
Gestion spatiotemporelle des observations sur la conduite en situation naturelle¹

Iulian Sandu Popa* — **Ahmed Kharrat *** — **Karine Zeitouni*** — **Francis Dupin**** — **Jacques Ehrlich** — **Guillaume Saint Pierre ****

* *Laboratoire PRISM*

45, avenue des Etats-Unis - 78035 Versailles

{Iulian.Sandu-Popa,Ahmed.Kharrat,Karine.Zeitouni}@prism.uvsq.fr

<http://www.prism.uvsq.fr>

** *LIVIC, Laboratoire sur les Interactions Véhicules-Infrastructure-Conducteurs*

Bâtiment 824 -14, route de la Minière - 78000 Versailles

Francis.Dupin@inrets.fr, {Guillaume.Saintpierre, Jacques.Ehrlich}@lcpc.fr

<http://www.inrets.fr/ur/livic/>

RÉSUMÉ. Cet article porte sur le problème de l'exploitation des données de capteurs embarqués dans les véhicules et collectées en situation de conduite naturelle (dite *naturalistic driving*). Plusieurs applications dans le domaine des systèmes de transports intelligents requièrent des analyses complexes de telles données. Ces analyses doivent tenir compte de la localisation spatiale et de la géométrie de la route. Afin de faire face aux limites des solutions actuelles, nous proposons une extension de système de gestion de bases de données, basée sur un modèle spécifique d'objets mobiles intégrant des capteurs, et qui permet de répondre à la grande partie des besoins d'applications du même type.

ABSTRACT. This paper addresses the problem of the usage of the in-vehicle sensor data collected in *naturalistic driving* conditions. Many applications in the intelligent transportation system research area require complex analysis of such data, taking account of the spatial localisation and the road network topology. An extended database management system using specific model for moving objects and sensor data, and fulfilling most of the application requirements, is presented.

MOTS-CLÉS : Collecte de données sur la conduite naturelle, analyse du comportement des conducteurs, évaluation de l'impact de l'assistance à la conduite, bases d'objets mobiles, modélisation des données, intégration et chargement de données, requêtes spatiotemporelle.

KEYWORDS: *Naturalistic driving data collection, analysis of drivers' behaviours, driving assistance impact assessment, moving object databases, data modelling, data integration and loading, spatiotemporal queries.*

¹ Ce projet est partiellement soutenu par le financement d'une allocation doctorale de la Région Ile-de-France.

1. Introduction

Les défaillances du conducteur étant à l'origine de la majorité des accidents de la route, la conception de systèmes d'aide à la conduite est devenue un enjeu de recherche majeur. Cependant, pour être pertinents et efficaces, ces systèmes doivent être évalués en termes de fiabilité, de robustesse, d'adaptabilité et d'acceptabilité sociale. Il est donc important d'avoir à disposition des connaissances précises et exhaustives concernant la conduite en situation naturelle, ainsi que des informations sur les situations de presque-accident (situations de conduites risquées où un accident a peut-être été évité). Excepté les données issues d'expériences sur simulateurs (biaisées par l'environnement imparfait de la simulation), nous avons actuellement très peu d'informations à disposition pouvant permettre de comprendre le comportement d'un conducteur en situation de conduite réelle, et par là même d'évaluer l'impact d'un système d'aide à la conduite (conduite avec et sans système).

Depuis quelques années le concept anglo-saxon de « naturalistic driving » permet d'avancer sur cette voie. Le principe est de collecter des données en situation de conduite naturelle sur une grande échelle et sur une période de temps significative, ce qui devient possible à moindre coût grâce aux équipements des véhicules modernes. En effet, les multiples capteurs installés sur les véhicules sont accessibles via une interface (Bus CAN) sur laquelle il est possible de brancher un boîtier enregistreur de données. La collecte de données à large échelle devient donc aisée, mais cela n'est pas sans poser de problèmes de gestion des grandes bases de données obtenues. En effet, la quantité d'information recueillie est énorme et les données de conduite ont un caractère complexe car elles sont associées à un objet en mouvement. Elles sont géo-référencées grâce au GPS, horodatées, et peuvent être projetées sur une carte du réseau routier munie d'une topologie. En outre, des corrélations évidentes existent entre les données de conduite et les caractéristiques de la route (rayon de courbure, vitesse légale, etc.) qui conditionnent la vitesse, le freinage et le braquage du volant par exemple. De même, des relevés portant sur des portions de route communes, même asynchrones, sont en général similaires. Seulement, le comportement général des conducteurs et surtout les écarts entre des comportements observés et un comportement théorique ou moyen restent méconnus aujourd'hui.

Bien que les données de mesures soient assez riches, la plupart des analyses et des requêtes effectuées jusqu'à maintenant ne considèrent que le paramètre « temps » et modélisent les données routières par des séries temporelles. Or, la situation de conduite mesurée par des données de type « naturalistic driving » se doit d'être analysée plus finement et intégrer l'aspect spatial en parallèle de l'aspect temporel. Si les Systèmes d'Information Géographique (SIG) et certains SGBD du marché gèrent bien les données spatiales telles que les cartes et les points GPS, ils ne suffisent pas à gérer et à analyser des mesures relatives à un objet en mouvement. Les problématiques spécifiques au domaine des transports (accès aux profils de vitesse, comparaison des trajets au cours du temps ou suite à des aménagements, etc.) nécessitent une vision spatiale que les solutions existantes de gestion de données ne sont pas en mesure de capturer. C'est ce qui a motivé le travail décrit

dans cet article, visant à développer un système de gestion de base de données comprenant un moteur de requêtes adapté à ces problématiques. Ce système a pour objectif de gérer aussi bien des trajectoires spatio-temporelles que des mesures variant dans le temps et dans l'espace. Il vise également à offrir un langage de requêtes permettant de les analyser simplement et efficacement. Ce travail est effectué en collaboration entre le LIVIC (Laboratoire de recherche sur les Interactions Véhicules-Infrastructure-Conducteur, rattaché à l'INRETS et au LCPC-France) et l'équipe PRISM (Laboratoire de recherche en informatique de l'Université de Versailles Saint Quentin).

La section suivante revient sur la problématique et l'illustre à travers des scénarios d'application. La section 3 décrit l'acquisition des données par un dispositif d'enregistrement embarqué. Ensuite, la section 4 décrit l'architecture globale du système comprenant l'acquisition, la fusion et le serveur de données, avant de se focaliser sur le serveur de données dans la section 5. Celle-ci décrit le modèle et situe les contributions par rapport à l'état de l'art, avant d'illustrer les possibilités du système proposé en revisitant les scénarios d'application de la section 2. En conclusion, nous rappelons les contributions et donnons quelques perspectives.

2. Illustration de la problématique

Les exemples d'applications présentés dans cette section sont une illustration des situations que rencontrent régulièrement les chercheurs du domaine des transports. Ces quelques détails ne sont bien entendu pas exhaustifs mais présentent de façon simple les retombées potentielles du travail présenté dans ce papier, ainsi que leurs implications pratiques pour la communauté des transports. Le contexte est celui d'une base de données rassemblant des informations complètes (quelques dizaines de variables : position GPS, mesures de vitesse, accélération, appui pédales, freinages, angle du volant, etc.) sur un grand nombre de trajets (plusieurs centaines), et pour un grand nombre de véhicules (plusieurs dizaines), le tout obtenu dans une région spécifique. A terme, ce type d'informations sera valable sur l'ensemble du territoire national, voire Européen, et les exemples qui suivent feront alors la démonstration de l'utilité et de l'efficacité de l'approche développée dans ce papier.

1- Analyse de la vitesse réglementaire : Dans le domaine des transports, la plupart des études sur la vitesse nécessitent l'usage de profils de vitesses qui sont généralement modélisés à l'aide des séries temporelles. L'approche développée ici permet de considérer les profils de vitesses comme des fonctions pouvant dépendre du temps, mais aussi de la position. Le modèle de données utilisé rend la manipulation de ce genre d'objet très simple. Par exemple, il devient possible d'accéder très facilement aux profils de vitesses des véhicules ayant relié un point A à un point B, dans un sens ou dans l'autre. De plus, si l'on dispose d'une base de données de vitesses légales sur le réseau routier considéré, la comparaison avec les profils de vitesses pourra alors se faire simplement. On imagine sans mal les multiples applications d'une telle évolution.

Pour illustrer ce qui vient d'être dit, considérons la question suivante :

Question 1 : « Disposant de données des vitesses réglementaires sur une section déterminée, retourner le pourcentage de véhicules dont la vitesse instantanée est supérieure de 30km/h à la vitesse autorisée pour une section de route donnée ».

2- Etude d'impact de mise en place d'un radar fixe : Le réseau routier Français a été récemment équipé de plusieurs centaines de radars fixes destinés à contrôler automatiquement la vitesse des véhicules. L'impact de telles installations en termes de réduction de la vitesse et donc du risque d'accident est bien connu (études en bord de route, réduction générale de la vitesse), mais en termes d'impacts sur le comportement des conducteurs, cela est plus difficile à mettre en œuvre. Il peut être par exemple très intéressant d'étudier les changements d'itinéraires suite à la mise en place d'un radar. Cela peut être exprimé au moins de deux manières différentes, dont la première est :

Question 2 : « Etant donné une portion de route, obtenir le taux d'occupation avant/après l'installation du radar »

Quant à la seconde, elle peut être :

Question 3 : « Sur une période de temps donnée, étant donné un point d'origine et une destination, obtenir l'ensemble des trajets reliant ces deux points avant/après l'installation du radar »

3- Analyse du comportement de conduite : L'analyse des comportements du conducteur est au cœur de l'étude de l'impact des systèmes d'aide à la conduite. Cela peut être entrepris en observant l'évolution de certains indicateurs pertinents pour un trajet donné : nombre de changements de vitesse par km, pourcentage de temps passé au-dessus de la vitesse légale, vitesse moyenne ou profil de vitesse, etc. L'obtention de ces indicateurs est facilitée par le travail décrit dans cet article.

Par exemple, une étude de risque portant sur une route précise peut se faire en utilisant le 85^{ème} percentile de la vitesse instantanée. En un point donné, il s'agit de la vitesse en dessous de laquelle roulent 85% des véhicules. Cette valeur instantanée calculable en tout point du parcours correspond à la notion de « vitesse pratiquée » et est reliée à la vitesse légale (chaque route incite à rouler à une certaine allure et chaque route est différente). Un exemple d'information utile dans ce cadre est le suivant :

Question 4 : « Etant donné une route particulière, obtenir le profil de vitesse de la « vitesse pratiquée » avant/après l'installation d'un radar automatique »

4- Relation entre vitesse et infrastructure : Le problème évoqué ici est celui de l'évaluation de l'impact de l'infrastructure sur la vitesse pratiquée, relié au fait de repérer des zones dangereuses (points noirs) à partir des informations recueillies au sein des véhicules. Une étude statistique rigoureuse de ces aspects nécessite une bonne représentativité statistique (un grand nombre de conducteurs), et une comparaison de plusieurs sites. Ces conditions sont difficiles à remplir et les études de type « naturalistic driving » peuvent apporter un éclairage nouveau vu la taille et la durée de ces expérimentations. L'étude des virages, par exemple, est un domaine bien connu qu'il devient plus aisé de traiter en utilisant l'approche développée dans ce papier. Il devient en effet possible de collecter des données sur des types de virages équivalents (une courbure entre 450 et 500m par exemple), de les filtrer pour

récupérer les situations de trafic libre, et d'étudier la vitesse pratiquée en fonction des paramètres géométriques du virage. Par exemple, on souhaite répondre à la demande suivante :

Question 5 : « Obtenir tous les profils de vitesse des véhicules en situation de trafic libre uniquement sur les virages ayant une courbure comprise entre 450 et 500m ».

3. Processus d'acquisition des données : Le DIRCO

Le DIRCO (Dispositif de Recueil de données Comportementales) a été conçu pour satisfaire le besoin d'un outil facile à installer sur un véhicule et qui permette d'enregistrer les paramètres de la conduite. L'idée est de pouvoir instrumenter une flotte de véhicules en un temps très court et sans apporter de modification à ces véhicules qu'il suffit alors de prêter le temps de l'expérimentation.

Le DIRCO est constitué de capteurs et d'un coeur pour l'enregistrement et le pré-traitement des données. Le coeur est conçu autour d'un PC embarqué de la taille d'un gros livre sur lequel est installé un système d'exploitation LINUX dans sa distribution Xenomai, laquelle apporte le bénéfice d'un noyau temps réel. Il permet d'enregistrer les données qui transitent sur le bus CAN du véhicule - bus de données qui équipe la plupart des véhicules récents -, d'enregistrer celles provenant d'une centrale inertielle qui fait partie du DIRCO, de connecter un GPS, ainsi qu'une Webcam. Il y a enfin la possibilité d'ajouter des capteurs optionnels à la condition qu'ils transmettent leurs données au format CAN-CANOpen. Le DIRCO intègre en effet une pile applicative au standard CANOpen-CanFestival (<http://canfestival.org>), dont les développements ont été repris et maintenus plusieurs années par le LIVIC, jusqu'à l'obtention d'un produit industrialisable.

Les performances du DIRCO sont les suivantes : Enregistrement des données, après prétraitement de conversion d'unité, sur mémoire flash - insensible aux vibrations - jusqu'à une fréquence de 100 Hertz. Datation de chaque donnée, avec une précision limitée par la résolution de la datation qui est de 1 milliseconde. Possibilité de limiter la masse de la récolte de données en faisant varier la fréquence d'échantillonnage en fonctions des mesures, avec la possibilité d'enregistrer rétroactivement à pleine cadence des données jusqu'à deux secondes. Cette fonctionnalité est par exemple intéressante lorsqu'on veut enregistrer avec la meilleure précision toutes les données qui ont précédé un freinage brusque.

4. Architecture globale

L'architecture fonctionnelle du système proposé est schématisée dans la figure 1 ci-dessous. Elle se compose d'une partie pour l'acquisition de données et l'accumulation des fichiers log brutes dans une base de données centrale intermédiaire. A partir des traces GPS, des données du bus CAN et d'autres capteurs additionnels, et en tenant compte d'un référentiel géographique (base de données routière), une étape de nettoyage, de fusion et de *map-matching* (Brakatsoulas et al., 2005) et de transformation est opérée sur les données brutes avant de les charger par

lots dans la base du serveur de données spatiotemporel. Dès lors, l'accès et la manipulation se feront simplement via le langage SQL intégrant l'ensemble des opérations sur les nouveaux types. Des services applicatifs tels que la visualisation cartographique ou l'analyse statistique seront développés au-dessus du serveur de données. Différentes applications, telles qu'un portail web, utiliseront ces services ou accéderont au serveur de données via des interfaces standards comme JDBC.

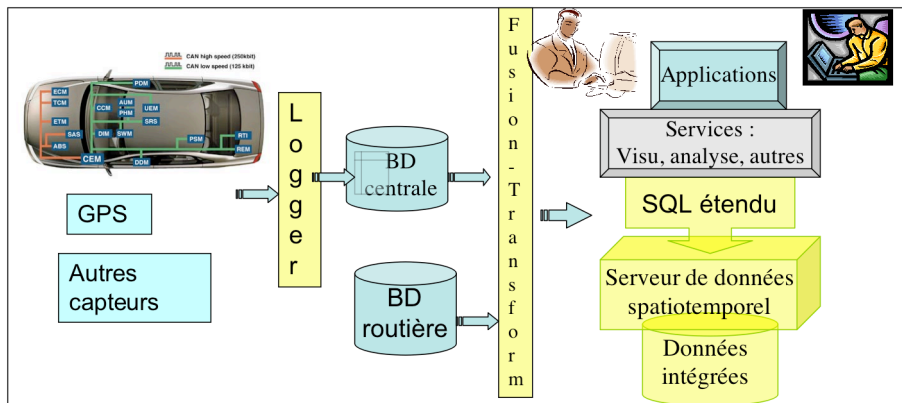


Figure 1. Architecture globale du système

5. Serveur de données

5.1. Etat de l'art

Les travaux les plus proches sont ceux sur les bases objets mobiles et dont la plupart traitent de la modélisation, de l'interrogation et des méthodes d'accès (Wolfson 2002), (Güting et Schneider, 2005). On y distingue généralement le cas des objets mobiles libres (Grumbach et al., 2003) de celui des objets mobiles contraints par un réseau comme le réseau de transport, car le rattachement au réseau mène à une représentation et une manipulation spécifiques. Orthogonalement, on distingue la gestion d'objets mobiles historiques (c'est-à-dire des trajectoires) de celle des objets mobiles courants. Ces derniers sont mis à jour fréquemment et en temps réel et mènent à des requêtes continues² ou prédictives³, à l'inverse des trajectoires qui sont plutôt statiques et dont les requêtes visent leur analyse.

Nous nous intéressons aux bases d'objets mobiles historiques contraints par le réseau. La proposition la plus complète pour modéliser les objets mobiles historiques est, à notre avis, celle issue des travaux de Güting (Güting et al., 2000, 2006). En effet, elle couvre la modélisation abstraite, le langage et l'implémentation. Comme nous le verrons plus loin, notre proposition se base sur ce modèle et l'étend

² Requêtes dont le résultat est calculé en temps réel selon l'évolution de la position des objets.

³ Requêtes basées sur l'estimation dans un future proche de la position des objets.

avec les données spécifiques aux capteurs mobiles. Par conséquent, nous résumons ce modèle dans le reste de cette section.

Güting et al. proposent une algèbre définie par un ensemble de types spécifiques et une collection d'opérations sur ces types. Ces types sont : des types scalaires (BASE), des types spatiaux 2D (SPATIAL), des types spatiaux liés à un réseau (GRAPH), des types variant dans le temps (TEMPOREL). Des exemples de types sont : *real*, *point* (position absolue en 2D), *gpoint* (position relative au réseau), *gline* (ligne relative au réseau), *moving (real)* (valeur temporelle), *moving (point)* (position 2D variant dans le temps), et *moving (gpoint)* (position relative au réseau variant dans le temps). La méthode *moving* est en fait un constructeur de type TEMPOREL. $\{\perp\}$ représente la valeur nulle (non définie).

Le type GRAPH dépend du réseau sous-jacent. Fondamentalement, le modèle proposé définit un réseau comme un ensemble de routes et de jonctions entre ces routes. Un emplacement dans le réseau est une position relative sur une route. Il est décrit par l'identifiant de la route, un nombre réel donnant la position relative et un côté. Ceci est directement lié au concept de la référence linéaire largement utilisée dans les applications en transport et dans des systèmes comme Oracle (Oracle, 2005). C'est ainsi que les types *gpoint* et *gline* sont représentés. Enfin, à partir des types de BASE, SPATIAL et GRAPH, les types temporels correspondants sont dérivés en utilisant le constructeur *moving*. Les types temporels sont des fonctions ou des ensembles infinis de paires (instant, valeur). En effet, le temps ici est isomorphe à l'ensemble des réels.

Les opérations sont alors définies sur les types de données précités. Pour éviter la prolifération des opérations, le même opérateur s'applique en s'adaptant sur plusieurs types. Un ensemble d'opérations non-temporelles est d'abord défini, puis, un processus dit de *lifting* permet de les adapter aux types temporels correspondants. Ainsi, toutes les opérations non-temporelles sont rendues disponibles sur les types temporels. Enfin, des opérations spécifiques sont ajoutées pour gérer les types temporels. Dans le contexte des trajectoires contraintes par le réseau (Güting et al., 2006), certaines opérations, comme la distance par la route, ont été revues sur les types ajoutés (i.e., *gpoint* et *gline*). De nouvelles classes d'opérations sont également ajoutées pour analyser l'interaction entre l'espace du réseau et l'espace 2D, ainsi que des opérations spécifiques au réseau comme le parcours de graphe.

Toutefois, aucun des travaux précédents ne permet de répondre à la problématique générale de gérer des objets mobiles munis de mesures, comme c'est le cas pour le type d'application visé.

5.2. Modèle proposé

Tout comme le modèle algébrique de (Güting et al., 2006) décrit ci-dessous, notre modèle intègre un type spatio-temporel pour modéliser la trajectoire de l'objet, ainsi qu'un type temporel pour modéliser les données générées par les capteurs, appelées *données de capteurs mobiles* ou *mesures* le long de ce papier. Le type

temporel correspond à une fonction du temps vers les types de base (*real*, *int*). Il exprime en effet la variabilité, du point de vue temporel, des mesures de capteurs.

Seulement, le point de vue temporel seul ne suffit pas pour modéliser *les données de capteurs mobiles*, car les mesures sont souvent étroitement liées à l'espace. Pour être complet, le modèle de données doit décrire en plus de l'évolution *temporelle*, l'évolution *spatiale* des *mesures*. A cet effet, nous étendons le modèle de (Güting *et al.*, 2006). Nous introduisons une nouvelle catégorie de types que l'on appelle *GVARIANT*, analogue à *TEMPORAL*, mais pour les variations dans l'espace réseau (GRAPH). De manière analogue, nous avons spécifié une catégorie de types pour les capteurs d'objets mobiles non contraints, mais cet article se focalise sur la mobilité dans un réseau. Nous proposons un nouveau constructeur de type *gmoving* :

$$\boxed{BASE \rightarrow GVARIANT} \quad \boxed{\underline{gmoving}}$$

La définition de ce constructeur est donnée ci-dessous en utilisant les notations de (Güting *et al.*, 2000) :

Définition 1: Soit α un type de *BASE* ayant le domaine A_α . Alors le domaine de définition pour $gmoving(\alpha)$ est défini comme suit: $A_{gmoving(\alpha)} = \{f \mid f: \bar{A}_{gpoint} \rightarrow \bar{A}_\alpha$ est une fonction partielle $\wedge \Gamma(f)$ est fini $\}$, où $\bar{A}_\beta = A_\beta \setminus \{\perp\}$.

Chaque valeur f du domaine de $gmoving(\alpha)$ est une fonction décrivant l'évolution dans l'espace (réseau) à une valeur de *BASE*. La condition " $\Gamma(f)$ est fini" indique que f est composée seulement d'un nombre fini de composants continus. Néanmoins, les constructeurs de type *gmoving* décrivent bien une infinité de paires (position, valeur), où la position est un *gpoint*. L'avantage de cette représentation continue est de s'affranchir de toute granularité de l'espace.

Dans (Güting *et al.*, 2000, 2006), les auteurs définissent un ensemble riche d'opérations que ce soit pour les objets mobiles libres ou pour ceux contraints par un réseau. La richesse de cette algèbre a motivé son utilisation comme point de départ. En introduisant des nouveaux types, nous devons analyser (i) comment ils influencent les opérations existantes et (ii) si de nouvelles classes d'opérations spécifiques sont nécessaires. Dans la suite de cette section, nous présentons d'une façon générale l'approche utilisée pour intégrer le nouveau type dans le système d'opérations. Ensuite, nous récrivons les requêtes ci-dessus dans le dialecte SQL d'Oracle et expliquons les opérations utilisées.

Premièrement, l'ensemble d'opérations non-temporelles devient des opérations non-gvariant non-temporelles dans le système étendu. Elles ont été rendues uniformément disponibles aux types temporels par un processus du *lifting*. Pour une opération ayant la signature $\alpha_1 \times \alpha_2 \times \dots \times \alpha_n \rightarrow \beta$, par un *lifting temporel*, l'opération est aussi disponible pour des signatures $\alpha'_1 \times \alpha'_2 \times \dots \times \alpha'_n \rightarrow moving(\beta)$, où $\alpha'_i \in \{\alpha_i, moving(\alpha_i)\}$. Chacun des arguments peut devenir dépendant du temps ce qui rend le résultat dépendant du temps également. Le système étendu inclut aussi les types dépendants de l'espace. Par conséquent, nous proposons un *lifting spatial* pour

les opérations non-gvariant non-temporelles. L'opération sera également disponible pour des signatures $\alpha'_1 \times \alpha'_2 \times \dots \times \alpha'_n \rightarrow moving(\beta)$, où $\alpha'_i \in \{\alpha_i, moving(\alpha_i)\}$.

Deuxièmement, nous avons défini de nouvelles opérations sur les types *GVARIANT*, de manière analogue à celles définies sur les types temporels, mais sur la vue spatiale. Par exemple **greaterThan** permet de retourner la portion du réseau où une mesure dépasse une autre mesure. Le résultat est de type *moving (bool)*. D'autres opérations ont été définies entre les types introduits et le réseau.

Troisièmement, d'autres classes d'opérations sont apportées par le nouveau type d'application. Nous proposons ainsi deux classes de telles opérations. Dans le cadre des travaux de Güting et al., les opérations algébriques '+', '-', '.' et '/' pour les types de *BASE* ne sont pas discutées du fait que la proposition était centrée sur les types spatio-temporels. Nous incluons les opérations algébriques de base dans les opérations non-gvariant non-temporelles. Par conséquent, elles deviennent sujettes au *lifting* temporel et spatial. Nous utilisons des fonctions nommées, c'est-à-dire **sum**, **sub**, **mul** et **div**, comme pour toutes les opérations définies. Ces opérations sont très utiles pour la comparaison numérique des fonctions modélisant les données capteur. Par exemple, elles aident à trouver la différence entre les vitesses pratiquées de deux objets mobiles pour une partie commune de la trajectoire ; ou, la différence entre les vitesses des objets mobiles et la vitesse légale.

La dernière classe des opérations adresse la même catégorie de types, c'est-à-dire des types *TEMPOREL* et *GVARIANT*. Il s'agit des agrégats sur des fonctions : **mean**, **min**, **max** et **no_transitions**. Chacune de ces opérations prend en entrée une fonction du temps ou de l'espace et retourne une valeur réelle résultat de l'agrégat correspondant. Leur intérêt est de permettre de calculer une moyenne ou une valeur extrême pour n'importe quelle mesure, et ce sur un intervalle temporel ou spatial.

Tous ces types et ces opérations peuvent être aisément implémentés comme une extension d'un SGBD comme Oracle (Oracle, 2005). Il devient ensuite possible de les exploiter à travers le langage standard SQL. Il reste ensuite à résoudre les problèmes d'optimisation de requêtes pouvant se poser. C'est exactement le schéma que nous avons suivi pour mettre en œuvre le serveur de données pour des objets et des capteurs mobiles.

Plutôt que de donner tous les détails du modèle et étant donné l'espace limité, nous préférons illustrer l'utilisation de quelques opérations à travers le scénario de requêtes donné en exemple. Pour une description plus complète du modèle, se référer à (Sandu Popa et al., 2008).

5.3. L'exemple d'application revisité

Les exemples donnés dans la section 2 nous permettent d'illustrer l'interface SQL du serveur de données et l'utilisation d'une partie des opérations définies dans l'algèbre. Nous reprenons ces exemples en déclinant pour chacun l'expression SQL correspondante. Les nouvelles opérations figurent en gras dans les requêtes. Nous avons utilisé des appellations assez parlantes pour plus de concision. Du fait que le

modèle est fermé, il est possible d'exprimer des requêtes plus complexes. Nous nous sommes exprès limités à des requêtes assez simples dans un but pédagogique.

Question 1 : « Disposant données des vitesses réglementaires sur une section déterminée, retourner le pourcentage de véhicules dont la vitesse instantanée est supérieure de 30km/h à la vitesse autorisée pour une section de route donnée ».

```
SELECT count(distinct mo_id)/nb_mo
FROM vehicle_trip, (SELECT count(distinct mo_id) AS nb_mo FROM vehicle_trip
WHERE intersects(g_speed, &section))
WHERE greaterThan(atline(g_speed, &section), &legalSpeed, 30) is not null;
```

Question 2 : « Etant donné une portion de route, obtenir le nombre d'objets mobiles avant/après l'installation du radar »

```
SELECT count(distinct mo_id) AS nbMOBefore FROM vehicle_trip
WHERE intersects(trajectory(trip), radarArea) = 1
AND max(deftime(trip)) < installationDate ;
```

```
SELECT count(distinct mo_id) AS nbMOAfter FROM vehicle_trip
WHERE intersects(trajectory(trip), radarArea) = 1
AND min(deftime(trip)) > installationDate
```

Question 3 : « Sur une période de temps donnée, étant donné un point d'origine et une destination, obtenir l'ensemble des trajets reliant ces deux points avant/après l'installation du radar »

```
SELECT mo_id, trajectory(trip) AS trajectBefore FROM vehicle_trip
WHERE intersects(trajectory(trip), originPoint) = 1
AND intersects(trajectory(trip), destinationPoint) = 1
AND max(deftime(trip)) < installationDate ;
```

```
SELECT mo_id, trajectory(trip) AS trajectAfter FROM vehicle_trip
WHERE intersects(trajectory(trip), originPoint) = 1
AND intersects(trajectory(trip), destinationPoint) = 1
AND min(deftime(trip)) > installationDate ;
```

Question 4 : « Etant donné une route particulière, obtenir le profil de vitesse de la « vitesse pratiquée » avant/après l'installation d'un radar automatique »

```
SELECT percentile(g_speed, 0.85) AS operatingSpeedBefore FROM vehicle_trip
WHERE intersects(trajectory(trip), cameraArea) = 1
AND max(deftime(trip)) < installationDate ;
```

```
SELECT percentile(g_speed, 0.85) AS operatingSpeedAfter FROM vehicle_trip
WHERE intersects(trajectory(trip), cameraArea) = 1
AND min(deftime(trip)) > installationDate ;
```

Question 5 : « Obtenir tous les profils de vitesse des véhicules sur les virages ayant une courbure comprise entre 450 et 500m ».

```
SELECT atgline(g_speed, curve), curve
FROM vehicle_trip, TABLE getCurves(network, radius)
WHERE inside(trajectory(trip), curve) = 1 ;
```

6. Conclusion

Nous avons décrit à travers cet article un modèle et un langage permettant l'analyse spatiotemporelle de la conduite en situation réelle. La particularité du système proposé est d'offrir en plus de la gestion d'objets mobiles contraints par le réseau, une modélisation spatiale des mesures des capteurs associés. Le concept de mesure variant dans l'espace ou dans un espace contraint est nouveau (I. Sandu Popa et al., 2008). Nous avons illustré à travers le scénario de requêtes l'intérêt de cette modélisation dans le cadre de l'application visée.

Ces types ainsi qu'une partie des opérations associées ont été implémentées et ont été fortement intégrés au SGBD Oracle 10g, grâce au mécanisme de *cartridge* offert par ce système (Oracle, 2005). Des travaux sur l'optimisation de requêtes ont été entamés, dont une partie est décrite dans (I. Sandu Popa et Zeitouni, 2008).

En perspective, il faudra compléter notre modèle par d'autres fonctions agrégats et des fonctions analytiques. Les techniques d'indexations doivent être étendues et leur analyse devra permettre des recommandations sur le paramétrage de ces index.

Références

- Brakatsoulas S., Pfoser D., Salas R., Wenk C.: On Map-Matching Vehicle Tracking Data. VLDB 2005: 853-864
- Grumbach S., Koubarakis M., Rigaux P., Scholl M. et Skiadopoulos S., Spatio-temporal Models and Languages: An Approach Based on Constraints, LNCS 2520, pp. 177-201, Springer, 2003.
- Güting R.H., Böhlen M.H., Erwig M., Jensen C.S., Lorentzos N.A., Schneider M., and Vazirgiannis M., A Foundation for Representing and Querying Moving Objects. ACM Transactions on Database Systems, 25(1): 1-42, 2000.
- Güting R.H., Schneider M.. Moving Objects Databases, Morgan Kaufmann, 2005
- Güting R.H., Almeida V.T. de, and Ding Z., Modeling and Querying Moving Objects in Networks. VLDB Journal 15:2 (2006), 165-190.
- Oracle Database Data Cartridge Developer's Guide, 10g Release 2, 2005.
- Sandu Popa I., Kharrat A., Zeitouni K., CALM : Un système de gestion de données de CAPteurs à Localisation Mobile, Conférence SAGEO, Montpellier, 2008.
- Sandu Popa I., Zeitouni K., Managing Sensor Data with Mobile Location, Rapport PRISM # 2008/2, <http://www.prism.uvsq.fr/rapports/bin/abstract.php?id=501>
- Wolfson, O.: Moving Objects Information Management: The Database Challenge. NGITS 2002: 75-89.