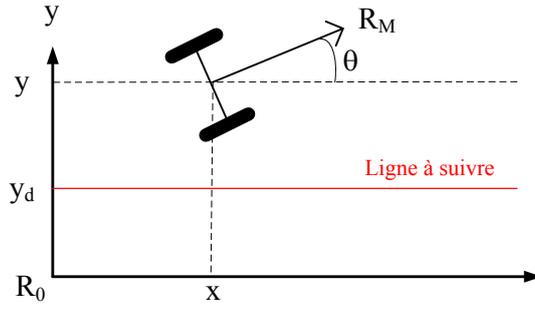


Figure 17 – Commande latérale le long d'un segment horizontal. L'entraxe L qui sépare les roues arrière des roues avant n'est pas représenté.

$$\begin{cases} \dot{y} = v.\theta \\ \dot{\theta} = \frac{v}{L}\delta \end{cases} \quad (9)$$

En dérivant la première équation par rapport au temps en supposant la vitesse constante, on obtient :

$$\ddot{y} = \frac{v^2}{L}\delta \quad (10)$$

Définissons une commande intermédiaire notée u :

$$u = \frac{v^2}{L}\delta \quad (11)$$

Notre système à contrôler devient maintenant

$$\ddot{y} = u \quad (12)$$

C'est un double intégrateur pour lequel une commande « proportionnelle-dérivée » est bien adaptée [38] :

$$u = \alpha_1 (y_d - y) + \alpha_2 (\dot{y}_d - \dot{y}) + \ddot{y}_d \quad (13)$$

où

- α_1 et α_2 sont les gains proportionnels et dérivés dont il faut trouver de bonnes valeurs,
- y_d , \dot{y}_d et \ddot{y}_d correspondent à la trajectoire désirée du centre du repère mobile (le point M).

Pour obtenir la dynamique du système en boucle fermée, on remplace u par sa loi de commande dans (12). En posant $e = y_d - y$, on a l'équation différentielle de l'erreur à réguler :

$$\ddot{e} + \alpha_2 \dot{e} + \alpha_1 e = 0 \quad (14)$$

L'équation différentielle (14) donne le polynôme en « s » après transformée de Laplace :

$$(s^2 + \alpha_2 s + \alpha_1) e = D(s).e = 0 \quad (15)$$

On choisit des pôles de sorte à avoir une erreur qui converge vers 0 (avec leur partie réelle négative) et de sorte à avoir un temps de réponse choisi et peu d'oscillations (par exemple, on peut se placer au régime critique pour avoir une réponse rapide sans dépassement).

Il ne reste plus qu'à retrouver l'entrée δ à appliquer. On reprend l'équation (11) ce qui donne :

$$\delta = \frac{L}{v^2} (\alpha_1 (y_d - y) + \alpha_2 (\dot{y}_d - \dot{y}) + \ddot{y}_d) \text{ et } v \neq 0 \quad (16)$$

Cette équation fait apparaître la dérivée de la sortie (c'est normal puisqu'on utilise une action dérivée dans la commande). Dériver un signal est toujours problématique à cause du bruit mais ici il n'y a pas besoin de le faire. En effet, l'équation (9) indique que $\dot{y} = v.\theta$ et donc

$$\delta = \frac{L}{v^2} (\alpha_1 (y_d - y) + \alpha_2 (\dot{y}_d - v.\theta) + \ddot{y}_d) \text{ et } v \neq 0 \quad (17)$$

Quand la consigne à suivre est parfaitement horizontale et ne varie pas dans le temps (comme c'est le cas ici puisqu'on fait le suivi d'un chemin en ligne droite), on a $\dot{y}_d = \ddot{y}_d = 0$ et donc :

$$\delta = \frac{L}{v^2} (\alpha_1 (y_d - y) + \alpha_2 (0 - v.\theta)) \text{ et } v \neq 0 \quad (18)$$

C'est un régulateur statique (car il n'a pas de vecteur d'état) qui est donc très simple à implémenter. Des résultats expérimentaux sont donnés lors de changements de voies sur autoroutes dans [39].

Pour aller plus loin

En pratique, le chemin n'est pas horizontal (aligné sur l'axe des abscisse du repère de Frenet). En utilisant une étape de map-matching telle que celle présentée précédemment en utilisant le chemin de consigne, on obtient l'écart e (signé) et la pente α du chemin. La variable e_d est une consigne de décalage latéral qu'on peut appliquer si besoin. Si le chemin de consigne contient aussi la courbure ζ (ce qui est très utile pour le confort de conduite), la loi de commande s'écrit comme suit :

$$\delta = \frac{L}{v^2} (\alpha_1.(e_d - e)) + \alpha_2.\frac{L}{v}.\text{submodpi2}(\alpha, \theta) + L.\zeta \text{ et } v \neq 0$$

On rappelle que la fonction « submodpi2 » calcule un écart angulaire entre $-\pi$ et π .

Ce genre de loi de commande cinématique suppose peu de glissement. Si c'est le cas parce que le véhicule roule vite ou bien si la chaussée est glissante, il faut utiliser des lois de commande plus performantes [40].

5.3.2 Contrôle longitudinal

Le contrôle longitudinal a pour objectif de suivre le profil de vitesse le long du chemin de consigne et de gérer de façon réactive les interactions avec les autres usagers de la route.

Contrôle selon le profil de vitesse

Une première stratégie de contrôle est de réguler la vitesse conformément au profil en vitesse fourni par le planificateur tel que défini dans le paragraphe 5.2.3.

Si on applique un régulateur proportionnel intégral de gains respectifs β_1 et β_2 , en notant v_d la consigne, on obtient la fonction de transfert suivante (s est la variable de Laplace) :

$$\frac{v}{v_d} = \frac{s\beta_1 + \beta_2}{(s^2 + \beta_1s + \beta_2)} \quad (19)$$

On note au dénominateur un zéro stable qui ne pose aucune difficulté tant que le profil de consigne ne présente pas de discontinuité. Cela montre l'importance d'avoir un planificateur de vitesse qui permet de maîtriser la dérivée du profil et ainsi de disposer d'un profil lisse.

De plus, un véhicule contrôlé par un tel régulateur a en général un temps de réponse de plusieurs secondes. Dans ce cas et en fonction du véhicule considéré, on choisit une consigne v_d par avance de phase, c'est-à-dire un point choisi dans le futur dans le profil calculé. Même si le profil est déterminé en abscisse curviligne le long du chemin de consigne, il est très simple d'extraire le bon échantillon en fonction de la vitesse du véhicule.

En pratique, on ajoute également une action dérivée grâce à une mesure de l'accélération longitudinale disponible dans tous les véhicules modernes. Il existe évidemment des régulateurs beaucoup plus performants qui intègrent le glissement des roues et les effets aérodynamiques.

Contrôle réactif avec interaction dans le couloir de navigation

L'usage d'un planificateur qui inclut l'interaction avec les autres usagers de la route tel que décrit précédemment peut induire des délais de traitement significatifs et ainsi retarder l'action. En effet, les cadences d'échantillonnage sont assez différentes : le planificateur fonctionne en général à 10 Hz alors que le contrôle est cadencé à une fréquence 10 fois plus élevée. Nous présentons ici une autre approche permettant de contrôler le mouvement longitudinal directement sur la base d'informations issues de la perception. Cette fois le planificateur de vitesse fournit simplement le profil de libre circulation le long du chemin.

Lorsque le véhicule doit s'arrêter (par exemple à un stop) ou bien lorsqu'il interagit avec un autre usager qui lui apporte une contrainte par rapport à sa vitesse nominale, une action en boucle fermée utilisant directement les données du système de perception est nécessaire. Il existe plusieurs façons d'aborder ce problème. On présente ci-dessous une méthode élémentaire (non optimale) qui agit sur le contrôle en vitesse.

On traite le cas d'un autre usager de la route (véhicule, cycliste, etc.) dans le couloir de navigation ou à proximité immédiate. On l'appelle en général le *Most Important Object (MIO)*. L'inter-distance à maintenir peut être définie comme suit :

$$d(t) = r + h.v(t) \quad (20)$$

où r est l'inter-distance à l'arrêt (3 m par exemple) et h correspond à un temps inter-véhicule (2 secondes dans le code de la route français). Grâce à la perception et à la connaissance de la longueur du véhicule, on mesure d le long du chemin de consigne. On en déduit la vitesse d'interaction due à la contrainte du MIO :

$$v_{Int}(t) = \frac{d(t) - r}{h} \quad (21)$$

La vitesse à appliquer en consigne du régulateur est donnée par :

$$v_d = \min(v_{lc}, v_{Int}) \quad (22)$$

où v_{lc} est la vitesse de libre circulation.

On peut améliorer grandement ce régulateur en utilisant aussi la vitesse relative quand celle-ci est directement mesurée (et non calculée en dérivant les position mesurées). C'est pour cette raison que les voitures disposant du régulateur ACC (*Adaptive Cruise Control*) sont équipées d'un radar hyperfréquence à effet Doppler. Les moyens de communication entre les véhicules (V2X) sont aussi une piste prometteuse pour enrichir les mesures disponibles et effectuer un contrôle longitudinal stable et réactif [41].

À retenir

- Une étape de décision au niveau des manœuvres permet de réduire l'espace de recherche pour la planification de chemin et d'avoir un véhicule dont le comportement est intelligible et explicable.
- L'approche à choisir pour le calcul du chemin dépend du type d'environnement dans lequel le mobile évolue. La faible complexité et la répétabilité de l'algorithme sont des éléments de choix importants.
- Le contrôle du véhicule s'effectue généralement en deux parties séparées : latéral et longitudinal. Cela permet de simplifier la synthèse et de régler leur dynamique indépendamment.
- Afin de réduire les temps de réaction, il est possible de transmettre directement des informations de perception au système de contrôle.

6 Conclusion et perspectives

Dans cet article, nous avons présenté une façon classique de décomposer la navigation autonome en sous problèmes. Pour que le véhicule puisse naviguer, ces différents systèmes doivent être intégrés dans une architecture plus globale. La sécurité étant fondamentale compte tenu des dégâts que peuvent engendrer ces véhicules, il faut que cette architecture intègre un moniteur qui vérifie en temps réel le bon fonctionnement des différents sous-systèmes. Ce moniteur doit aussi vérifier que la qualité des informations de navigation est adaptée à la tâche à réaliser. C'est la notion d'intégrité. Les systèmes les plus critiques doivent faire également l'objet de considérations liées à la sûreté de fonctionnement comme la tolérance aux fautes avec redondance par exemple. Comme nous l'avons évoqué, les informations externes reçues par des moyens de communication sans fil doivent être traitées avec une approche sécuritaire. Les problèmes de cyber-sécurité liés aux véhicules autonomes sont en effet nombreux et importants.

La navigation autonome sur route ouverte à la circulation publique doit également prendre en compte les interactions avec les autres acteurs du trafic. La compréhension du comportement et des intentions des autres agents de façon à prendre des décisions compréhensibles par les autres est un sujet d'une grande importance pour le déploiement des véhicules autonomes. Dans ce cadre, l'autonomie doit également considérer des principes d'interaction sociale. C'est un domaine de recherche interdisciplinaire émergent.

Pour finir, nous pensons que le conducteur doit continuer à jouer un rôle quand la tâche de navigation présente un niveau de risque élevé, c'est-à-dire qu'un accident grave peut se produire et ce, avec une probabilité d'apparition non négligeable. Dans ce cadre, les questions d'interaction Homme-Machine doivent faire l'objet de recherches actives.

7 Glossaire

Système autonome ; *Autonomous system*

Système capable de prendre seul des décisions et de les exécuter sans une intervention humaine explicite.

GNSS (*Global Navigation Satellite System*)

Système de localisation par satellites constitué d'un segment spatial (constellation de satellites), d'un segment terrestre (réseau de bases de contrôle et de commande) et d'un segment utilisateur (récepteurs en charge de déterminer leurs positions et le temps courant). Ce terme générique inclut les systèmes européen (Galileo), américain (GPS), russe (GLONASS) et chinois (Beidou).

Mise en correspondance avec une carte ; *Map-matching*

Processus consistant à associer une mesure de position à un point (ou une zone) du réseau routier décrit dans la carte numérique de navigation. Il est utilisé pour réduire l'incertitude de localisation et pour calculer des écarts avec des éléments géo-référencés.

Capteur extéroceptif ; *Exteroceptive sensor*

Catégorie de capteurs de perception permettant d'acquérir des informations à l'extérieur du véhicule (par opposition aux capteurs proprioceptifs). Elle regroupe par exemple les caméras, lidars, radars.

Lidar (*Light Detection and Ranging*)

Technologie de capteurs de perception actifs ayant pour principe d'émettre un signal laser et d'en mesurer le temps de vol (aller - retour) ainsi que l'intensité de retour. Elle permet donc de fournir une mesure directe de distance et de réflectivité des objets environnants.

Repère de Frenet ; *Frenet frame*

Repère associé à une courbe géométrique dont les vecteurs unitaires sont tangents et normaux en tous point de la courbe. Il est déterminé par map-matching.

Pour en savoir plus

Sources bibliographiques générales

- Handbook of Intelligent Vehicles [42]
- Darpa Challenge [43]
- Understanding GPS : principles and applications [44]

À lire également dans nos bases

BATTU Daniel. – Voiture connectée. [te7620]

KASSER Michel. – Topographie – Topométrie – Géodésie. [c5010]

LUCIDARME Philippe et SIMONIN Olivier. – Cartographie et localisation simultanées multirobots. [s7738]

FILLIAT David. – Cartographie et localisation simultanées en robotique mobile. [s7785]

EHRlich Jacques. – Technologies d'assistance à la conduite automobile. [se3772]

BENAMAR Faïz et GRAND Christophe. – Robotique mobile : conception, modélisation et commande. [s7755]

GIRARD Benoît et KHAMASSI Mehdi. – Coopération d'algorithmes d'apprentissage par renforcement multiples. [s7793]

Outils logiciels

Robot Operating System (ROS).

Normes et standards

- SAE <https://www.sae.org/>
- National Highway Traffic Safety Administration (NHTSA) <https://www.nhtsa.gov/>
- European Telecommunications Standards Institute (ETSI) <https://www.etsi.org/>

Références

- [1] J. ZIEGLER et al. « Making Bertha Drive - An Autonomous Journey on a Historic Route ». In : *IEEE Intelligent Transportation Systems Magazine* 6.2 (2014), p. 8-20.
- [2] SAE On-Road Automated Vehicle Standards COMMITTEE et al. « Taxonomy and definitions for terms related to on-road motor vehicle automated driving systems ». In : *SAE Standard J 3016* (2014), p. 1-16.
- [3] D. BÉTAILLE et R. TOLEDO-MOREO. « Creating Enhanced Maps for Lane-Level Vehicle Navigation ». In : *IEEE Transactions on Intelligent Transportation Systems* 11.4 (2010), p. 786-798.

- [4] R. BEHRENS et al. « Digitale Karten im Navigation Data Standard Format ». In : *Handbuch Fahrerassistenzsysteme : Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort*. Sous la dir. d'Hermann WINNER et al. Wiesbaden : Springer Fachmedien Wiesbaden, 2015, p. 513-523.
- [5] M. DUPUIS, M. STROBL et H. GREZLIKOWSKI. « OpenDRIVE 2010 and Beyond—Status and Future of the de facto Standard for the Description of Road Networks ». In : *Proc. of the Driving Simulation Conference Europe*. 2010, p. 231-242.
- [6] P. BENDER, J. ZIEGLER et C. STILLER. « Lanelets : Efficient map representation for autonomous driving ». In : *2014 IEEE Intelligent Vehicles Symposium Proceedings*. 2014, p. 420-425.
- [7] F. LI. « Lane-level vehicle localization with integrity monitoring for data aggregation ». Theses. Université de Technologie de Compiègne, déc. 2018.
- [8] I. MILLER, M. CAMPBELL et D. HUTTENLOCHER. « Map-aided localization in sparse global positioning system environments using vision and particle filtering ». In : *Journal of Field Robotics* 28.5 (2011), p. 619-643. DOI : 10.1002/rob.20395.
- [9] J. E. GUIVANT, F. R. MASSON et E. M. NEBOT. « Simultaneous localization and map building using natural features and absolute information ». In : *Robotics and Autonomous Systems* 40.2 (2002), p. 79-90.
- [10] Z. TAO et al. « Lane marking aided vehicle localization ». In : *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. 2013, p. 1509-1515.
- [11] S. J. JULIER, J. K. UHLMANN et H. F. DURRANT-WHYTE. « A new approach for filtering nonlinear systems ». In : *Proceedings of 1995 American Control Conference - ACC'95*. T. 3. 1995, 1628-1632 vol.3.
- [12] M. KAESS, A. RANGANATHAN et F. DELLAERT. « iSAM : Incremental Smoothing and Mapping ». In : *IEEE Transactions on Robotics* 24.6 (déc. 2008), p. 1365-1378.
- [13] G. BRESSON, Z. ALSAYED et S. JONCHERY. « Graph-based Map-Aided Localization using Cadastral Maps as Virtual Laser Scans ». In : *IEEE Intelligent Transportation Systems Conference (ITSC)*. 2019, p. 4074-4080.
- [14] E. HÉRY. « Localisation coopérative de véhicules autonomes communicants ». Theses. Université de Technologie de Compiègne, nov. 2019.
- [15] J. AL HAGE et al. « Localization Integrity for Intelligent Vehicles through Fault Detection and Position Error Characterization. » In : *IEEE Transactions on Intelligent Transportation Systems* (2020).
- [16] E. HÉRY et al. « Map-based Curvilinear Coordinates for Autonomous Vehicles ». In : *20th IEEE International Conference on Intelligent Transportation (ITSC 2017)*. Yokohama, Japan, oct. 2017, p. 1-7.
- [17] J. VAN BRUMMELEN et al. « Autonomous vehicle perception : The technology of today and tomorrow ». In : *Transportation Research Part C : Emerging Technologies* 89 (2018), p. 384-406.
- [18] R. DE CHARETTE et F. NASHASHIBI. « Real time visual traffic lights recognition based on Spot Light Detection and adaptive traffic lights templates ». In : *2009 IEEE Intelligent Vehicles Symposium*. 2009, p. 358-363.
- [19] P. XU et al. « Multimodal information fusion for urban scene understanding ». In : *Machine Vision and Applications* 27.3 (avr. 2016), p. 331-349.

- [20] C. MENDES, V. FRÉMONT et D. WOLF. « Exploiting fully convolutional neural networks for fast road detection ». In : *IEEE International Conference on Robotics and Automation*. 2016, p. 3174-3179.
- [21] W. ZHOU et al. « Automated Evaluation of Semantic Segmentation Robustness for Autonomous Driving ». In : *IEEE Transactions on Intelligent Transportation Systems* 21.5 (2020), p. 1951-1963.
- [22] C. YU, V. CHERFAOUI et P. BONNIFAIT. « An Evidential Sensor Model for Velodyne Scan Grids ». In : *13th International Conference on Control, Automation, Robotics and Vision*. Singapore, déc. 2014, p. 583-588.
- [23] A. LANG et al. *PointPillars : Fast Encoders for Object Detection from Point Clouds*. 2019. arXiv : 1812.05784.
- [24] T. NGUYEN et al. « A comparison of cellular vehicle-to-everything and dedicated short range communication ». In : *IEEE Vehicular Networking Conference*. 2017. doi : 10.1109/VNC.2017.8275618.
- [25] D. SUO et S. SARMA. « Real-time Trust-Building Schemes for Mitigating Malicious Behaviors in Connected and Automated Vehicles ». In : *IEEE Intelligent Transportation Systems Conference (ITSC)*. 2019.
- [26] E. BERNARDI et al. « High Integrity Lane-level Occupancy Estimation of Road Obstacles Through LiDAR and HD Map Data Fusion ». In : *Intelligent Vehicles Symposium*. Las Vegas, United States, oct. 2020.
- [27] M. BOJARSKI et al. « End to end learning for self-driving cars ». In : *arXiv preprint arXiv :1604.07316* (2016).
- [28] F. OLIEHOEK et C. AMATO. « The Decentralized POMDP Framework ». In : *A Concise Introduction to Decentralized POMDPs*. Cham : Springer International Publishing, 2016, p. 11-32.
- [29] C. HUBMANN et al. « Decision making for autonomous driving considering interaction and uncertain prediction of surrounding vehicles ». In : *2017 IEEE Intelligent Vehicles Symposium (IV)*. 2017, p. 1671-1678. doi : 10.1109/IVS.2017.7995949.
- [30] S. ULBRICH et M. MAURER. « Probabilistic online POMDP decision making for lane changes in fully automated driving ». In : *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. 2013, p. 2063-2067. doi : 10.1109/ITSC.2013.6728533.
- [31] D. GONZALEZ et al. « A Review of Motion Planning Techniques for Automated Vehicles ». In : *IEEE Transactions on Intelligent Transportation Systems* 17.4 (2016), p. 1135-1145.
- [32] Y. WANG et G. S. CHIRIKJIAN. « A new potential field method for robot path planning ». In : *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation*. T. 2. 2000, 977-982 vol.2.
- [33] S. GARRIDO et al. « Path Planning for Mobile Robot Navigation using Voronoi Diagram and Fast Marching ». In : *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2006, p. 2376-2381. doi : 10.1109/IROS.2006.282649.
- [34] Y. BESTAOUI SEBBANE. *Planning and Decision Making for Aerial Robots*. T. 71. Springer, 2014.
- [35] CW. AHN. *Advances in evolutionary algorithms*. Springer, 2006.
- [36] S. MASI, P. XU et P. BONNIFAIT. « Adapting the Virtual Platooning Concept to Roundabout Crossing ». In : *IEEE Intelligent Vehicles Symposium (IV 2018)*. Changshu, China, juin 2018, p. 1366-1372.

- [37] R. ATTIA et al. « Reference generation and control strategy for automated vehicle guidance ». In : *IEEE Intelligent Vehicles Symposium*. 2012, p. 389-394.
- [38] L. JAULIN. *La robotique mobile ; cours et exercices*. ISTE editions, 2015.
- [39] P. XU et al. « System Architecture of a Driverless Electric Car in the Grand Cooperative Driving Challenge ». In : *IEEE Intelligent Transportation Systems Magazine* 10.1 (2018), p. 47-59.
- [40] A. CHEBLY, R. TALJ et A. CHARARA. « Coupled longitudinal/lateral controllers for autonomous vehicles navigation, with experimental validation ». In : *Control Engineering Practice* 88 (juil. 2019), p. 79-96.
- [41] S. E. SHLADOVER et al. « Using cooperative adaptive cruise control (CACC) to form high-performance vehicle streams ». In : (2014).
- [42] A. ESKANDARIAN. *Handbook of intelligent vehicles*. T. 2. Springer, 2012.
- [43] M. BUEHLER, K. IAGNEMMA et S. SINGH. *The DARPA urban challenge : autonomous vehicles in city traffic*. T. 56. springer, 2009.
- [44] Elliott KAPLAN et Christopher HEGARTY. *Understanding GPS : principles and applications*. Artech house, 2005.